General Tensor Discriminant Analysis and Gabor Features for Gait Recognition

Dacheng Tao[1], Xuelong Li[1], Xindong Wu[2], and Stephen J. Maybank[1]

1. School of Computer Science and Information Systems, Birkbeck College, University of London, Malet Street, London WC1E 7HX, United Kingdom.
2. Department of Computer Science, University of Vermont, 33 Colchester Avenue, Burlington, Vermont 05405, United States of America.
{dacheng, xuelong, sjmaybank}@dcs.bbk.ac.uk; xwu@cs.uvm.edu.

Abstract— The traditional image representations are not suited to conventional classification methods, such as the linear discriminant analysis (LDA), because of the under sample problem (USP): the dimensionality of the feature space is much higher than the number of training samples. Motivated by the successes of the two dimensional LDA (2DLDA) for face recognition, we develop a general tensor discriminant analysis (GTDA) as a preprocessing step for LDA. The benefits of GTDA compared with existing preprocessing methods, e.g., principal component analysis (PCA) and 2DLDA, include 1) the USP is reduced in subsequent classification by, for example, LDA; 2) the discriminative information in the training tensors is preserved; and 3) GTDA provides stable recognition rates because the alternating projection optimization algorithm to obtain a solution of GTDA converges, while that of 2DLDA does not.

We use human gait recognition to validate the proposed GTDA. The averaged gait images are utilized for gait representation. Given the popularity of Gabor function based image decompositions for image understanding and object recognition, we develop three different Gabor function based image representations: 1) the GaborD representation is the sum of Gabor filter responses over directions, 2) GaborS is the sum of Gabor filter responses over scales, and 3) GaborSD is the sum of Gabor filter responses over scales and directions. The GaborD, GaborS and GaborSD representations are applied to the problem of recognizing people from their averaged gait images.

A large number of experiments were carried out to evaluate the effectiveness (recognition rate) of gait recognition based on first obtaining a Gabor, GaborD, GaborS or GaborSD image representation, then using GDTA to extract features and finally using LDA for classification. The proposed methods achieved good performance for gait recognition based on image sequences from the USF HumanID Database. Experimental comparisons are made with nine state of the art classification methods in gait recognition.

***Index Terms*—Gabor Gait, General Tensor Discriminant Analysis, Human Gait Recognition, Linear Discriminant Analysis, Tensor Rank, Visual Surveillance.**

I. INTRODUCTION

BIOMETRICS research is a hot topic because of the demanding requirements for automatic human authentication and authorization in computer systems. Biometric resources, such as iris, fingerprint, palmprint, and shoeprint, have been thoroughly studied and employed in many applications. These biometric resources have two disadvantages: 1) they do not work well in low resolution images, for example those taken at a distance; and 2) user cooperation is required to achieve good results.

Human gait, or manner of walking, is a biometric which does not suffer from the above two disadvantages. It can be obtained from images taken at a distance and it does not require user cooperation. Gait contains information about the walker's physical situation and about his or her psychological state. In certain cases gait information is sufficient to identify the person [9].

The original research on measuring human gait was entirely for medical purposes. For example, Murray [33] used gait to classify patients into groups suited to different types of medical treatment. This classification was achieved by comparing the patients' gait patterns against the gait patterns obtained from a control group. Johansson [18] obtained video sequences of people walking in a darkened room with lights

affixed to the major joints of the body. He reported that the walking people could be recognized by observers who were familiar with them.

In computer vision research human motion has been studied intensively [32], and there are a number of works on human gait analysis and recognition. The performance of gait recognition is affected by many factors [34][4], such as the silhouette quality, walking speed, dynamic/static component, elapsed time, shoes, carrying condition [2][17], physical and medical condition, disguise, indoor/outdoor, etc. The effects of different factors may be correlated. For example, a change in walking surface or shoe type may bring about a change in speed. Although gait is affected by so many factors, it is still useful for recognition [34].

The effective representation of gait is a key issue. Currently there are several successful representation models, such as appearance–based models [6][16][19][22][29][30][34][37][40], stochastic statistical models [19], articulated biomechanics models [36][3][1][7], in which a set of parameters describes the gait, and other parameter–based models [24][8]. Several of these models can be combined to further improve the representation of the gait. In this paper, we focus on appearance–based models for gait representation and recognition, because the existing appearance–based gait recognition methods achieve the best recognition rates obtained so far [16][30].

There are many appearance–based models for human gait recognition [16][30]. Some models use the silhouette of the entire body [28][16][19] while others use the most discriminant parts [12][21], such as the torso and the thighs. In the following, we use the averaged gait image [16][30] as the appearance model. The effectiveness of the averaged gait image for recognition is shown experimentally in [16][30].

In this paper, the averaged gait image is decomposed by Gabor filtering [38]. We combine the decomposed images to give a new representation, which is suitable for recognition. There are three major reasons for introducing the Gabor based representation for the averaged gait image based recognition: 1) human brains seem to have a special function to process information in multi–resolution levels [31][10][11], which can be simulated by controlling the scale parameter in Gabor functions; 2) it is supposed that Gabor functions are similar to the receptive field profiles in the mammalian cortical simple cells [31][10][11]; and 3)

Gabor function based representations have been successfully employed in many computer vision applications, such as face recognition [27][25], gait recognition, and texture analysis [14].

Although Gabor function based representations are effective for object recognition [27][25] and image understanding, the computational cost of the representations is high. Therefore, three simplified Gabor function representations are introduced. Each representation is obtained by filtering the image with a certain sum of Gabor functions. This reduces the number of filtering operations that are required. The three image representations are: 1) the sum over directions of Gabor function based representations (GaborD), 2) the sum over scales of Gabor function based representations (GaborS), and 3) the sum over scales and directions of Gabor function based representations (GaborSD).

In the averaged gait image based recognition, the dimensionality of the feature space is usually much larger than the size of the training set. This is known as the under sample problem (USP). Conventional classifiers, e.g., LDA, often fail when faced by USP. One solution is to reduce the dimensionality of the feature space using principal component analysis (PCA) [26][16] or multilinear subspace analysis (MSA) [39], but unfortunately, some useful information is discarded by PCA and MSA.

To reduce USP and to preserve discriminative information, we propose the general tensor discriminant analysis (GTDA) as a pre-processing step for conventional classifiers, such as LDA. GTDA is motivated by the successes of the two dimensional LDA (2DLDA) [43] in face recognition. The 2DLDA reduces USP, because it operates on matrices (in [43] grey level face images) directly and constructs projection matrices for each order (the row space or the column space) of the face images, i.e., 2DLDA preserves the matrix structure of the data. Moreover, 2DLDA preserves the information needed for classification, because it considers the class label information of the face images when constructing projection matrices. By utilizing 2DLDA as a pre-processing step for LDA, the face recognition rate is improved. However, 2DLDA is not stable, because the optimization algorithm used in the training stage of 2DLDA fails to converge. Therefore, it is impossible to obtain optimal projection matrices with respect to the criterion used in 2DLDA. In order to obtain convergence at the training stage, the differential scatter discriminant criterion (DSDC), which is a

variation of LDA, is combined with the tensor representation to give GTDA. We optimize GTDA through the alternating projection method. Mathematical arguments show that the alternating projection method converges, and convergence is verified by the experimental results in Section VI.C. The combination of DSDC with the tensor representation of data is based on a reformulation of the DSDC in terms of tensor algebra. The result is GTDA. The benefits of GTDA are as follows: 1) GTDA operates on each order of the training tensors separately to reduce USP; 2) GTDA preserves the discriminative information in the training tensors by taking the class label information into account; and 3) the optimization algorithm (the training stage) of GTDA converges as proved mathematically in this paper. When the data is pre-processed by GTDA, the dimensionality of the feature space is significantly reduced to the point where LDA can be used for recognition. A large number of experimental results show the effectiveness of the proposed Gabor representations when combined with GTDA+LDA. A comparison with existing algorithms is made.

The rest of the paper is organized as follows. Section II describes the Gabor based gait representations. In Section III, tensor algebra is briefly introduced. LDA is briefly reviewed in Section IV. In Section V, we propose GTDA and prove that the optimization algorithm for GTDA converges. The USF HumanID database is described in Section VI and a comparison of the proposed methods with many existing algorithms is made. Section VII concludes.

II. Gabor Gait Representation

As demonstrated in [16][30], the averaged gait image is a robust feature for gait recognition tasks. In Fig. 1, the sample averaged gait images are shown for nine different people under different circumstances. It can be observed that: 1) the averaged gait images of the same person under different circumstances share similar visual effects; and 2) the averaged gait images of different people even under the same circumstance are very different. So, it is possible to recognize a person by his/her averaged gait images. Furthermore, according to research results reported in [31][23][11][10], Gabor functions based image decomposition is biologically

relevant to and is useful for image understanding and recognition. Consequently, it is reasonable to use Gabor functions to find averaged gait images.

Fig. 1. The columns show the averaged gait images of nine different people in the Gallery of the USF database. The four rows in the figure from top to bottom are based on images taken from the Gallery, ProbeB, ProbeH, and ProbeK, respectively. The averaged gait images in each column come from the same person.

A. Gabor Functions

Marcelja [31] and Daugman [10][11] modeled the responses of the visual cortex by Gabor functions, because they are similar to the receptive field profiles in the mammalian cortical simple cells. Daugman [10][11] developed the 2D Gabor functions, a series of local spatial bandpass filters, which have good spatial localization, orientation selectivity, and frequency selectivity. Lee [23] gives a good introduction to image representation using Gabor functions. A Gabor (wavelet, kernel, or filter) function is the product of an elliptical Gaussian envelope and a complex plane wave, defined as:

$$\psi_{s,d}\left(x,y\right)=\psi_{\bar{k}}\left(\bar{x}\right)=\frac{\left\|\bar{k}\right\|}{\delta^2}\cdot e^{-\frac{\left\|\bar{k}\right\|^2\cdot\left\|\bar{x}\right\|^2}{2\delta^2}}\cdot\left[e^{i\bar{k}\cdot\bar{x}}-e^{-\frac{\delta^2}{2}}\right], \tag{1}$$

where $\bar{x}=\left(x,y\right)$ is the variable in a spatial domain and $\bar{k}$ is the frequency vector, which determines the scale and direction of Gabor functions, $\bar{k}=k_s e^{i\phi_d}$, where $k_s=k_{\max}/f^s$, $k_{\max}=\pi/2$. In our application, $f=2$, $s=0,1,2,3,4$, and $\phi_d=\pi d/8$, for $d=0,1,2,3,4,5,6,7$. The term $\exp\left(-\sigma^2/2\right)$ is subtracted in order to make the kernel DC–free, and thus insensitive to illumination. Examples of the real part of Gabor

functions used in this paper are presented in Fig. 2. We use Gabor functions with five different scales and eight different orientations, making a total of forty Gabor functions. The number of oscillations under the Gaussian envelope is determined by $\delta = 2\pi$.

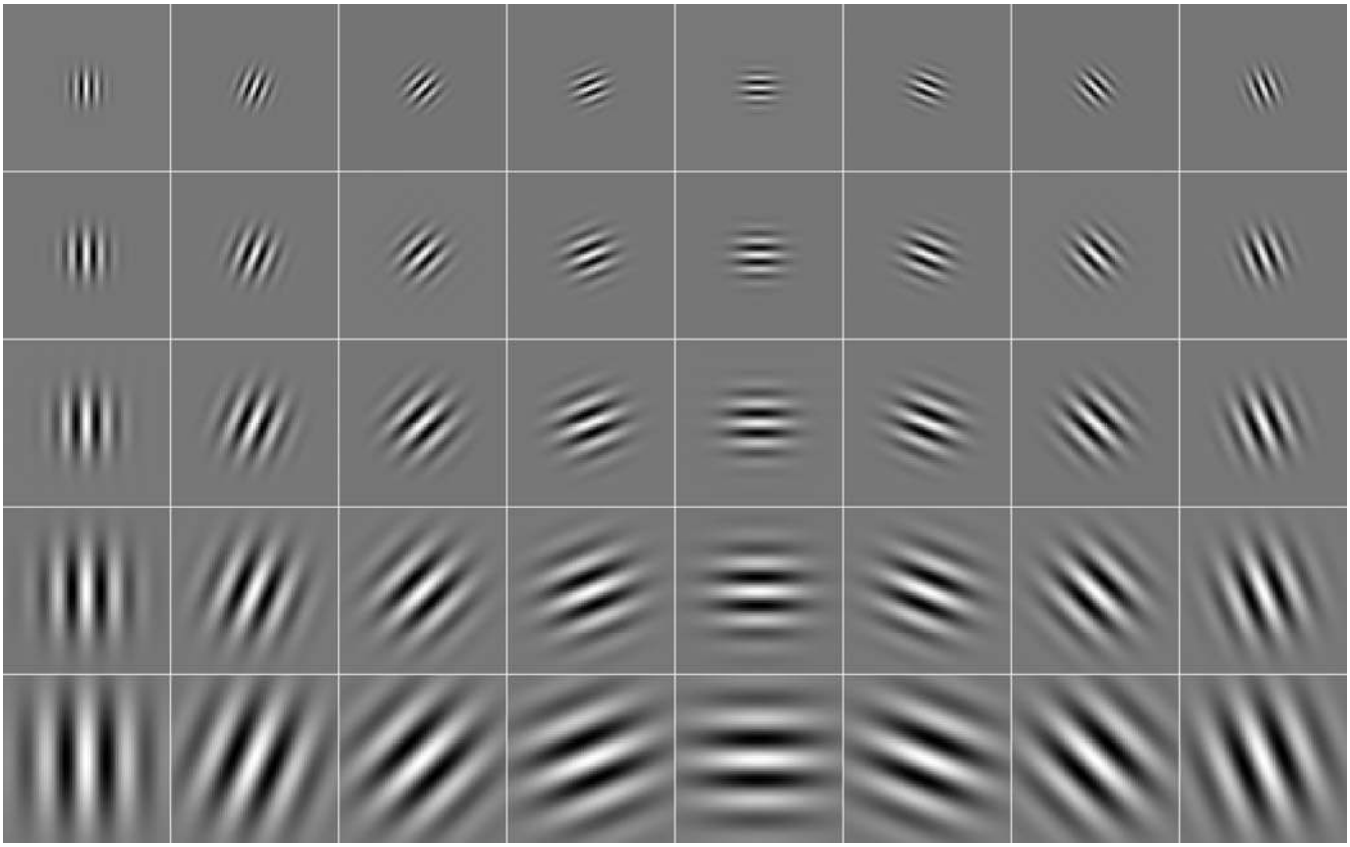

Fig. 2. The real part of Gabor functions for five different scales and eight different directions.

B. Gabor based Gait Representation

The Gabor function representation of a gait image is obtained by convolving the Gabor functions with the averaged gait image. The result is a 4^{th} order tensor in $R^{N_1 \times N_2 \times 5 \times 8}$. The first two indices give the pixel location; the third index gives the value of the scale and the fourth index gives the direction. The entries of the 4^{th} order tensor are complex numbers and the magnitude part of this 4^{th} order tensor is defined as the Gabor gait, as shown in Fig. 3. In Gabor gait, there are 40 components (images), and each one is the magnitude part of the output, which is obtained by convolving the averaged gait image with a Gabor function.

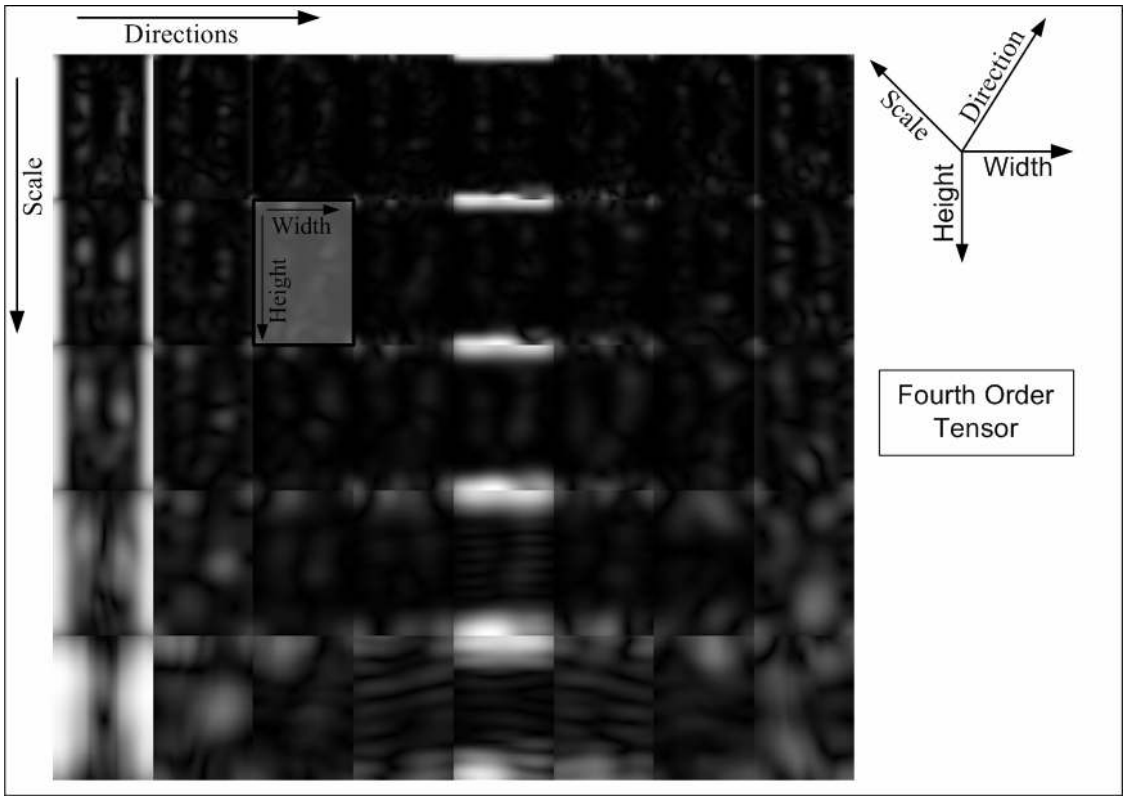

Fig. 3. Gabor gait: the rows show different scales and the columns show different directions for an averaged gait image.

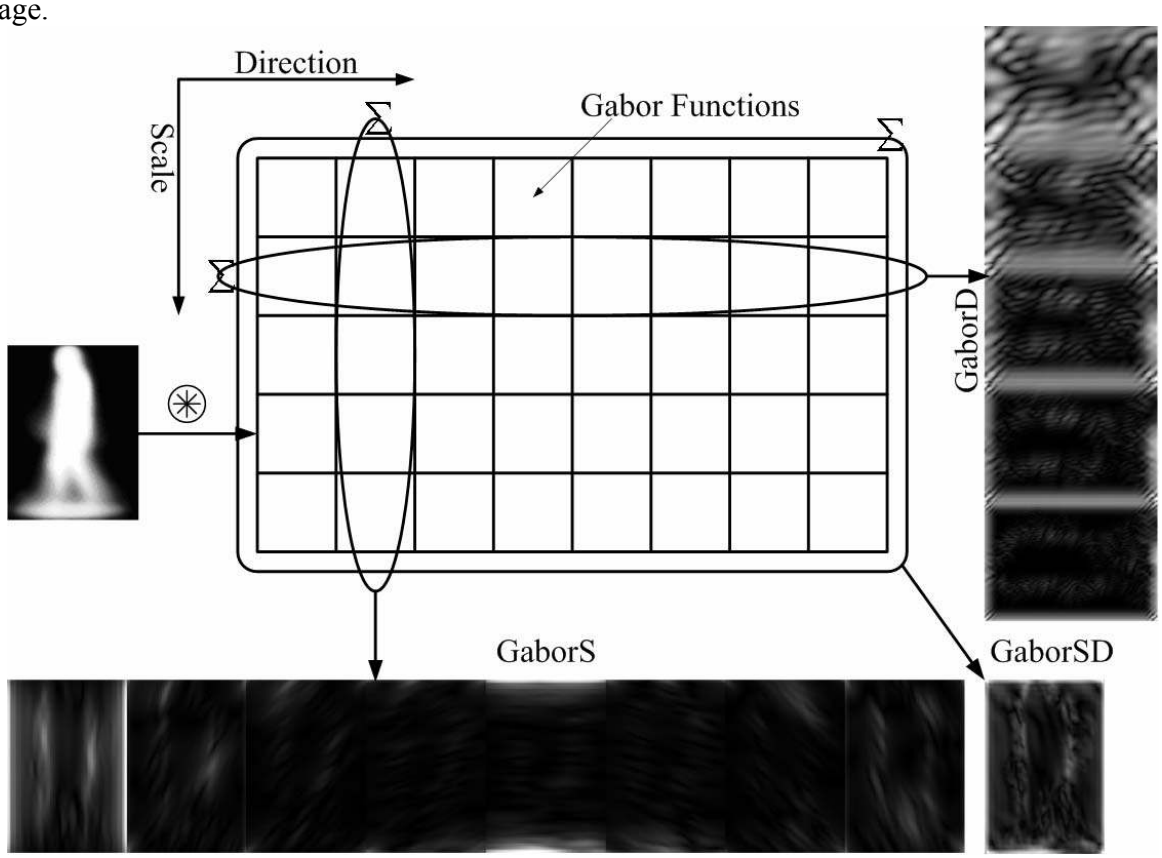

Fig. 4. Three new methods for averaged gait image representation using Gabor functions: GaborS, GaborD, and GaborSD.

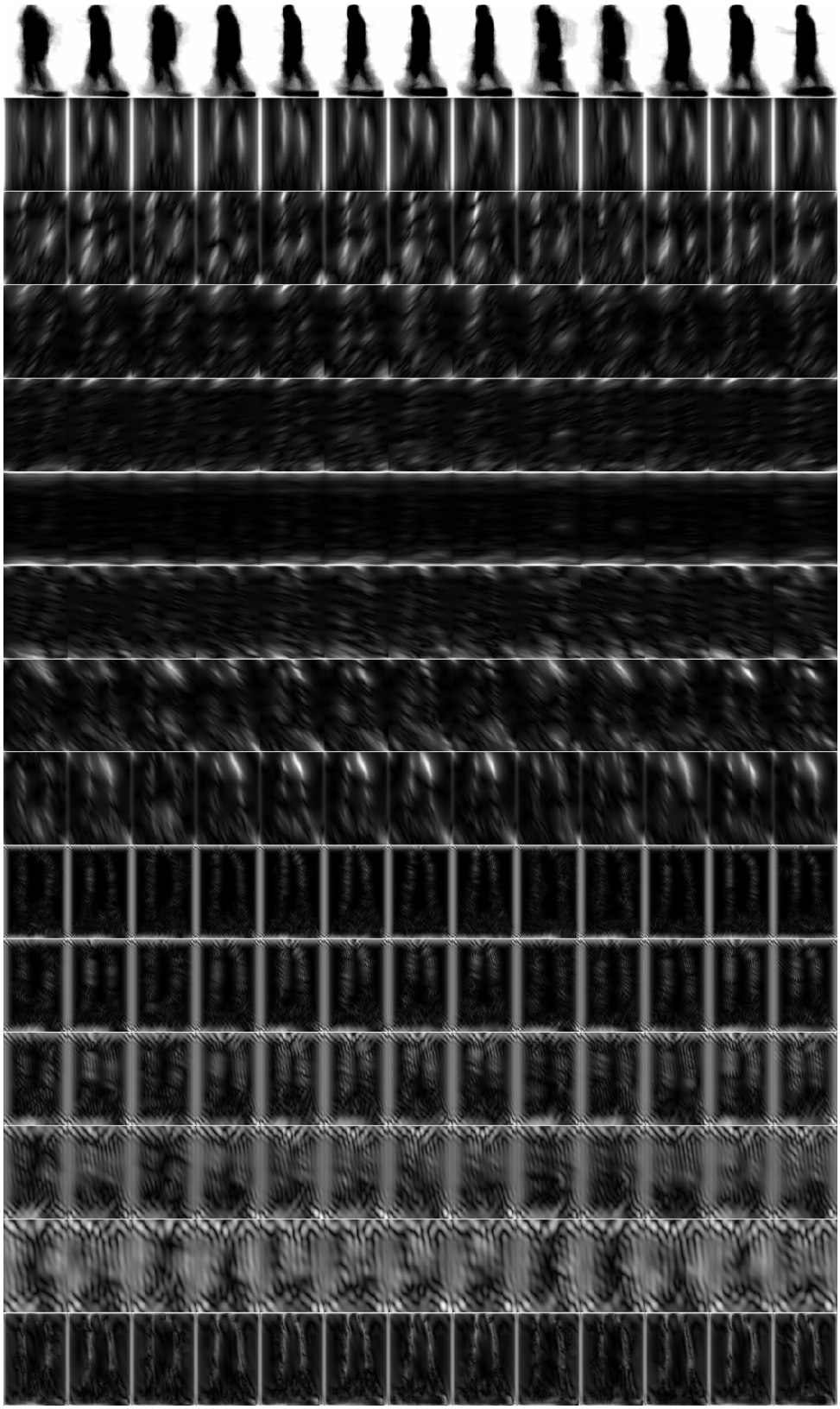

Fig. 5. The thirteen columns are Gallery gait, ProbeA gait, ProbeB gait, ProbeC gait, ProbeD gait, ProbeE gait, ProbeF gait, ProbeG gait, ProbeH gait, ProbeI gait, ProbeJ gait, ProbeK gait, and ProbeL gait, respectively. The rows are the original gait, GaborD (from 0 to 5), GaborS (from 0 to 7), and GaborSD, respectively.

The gait representation method in Fig. 3 is similar to the face representation method [26][25], which is also Gabor functions. Although this method for representation is powerful, its computational costs both for recognition and calculation for representation are high compared with the original image based recognition. The computational cost in recognition is analyzed in Section V.D.

We introduce three new representations of the averaged gait images. These are the sum over directions of Gabor functions based representation (GaborD), the sum over scales of Gabor functions based representation (GaborS), and the sum over scales and directions of Gabor functions based representation (GaborSD). The most important benefit of these new representations is that the cost of computing them is low. The computational cost of the Gabor based representation is given at the end of this Section and the complexity analysis for alternating projection optimization procedure for GTDA based dimension reduction (training cost) with different representations is given in Section V.D.

GaborD is the magnitude part of the outputs generated by convolving an averaged gait image with the sum of Gabor functions over the eight directions with the scale fixed,

$$\mathrm{GaborD}(x,y)=\left|\sum_{d} I(x,y)*\psi_{s,d}(x,y)\right|=\left|I(x,y)*\sum_{d}\psi_{s,d}(x,y)\right|, \tag{3}$$

where $I(x,y)$ is the averaged gait image; $\psi_{s,d}(x,y)$ is the Gabor function defined in (1); and $\mathrm{GaborD}(x,y)$ is the output of the GaborD method for representation. Therefore, we have five different outputs to represent the original gait image in the GaborD decomposition. We combine the $\mathrm{GaborD}(x,y)$ for the different scales into a 3rd order tensor $\mathbf{G}_D$ in $R^{N_1\times N_2\times 5}$. The first two indices are for the pixel locations and the third index is for the scale. The calculations are shown in Fig. 4. Examples of GaborD based gait representation are shown in Fig. 5.

GaborS is the magnitude part of the outputs generated by convolving an averaged gait image with the sum of Gabor functions over the five scales with the direction fixed,

$$\text{GaborS}(x,y)=\left|\sum_{s} I(x,y)*\psi_{s,d}(x,y)\right|=\left|I(x,y)*\sum_{s}\psi_{s,d}(x,y)\right|, \quad (2)$$

where $I(x,y)$ is the averaged gait image, $\psi_{s,d}(x,y)$ is the Gabor function defined in (1), and $\text{GaborS}(x,y)$ is the output of the GaborS method for representation. Therefore, we have eight different outputs to represent the original gait image in the GaborS decomposition. We combine the $\text{GaborS}(x,y)$ with different directions together as a 3rd order tensor $\mathbf{G}_S$ in $R^{N_1\times N_2\times 8}$. The third index is for the directions. The calculations are shown in Fig. 4. Examples of GaborS based gait representation are shown in Fig. 5.

GaborSD is the magnitude part of the output generated by convolving an averaged gait image with the sum of all forty Gabor functions,

$$\text{GaborSD}(x,y)=\left|\sum_{s}\sum_{d} I(x,y)*\psi_{s,d}(x,y)\right|=\left|I(x,y)*\sum_{s}\sum_{d}\psi_{s,d}(x,y)\right|, \quad (4)$$

where $I(x,y)$ is the averaged gait image, $\psi_{s,d}(x,y)$ is the Gabor function defined in (1), and $\text{GaborSD}(x,y)$ is the output of the GaborSD method for representation. It is a 2nd order tensor in $R^{N_1\times N_2}$. The calculation procedure is shown in Fig. 4. Examples of GaborD based gait representation are shown in Fig. 5.

C. Computational Complexity

In this paper, all the filters, which are used as discrete approximations to Gabor functions with different scales and directions, have a fixed size $G_1\times G_2$ (in experiments, $G_1=G_2=64$). The averaged gait images are in $R^{N_1\times N_2}$. Therefore, the time complexities for generating a Gabor gait in $R^{N_1\times N_2\times 5\times 8}$, a GaborD gait in $R^{N_1\times N_2\times 5}$, a GaborS gait in $R^{N_1\times N_2\times 8}$, and a GaborSD gait in $R^{N_1\times N_2}$ are $O(40N_1N_2G_1G_2)$, $O(5N_1N_2G_1G_2)$, $O(8N_1N_2G_1G_2)$, and $O(N_1N_2G_1G_2)$, respectively. Based on this observation, the GaborD, GaborS, and GaborSD based gait representation can reduce the computational complexity of the Gabor based representation, because the numbers of filters (the sum of Gabor function) for decomposition in the

GaborD/GaborS/ GaborSD based representation are smaller than the number of filters (Gabor functions) for decomposition in Gabor based representation. The computational costs of different representations are ordered as: Gabor > GaborS > GaborD > GaborSD. The experiments in Section VI.B show that GaborD and GaborS based representations perform slightly better than Gabor based representation for gait recognition.

III. TENSOR ALGEBRA

Tensors are multidimensional arrays of numbers which transform linearly under coordinate transformations [20]. The order of a tensor $\mathbf{X} \in R^{N_1 \times N_2 \times \ldots \times N_M}$ is M. An element of $\mathbf{X}$ is denoted by $\mathbf{X}_{n_1, n_2, \ldots, n_M}$, where $1 \le n_i \le N_i$ and $1 \le i \le M$. We introduce the following definitions [20] relevant to this paper.

Definition 2.1 (Tensor Product) The tensor product $\mathbf{X} \otimes \mathbf{Y}$ of a tensor $\mathbf{X} \in R^{N_1 \times N_2 \times \ldots \times N_M}$ and another tensor $\mathbf{Y} \in R^{N'_1 \times N'_2 \times \ldots \times N'_{M'}}$ is the tensor defined by

$$\left(\mathbf{X} \otimes \mathbf{Y}\right)_{n_1 \times n_2 \times \ldots \times n_M \times n'_1 \times n'_2 \times \ldots \times n'_{M'}} = \mathbf{X}_{n_1 \times n_2 \times \ldots \times n_M} \mathbf{Y}_{n'_1 \times n'_2 \times \ldots \times n'_{M'}}, \tag{5}$$

for all index values.

Definition 2.2 (Mode–*d* Matricizing or Matrix Unfolding) The mode–*d* matricizing or matrix unfolding of an *M*th order tensor $\mathbf{X} \in R^{N_1 \times N_2 \times \ldots \times N_M}$ is a matrix $X_{(d)} \in R^{N_d \times N_{\bar{d}}}$, which is the ensemble of vectors in R^{N_d} obtained by keeping index i_d fixed and varying the other indices. Here $N_{\bar{d}} = \left(\prod_{i \ne d} N_i\right)$. We denote the mode–*d* matricizing of $\mathbf{X}$ as $\text{mat}_d\left(\mathbf{X}\right)$ or $X_{(d)}$.

Definition 2.3 (Tensor Contraction) The contraction of a tensor is obtained by equating two indices and summing over all values of the repeated indices. Contraction reduces the tensor order by 2. A notation for contraction is Einstein's summation convention[1]. For example, given two vectors $x, y \in R^N$; the tensor product of x and y is $Z = x \otimes y$; and the contraction of Z is $Z_{ii} = x \cdot y$, where the repeated indices imply

[1] "*When any two subscripts in a tensor expression are given the same symbol, it is implied that the convention is formed.*" A. Einstein, Die Grundlage der Allgemeinen Relativitatstheorie, Ann. Phys., pp. 49-769, 1916.

summation. The value of Z_{ii} is the inner product of x and y. In general, for tensors $\mathbf{X} \in R^{N_1 \times N_2 \times \ldots \times N_M \times K_1 \times K_2 \times \ldots \times K_L}$ and $\mathbf{Y} \in R^{N_1 \times N_2 \times \ldots \times N_M \times P_1 \times P_2 \times \ldots \times P_Q}$, the contraction on the tensor product $\mathbf{X} \otimes \mathbf{Y}$ is

$$\left[\!\left[\mathbf{X} \otimes \mathbf{Y}; (1:M)(1:M) \right]\!\right] = \sum_{n_1=1}^{N_1} \cdots \sum_{n_M=1}^{N_M} (\mathbf{X})_{n_1 \times \ldots \times n_M \times k_1 \times \ldots k_L} (\mathbf{Y})_{n_1 \times \ldots \times n_M \times p_1 \times p_2 \times \ldots \times p_Q} . \tag{6}$$

In this paper, when the contraction is conducted on all indices except the i^{th} index on the tensor product of $\mathbf{X}$ and $\mathbf{Y}$ in $R^{N_1 \times N_2 \times \ldots \times N_M}$, we denote this procedure as

$$\begin{aligned} \left[\!\left[\mathbf{X} \otimes \mathbf{Y}; (\bar{i})(\bar{i}) \right]\!\right] &= \left[\!\left[\mathbf{X} \otimes \mathbf{Y}; (1:i-1, i+1:M)(1:i-1, i+1:M) \right]\!\right] \\ &= \sum_{n_1=1}^{N_1} \cdots \sum_{n_{i-1}=1}^{N_{i-1}} \sum_{n_{i+1}=1}^{N_{i+1}} \cdots \sum_{n_M=1}^{N_M} (\mathbf{X})_{n_1 \times \ldots \times n_{i-1} \times n_i \times n_{i+1} \times \cdots \times n_M} (\mathbf{Y})_{n_1 \times \ldots \times n_{i-1} \times n_i \times n_{i+1} \times \ldots \times n_M} , \\ &= \mathrm{mat}_i(\mathbf{X}) \mathrm{mat}_i^T(\mathbf{Y}) = X_{(i)} Y_{(i)}^T \end{aligned} \tag{7}$$

and $\left[\!\left[\mathbf{X} \otimes \mathbf{Y}; (\bar{i})(\bar{i}) \right]\!\right] \in R^{N_i \times N_i}$.

Definition 2.3 (Mode–*d* product) The mode–*d* product $\mathbf{X} \times_d U$ of a tensor $\mathbf{X} \in R^{N_1 \times N_2 \times \ldots \times N_M}$ and a matrix $U \in R^{N'_d \times N_d}$ is the $N_1 \times N_2 \times \cdots \times N_{d-1} \times N'_d \times N_{d+1} \times \cdots \times N_M$ tensor defined by

$$(\mathbf{X} \times_d U)_{i_1 \times i_2 \times \cdots \times i_{d-1} \times j \times i_{d+1} \times \cdots \times i_M} = \sum_{i_d} \left(\mathbf{X}_{i_1 \times i_2 \times \cdots \times i_{d-1} \times i_d \times i_{d+1} \times \cdots \times i_M} U_{j \times i_d} \right) = \left[\!\left[\mathbf{X} \otimes U; (d)(2) \right]\!\right], \tag{8}$$

for all index values. The mode–*d* product is a type of contraction.

To simplify the notation in this paper, we denote

$$\mathbf{X} \times_1 U_1 \times_2 U_2 \times \cdots \times_M U_M \triangleq \mathbf{X} \prod_{i=1}^{M} \times_i U_i \tag{9}$$

and

$$\mathbf{X} \times_1 U_1 \times \cdots \times_{i-1} U_{i-1} \times_{i+1} U_{i+1} \times \cdots \times_M U_M = \mathbf{X} \prod_{d=1; d \neq i}^{M} \times_d U_d \triangleq \mathbf{X} \bar{\times}_i U_i . \tag{10}$$

IV. Linear Discriminant Analysis

Given a number of training samples $x_{i;j} \in R^N$ in c known classes, where i is the class number, $1 \le i \le c$, and j is the sample ID in the i^{th} class with $1 \le j \le n_i$, the aim of LDA is to find a projection of the $x_{i;j}$, which

is optimal for separating the different classes in a low dimensional space. In the training set, there are $n = \sum_{i=1}^{c} n_i$ samples; the mean vector for the individual class C_i is $m_i = (1/n_i)\sum_{j=1}^{n_i} x_{i;j}$; and the total mean vector is $m = (1/n)\sum_{i=1}^{c}\sum_{j=1}^{n_i} x_{i;j}$. The between-class scatter matrix S_b and the within-class scatter matrix S_w are:

$$S_b = \frac{1}{n}\sum_{i=1}^{c} n_i (m_i - m)(m_i - m)^T \quad S_w = \frac{1}{n}\sum_{i=1}^{c}\sum_{j=1}^{n_i} (x_{i;j} - m_i)(x_{i;j} - m_i)^T . \tag{11}$$

The projection, which is defined by a set of vectors $U = [u_1, \ldots, u_{c-1}]$, is chosen to maximize the ratio between the trace of S_b and the trace of S_w:

$$U^* = \arg\max_U \frac{\mathrm{tr}(U^T S_b U)}{\mathrm{tr}(U^T S_w U)}. \tag{12}$$

The projection matrix U^* can be computed from the leading eigenvectors of $S_w^{-1} S_b$. If x is a new feature vector, then it is projected to $y = U^T (x - m)$. The vector y is used in place of x for representation and classification. If c is 2, LDA reduces to the Fisher linear discriminant. In many computer vision applications, the dimensionality N of the feature space is much larger than the size of the training set (i.e., $N \gg n$). Since the rank of S_w is at most $n - c$, S_w is singular if N is large, and it is more difficult to construct U. This is known as the Under Sample Problem (USP).

To reduce the USP, the regularization method [15] is widely used, although it is not optimal. Small quantities are added to the diagonal entries of the scatter matrices. In recent face recognition research, several new algorithms have been proposed to deal with USP, such as the direct linear discriminant analysis (DLDA) [44] and the null–space linear discriminant analysis (NLDA) [5]. DLDA utilizes discriminant vectors, which are the eigenvectors of S_w associated with the smallest eigenvalues after discarding those eigenvectors of S_w, which are in the null space of S_b; while in NLDA, discriminant information is extracted from the null space of S_w. It can be observed that in DLDA and NLDA, some discriminant information may

be lost, because the null space of S_b and the principle space (spanned by the eigenvectors of S_w associated with the largest eigenvalues) of S_w, which are removed in DLDA and NLDA, may contain some discriminant information. Aiming at utilizing all the discriminant information provided by S_b and S_w, Wang and Tang [42] developed the dual-space method, but they require a threshold which is difficult to determine in applications. In face recognition, PCA+LDA [1][35] (i.e., PCA serves as a pre-processing step for LDA) is used to reduce USP, and the method also achieves good performance in human gait recognition. However, some important discriminant information is lost in the PCA stage.

V. General Tensor Discriminant Analysis

Human gait images are naturally represented by second–order tensors (matrices), or third–order tensors in the case of image sequences. Traditionally, such high order tensors are scanned into vectors (vectorization) to meet the input requirements of data processing techniques such as PCA and LDA. However, during vectorization, a great deal of useful structure information is lost. The dimensionality of the resulting vectors is much larger than the number of examples in the training set which leads to the USP. To reduce USP and to preserve the discriminant information in the original data, we propose the general tensor discriminant analysis (GTDA), which is a tensor extension of the differential scatter discriminant criterion (DSDC).

A. *Differential Scatter Discriminant Criterion* (*DSDC*)

The Differential Scatter Discriminant Criterion (DSDC) [15] is defined by,

$$U^* = \arg\max_{U^T U = I} \left(\mathrm{tr}\left(U^T S_b U \right) - \zeta \, \mathrm{tr}\left(U^T S_w U \right) \right), \tag{13}$$

where ζ is a tuning parameter; $U \in R^{N \times N^*}$ ($N^* \ll N$), constrained by $U^T U = I$, is the projection matrix; and S_b, S_w are defined in (11).

According to [15] (pp.446-447), the solution to (13) is equivalent to the solution to (12), if ζ is the Lagrange multiplier in (13). If we extract only one feature, i.e. U degenerates to a vector, then

$\zeta = \lambda_{\max}\left(S_w^{-1}S_b\right)$, which is the maximum eigenvalue of $S_w^{-1}S_b$. If we want to extract N^* features simultaneously, we estimate ζ as $\sum_{i=1}^{N^*}\lambda_i$, where $\lambda_i\,|_{i=1}^{t}$ are the largest N^* eigenvalues of $S_w^{-1}S_b$. From [15] (pp. 446-447), it is not difficult to show that the optimal ζ in (13) is $\mathrm{tr}\left(U_{opt}^T S_b U_{opt}\right)\Big/\mathrm{tr}\left(U_{opt}^T S_w U_{opt}\right)$[2]. An accurate solution of (13) can be obtained by the alternating projection method. Here, we use the approximation (on setting ζ as the maximum eigenvalue of $S_w^{-1}S_b$) in (13) to avoid the alternating projection method for optimization.

In real-world applications, because the distribution of the test set diverges from the distribution of the training set, a manually chosen value of ζ always achieves better prediction results than the calculated value. However, manually setting ζ is not practical for real applications, because we do not know which ζ is suitable for classification. In this paper, we automatically select ζ during the training procedure.

B. *General Tensor Discriminant Analysis*

On defining S_b and S_w by (11), it follows from (13) that

$$\begin{aligned}
U^* &= \arg\max_{U^TU=I}\left(\mathrm{tr}\left(U^TS_bU\right)-\zeta\,\mathrm{tr}\left(U^TS_wU\right)\right)\\
&= \arg\max_{U^TU=I}\left(\mathrm{tr}\left(U^T\left[\sum_{i=1}^{c}n_i\left(m_i-m\right)\left(m_i-m\right)^T\right]U\right)-\zeta\,\mathrm{tr}\left(U^T\left[\sum_{i=1}^{c}\sum_{j=1}^{n_i}\left(x_{i;j}-m_i\right)\left(x_{i;j}-m_i\right)^T\right]U\right)\right)\\
&= \arg\max_{U^TU=I}\left(\mathrm{tr}\left(\sum_{i=1}^{c}n_iU^T\left[\left(m_i-m\right)\left(m_i-m\right)^T\right]U\right)-\zeta\,\mathrm{tr}\left(\sum_{i=1}^{c}\sum_{j=1}^{n_i}U^T\left[\left(x_{i;j}-m_i\right)\left(x_{i;j}-m_i\right)^T\right]U\right)\right)\\
&= \arg\max_{U^TU=I}\left(\sum_{i=1}^{c}n_i\,\mathrm{tr}\left(U^T\left[\left(m_i-m\right)\left(m_i-m\right)^T\right]U\right)-\zeta\sum_{i=1}^{c}\sum_{j=1}^{n_i}\mathrm{tr}\left(U^T\left[\left(x_{i;j}-m_i\right)\left(x_{i;j}-m_i\right)^T\right]U\right)\right)
\end{aligned} \tag{14}$$

[2] The derivative of $\mathrm{tr}\left(U^TS_bU\right)-\zeta\,\mathrm{tr}\left(U^TS_wU\right)$ with U is given by $S_bU-\zeta S_wU$. To obtain the optimal solution of (13), we need to set $S_bU-\zeta S_wU$ equal to 0 (as we have a strict condition here, i.e., $\left(S_b-\zeta S_w\right)u_k=0$, $\forall u_k\in U$, u_k is a column vector in U). Consequently, we have $\mathrm{Tr}\left(U_{opt}^TS_bU_{opt}\right)=\zeta\,\mathrm{Tr}\left(U_{opt}^TS_wU_{opt}\right)$, where U_{opt} is the optimal solution of (13).

$$= \arg\max_{U^T U = I} \left(\sum_{i=1}^{c} n_i \operatorname{tr}\left(U^T \left(m_i - m \right) \left[U^T \left(m_i - m \right) \right]^T \right) - \zeta \sum_{i=1}^{c} \sum_{j=1}^{n_i} \operatorname{tr}\left(U^T \left(x_{i;j} - m_i \right) \left[U^T \left(x_{i;j} - m_i \right) \right]^T \right) \right)$$

$$= \arg\max_{U^T U = I} \begin{pmatrix} \sum_{i=1}^{c} n_i \left[\!\left[\left(\left(m_i - m \right) \times_1 U^T \right) \otimes \left(\left(m_i - m \right) \times_1 U^T \right); (1)(1) \right]\!\right] \\ -\zeta \sum_{i=1}^{c} \sum_{j=1}^{n_i} \left[\!\left[\left(\left(x_{i;j} - m_i \right) \times_1 U^T \right) \otimes \left(\left(x_{i;j} - m_i \right) \times_1 U^T \right); (1)(1) \right]\!\right] \end{pmatrix},$$

where $\|\cdot\|_{Fro}$ is the Frobenius norm and the projection matrix $U \in R^{N \times N^*}$ ($N^* < N$) is constrained by $U^T U = I$.

Let $\mathbf{X}_{i;j}$ denote the j^{th} training sample (tensor) in the i^{th} individual class C_i, $\mathbf{M}_i = \left(1/n_i\right) \sum_{j=1}^{n_i} \mathbf{X}_{i;j}$ is the class mean tensor of the i^{th} class, $\mathbf{M} = \left(1/n\right) \sum_{i=1}^{c} n_i \mathbf{M}_i$ is the total mean tensor of all training tensors, and U_l denotes the l^{th} projection matrix obtained during the training procedure. Moreover, $\mathbf{X}_{i;j} \,|_{1 \le i \le c}^{1 \le j \le n_i}$, $\mathbf{M}_i \,|_{i=1}^{c}$, and $\mathbf{M}$ are all M^{th}–order tensors that lie in $R^{N_1 \times N_2 \times \ldots N_M}$. Based on an analogy with (14), we define GTDA by replacing $x_{i;j}$, m_i, and m with $\mathbf{X}_{i;j}$, $\mathbf{M}_i$, and $\mathbf{M}$, respectively, as:

$$U_l^* \,|_{l=1}^{M} = \arg\max_{U_l^T U_l = I} \begin{pmatrix} \sum_{i=1}^{c} n_i \left[\!\left[\left(\left(\mathbf{M}_i - \mathbf{M} \right) \prod_{k=1}^{M} \times_k U_k^T \right) \otimes \left(\left(\mathbf{M}_i - \mathbf{M} \right) \prod_{k=1}^{M} \times_k U_k^T \right); (1:M)(1:M) \right]\!\right] \\ -\zeta \sum_{i=1}^{c} \sum_{j=1}^{n_i} \left[\!\left[\left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i \right) \prod_{k=1}^{M} \times_k U_k^T \right) \otimes \left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i \right) \prod_{k=1}^{M} \times_k U_k^T \right); (1:M)(1:M) \right]\!\right] \end{pmatrix}. \tag{15}$$

The problem defined in (15) does not have a closed form solution, so we choose to use the alternating projection method, which is an iterative procedure, to obtain a numerical solution. Therefore, (15) is decomposed into *M* different optimization sub-problems, as follows,

$$\begin{aligned} U_l^* \,|_{l=1}^{M} &= \arg\max_{U_l \,|_{l=1}^{M}} \begin{pmatrix} \sum_{i=1}^{c} n_i \left[\!\left[\left(\left(\mathbf{M}_i - \mathbf{M} \right) \prod_{k=1}^{M} \times_k U_k^T \right) \otimes \left(\left(\mathbf{M}_i - \mathbf{M} \right) \prod_{k=1}^{M} \times_k U_k^T \right); (1:M)(1:M) \right]\!\right] \\ -\zeta \sum_{i=1}^{c} \sum_{j=1}^{n_i} \left[\!\left[\left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i \right) \prod_{k=1}^{M} \times_k U_k^T \right) \otimes \left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i \right) \prod_{k=1}^{M} \times_k U_k^T \right); (1:M)(1:M) \right]\!\right] \end{pmatrix} \\ &= \arg\max_{U_l \,|_{l=1}^{M}} \begin{pmatrix} \sum_{i=1}^{c} n_i \left[\!\left[\left(\left(\mathbf{M}_i - \mathbf{M} \right) \bar{\times}_l U_l^T \times_l U_l^T \right) \otimes \left(\left(\mathbf{M}_i - \mathbf{M} \right) \bar{\times}_l U_l^T \times_l U_l^T \right); (1:M)(1:M) \right]\!\right] \\ -\zeta \sum_{i=1}^{c} \sum_{j=1}^{n_i} \left[\!\left[\left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i \right) \bar{\times}_l U_l^T \times_l U_l^T \right) \otimes \left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i \right) \bar{\times}_l U_l^T \times_l U_l^T \right); (1:M)(1:M) \right]\!\right] \end{pmatrix} \end{aligned} \tag{16}$$

$$= \arg\max_{U_l|_{l=1}^{M}} \left(\begin{array}{l} \sum_{i=1}^{c} n_i \operatorname{tr}\left(U_l^T \left[\!\left[\left((\mathbf{M}_i - \mathbf{M}) \bar{\times}_l U_l^T \right) \otimes \left((\mathbf{M}_i - \mathbf{M}) \bar{\times}_l U_l^T \right); \left(\bar{l}\right)\left(\bar{l}\right) \right]\!\right] U_l \right) \\ -\zeta \sum_{i=1}^{c} \sum_{j=1}^{n_i} \operatorname{tr}\left(U_l^T \left[\!\left[\left((\mathbf{X}_{i;j} - \mathbf{M}_i) \bar{\times}_l U_l^T \right) \otimes \left((\mathbf{X}_{i;j} - \mathbf{M}_i) \bar{\times}_l U_l^T \right); \left(\bar{l}\right)\left(\bar{l}\right) \right]\!\right] U_l \right) \end{array} \right)$$

$$= \arg\max_{U_l|_{l=1}^{M}} \operatorname{tr} \left(U_l^T \left(\begin{array}{l} \sum_{i=1}^{c} \left[n_i \operatorname{mat}_l \left((\mathbf{M}_i - \mathbf{M}) \bar{\times}_l U_l^T \right) \operatorname{mat}_l^T \left((\mathbf{M}_i - \mathbf{M}) \bar{\times}_l U_l^T \right) \right] \\ -\zeta \sum_{i=1}^{c} \sum_{j=1}^{n_i} \left[\operatorname{mat}_l \left((\mathbf{X}_{i;j} - \mathbf{M}_i) \bar{\times}_l U_l^T \right) \operatorname{mat}_l^T \left((\mathbf{X}_{i;j} - \mathbf{M}_i) \bar{\times}_l U_l^T \right) \right] \end{array} \right) U_l \right).$$

To simplify (16), we define

$$B_l = \sum_{i=1}^{c} \left[n_i \operatorname{mat}_l \left((\mathbf{M}_i - \mathbf{M}) \bar{\times}_l U_l^T \right) \operatorname{mat}_l^T \left((\mathbf{M}_i - \mathbf{M}) \bar{\times}_l U_l^T \right) \right], \tag{17}$$

$$W_l = \sum_{i=1}^{c} \sum_{j=1}^{n_i} \left[\operatorname{mat}_l \left((\mathbf{X}_{i;j} - \mathbf{M}_i) \bar{\times}_l U_l^T \right) \operatorname{mat}_l^T \left((\mathbf{X}_{i;j} - \mathbf{M}_i) \bar{\times}_l U_l^T \right) \right]. \tag{18}$$

Therefore, (16) is simplified as,

$$U_l^* |_{l=1}^{M} = \arg\max_{U_l|_{l=1}^{M}} \operatorname{tr} \left(U_l^T \left(B_l - \zeta W_l \right) U_l \right). \tag{19}$$

As in Part A of this Section, ζ is a tuning parameter.

Table 1 lists the alternating projection optimization procedure for GTDA with the pre-defined tuning parameter ζ. At the end of this sub-Section, we describe how to determine ζ and the dimensionality of the output tensors $N_{1^*} \times N_{2^*} \times \cdots N_{M^*}$ automatically. The key steps in the alternating projection procedure are Steps 3-5, which involve finding the l^{th} projection matrix $U_l^{(t)}$ in the t^{th} iteration using $U_k^{(t-1)} |_{1 \le k \le M}^{k \ne l}$ found in the $(t-1)^{\text{th}}$ iteration. In Steps 3 and 4, we obtain the between-class scatter matrix $B_l^{(t-1)}$ and the within-class scatter matrix $W_l^{(t-1)}$ with the given $U_k^{(t-1)} |_{1 \le k \le M}^{k \ne l}$ in the $(t-1)^{\text{th}}$ iteration. The singular value decomposition (SVD) of $B_l^{(t-1)} - \zeta W_l^{(t-1)}$ is obtained and $U_l^{(t-1)}$ updated using the eigenvectors of $B_l^{(t-1)} - \zeta W_l^{(t-1)}$, which correspond to the largest eigenvalues of $B_l^{(t-1)} - \zeta W_l^{(t-1)}$. By iteratively conducting the steps 3-5 in Table 1, we obtain a solution $U_l |_{l=1}^{M} \in R^{N_{l^*} \times N_l}$ $N_{l^*} \le N_l$ constrained by $U_l^T U_l = I$.

Table 1. Alternating projection optimization procedure for GTDA.

Input: Training tensors $\mathbf{X}_{i;j}\,|_{1\le i\le c}^{1\le j\le n_i} \in R^{N_1\times N_2\times\ldots N_M}$, the dimensionality of the output tensors $\mathbf{Y}_{i;j} \in R^{N_{1^*}\times N_{2^*}\times\ldots N_{M^*}}$, the tuning parameters ζ, and the maximum number of training iterations T.

Output: The projection matrix $U_l\,|_{l=1}^{M} \in R^{N_{l^*}\times N_l}$ constrained by $U_l^T U_l = I$ and the output tensors $\mathbf{Y}_{i;j} \in R^{N_{1^*}\times N_{2^*}\times\ldots N_{M^*}}$.

Initialization: Set $U_l^{(0)}\,|_{l=1}^{M} = 1_{N_{l^*}\times N_l}$. (All entries of $U_l^{(0)}$ are 1.)

Step 1. For $t = 1$ to T {

Step 2. For $l = 1$ to M {

Step 3. Calculate $B_l^{(t-1)} = \sum_{i=1}^{c}\left[n_i \,\mathrm{mat}_l\left(\left(\mathbf{M}_i - \mathbf{M}\right)\overline{\times}_l \left(U_l^{(t-1)}\right)^T\right)\mathrm{mat}_l^T\left(\left(\mathbf{M}_i - \mathbf{M}\right)\overline{\times}_l \left(U_l^{(t-1)}\right)^T\right)\right]$;

Step 4. Calculate $W_l^{(t-1)} = \sum_{i=1}^{c}\sum_{j=1}^{n_i}\left[\mathrm{mat}_l\left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i\right)\overline{\times}_l \left(U_l^{(t-1)}\right)^T\right)\mathrm{mat}_l^T\left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i\right)\overline{\times}_l \left(U_l^{(t-1)}\right)^T\right)\right]$;

Step 5. Optimize $U^{*(t)}_l = \arg\max_{U} \mathrm{tr}\left(U^T\left(B_l^{(t-1)} - \zeta W_l^{(t-1)}\right)U\right)$ by SVD on $B_l^{(t-1)} - \zeta W_l^{(t-1)}$.

}//For loop in Step 2.

Check convergence: the training stage of GTDA converges if

Step 6. $\mathrm{Err}(t) = \sum_{l=1}^{M}\left\| U_l^{(t)}\left(U_l^{(t-1)}\right)^T - I\right\| \le \varepsilon$.

}// For loop in Step 1.

Step 7. $\mathbf{Y}_{i;j} = \mathbf{X}_{i;j}\prod_{l=1}^{M}\times_l U_l$.

In GTDA, we use the projected tensor $\mathbf{Y} = \mathbf{X}\prod_{l=1}^{M}\times_l U_l$ to replace the original general tensor $\mathbf{X}$ for recognition. Unlike 2DLDA [43], the alternating projection optimization procedure for GTDA converges, as proved in Theorem 1. Intuitively, this is because: 1) the alternating projection optimization procedure never decreases the function value $f\left(U_l\,|_{l=1}^{M}\right)$ (defined in Theorem 1 in (20)) of GTDA between two successive

iterations (that is the alternating projection optimization procedure for GTDA is a monotonic increasing procedure) and 2) the function value is lower and upper bounded by two limiting values.

Theorem 1: The alternating projection optimization procedure for GTDA converges.

Proof.

Let S_l be the set, which includes all possible U_l, i.e., $U_l \in S_l$, constrained by $U_l^T U_l = I$. We define a continuous function $f : S_1 \times S_2 \times \cdots \times S_M = \prod_{l=1}^{M} S_l \to R^+$,

$$\begin{aligned} f\left(U_l \mid_{l=1}^{M}\right) \doteq & \sum_{i=1}^{c} n_i \left[\!\!\left[\left(\left(\mathbf{M}_i - \mathbf{M}\right) \prod_{k=1}^{M} \times_k U_k^T \right) \otimes \left(\left(\mathbf{M}_i - \mathbf{M}\right) \prod_{k=1}^{M} \times_k U_k^T \right); \left(1:M\right)\left(1:M\right) \right]\!\!\right] \\ & - \zeta \sum_{i=1}^{c} \sum_{j=1}^{n_i} \left[\!\!\left[\left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i\right) \prod_{k=1}^{M} \times_k U_k^T \right) \otimes \left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i\right) \prod_{k=1}^{M} \times_k U_k^T \right); \left(1:M\right)\left(1:M\right) \right]\!\!\right]. \end{aligned} \tag{20}$$

The function $f\left(U_l \mid_{l=1}^{M}\right)$ is invariant to orthogonal transformation of U_l, i.e., if $Q_l Q_l^T = I$, we have $f\left(U_l \mid_{l=1}^{M}\right) = f_l\left(U_l Q_l \mid_{l=1}^{M}\right)$. According to (16), we can construct M different mappings based on f: $f_l\left(U_l\right) = f\left(U_l; U_d \mid_{d=1}^{l-1}, U_d \mid_{d=l+1}^{M}\right)$, $l \in \left\{1, 2, \cdots M\right\}$, where $f\left(U_l; U_d \mid_{d=1}^{l-1}, U_d \mid_{d=l+1}^{M}\right)$ means the function f varies with U_l with fixed $U_d \mid_{d=1}^{l-1}$ and $U_d \mid_{d=l+1}^{M}$. Based on these mappings, we define:

$$\begin{aligned} g_l\left(U_l\right) & \doteq \underset{U_l^T U_l = I, U_l \in S_l}{\arg\max}\ f_l\left(U_l\right) \\ & = \underset{U_l^T U_l = I, U_l \in S_l}{\arg\max}\ \mathrm{tr}\left(U_l^T \left(\begin{array}{l} \sum_{i=1}^{c} \left[n_i \, \mathrm{mat}_l\left(\left(\mathbf{M}_i - \mathbf{M}\right) \bar{\times}_l U_l^T \right) \mathrm{mat}_l^T\left(\left(\mathbf{M}_i - \mathbf{M}\right) \bar{\times}_l U_l^T \right) \right] \\ -\zeta \sum_{i=1}^{c} \sum_{j=1}^{n_i} \left[\mathrm{mat}_l\left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i\right) \bar{\times}_l U_l^T \right) \mathrm{mat}_l^T\left(\left(\mathbf{X}_{i;j} - \mathbf{M}_i\right) \bar{\times}_l U_l^T \right) \right] \end{array} \right) U_l \right). \end{aligned} \tag{21}$$

The mapping $g_l\left(U_l\right)$ is calculated by arg-maximizing $f_l\left(U_l^{(t)}\right)$ with the given $U_d \mid_{d=1}^{l-1}$ in the t^{th} iteration (i.e., $U_d^{(t)} \mid_{d=1}^{l-1}$) and $U_d \mid_{d=l+1}^{M}$ in the $(t-1)^{\text{th}}$ iteration (i.e., $U_d^{(t-1)} \mid_{d=l+1}^{M}$) of the for-loop in Steps 3–5 in Table 1, and $g_l\left(U_l\right) \in S_l$ is for all $l \in \left\{1, 2, \cdots M\right\}$.

Given randomly initialized $U_l^{(0)}|_{l=1}^M \in S_l$, the alternating projection generates a sequence of items $\left\{U_l^{(t)}|_{l=1}^M\right\}$ via $g_l(U_l)$ defined in (21). The corresponding function value $f_l\left(U_l^{(t)}\right)$ has the following relationship:

$$\begin{aligned} a = f_1\left(U_1^{(1)}\right) \le f_2\left(U_2^{(1)}\right) \le \cdots \le f_M\left(U_M^{(1)}\right) \le f_1\left(U_1^{(2)}\right) \le f_2\left(U_2^{(2)}\right) \le \\ \cdots \le f_1\left(U_1^{(t)}\right) \le f_2\left(U_2^{(t)}\right) \le \cdots \le f_1\left(U_1^{(T)}\right) \le f_2\left(U_2^{(T)}\right) \le \\ \cdots \le f_M\left(U_M^{(T)}\right) = b. \end{aligned} \tag{22}$$

where a and b are limiting values in the R^+ space and $T \to +\infty$. We have the relationship in (22), because $U_l^{(t)} = g_l\left(U_l^{(t)}\right) \doteq \underset{\left(U_l^{(t)}\right)^T U_l^{(t)} = I, U_l^{(t)} \in S_l}{\arg\max} f_l\left(U_l^{(t)}\right)$, i.e., the calculated $U_l^{(t)}$ maximizes $f\left(U_l^{(t)}; U_d^{(t)}|_{d=1}^{l-1}, U_d^{(t-1)}|_{d=l+1}^M\right)$ and the inequality $f\left(U_l^{(t)}; U_d^{(t)}|_{d=1}^{l-1}, U_d^{(t-1)}|_{d=l+1}^M\right) \ge f\left(U_{l-1}^{(t)}; U_d^{(t)}|_{d=1}^{l-2}, U_d^{(t-1)}|_{d=l}^M\right)$ holds, i.e., $f_l\left(U_l^{(t)}\right) \ge f_{l-1}\left(U_{l-1}^{(t)}\right)$.

Based on (21), the alternating projection optimization procedure can be illustrated by a composition of M sub-algorithms defined as

$$\Omega_l : \left(U_l|_{l=1}^M\right) \mapsto \prod_{d=1}^{l-1}{}_{\times_d} U_d \times g_l(U_l) \times \prod_{d=l+1}^{M}{}_{\times_d} U_d \,. \tag{23}$$

It follows that $\Omega \doteq \Omega_1 \circ \Omega_2 \circ \cdots \circ \Omega_M$ is a closed algorithm for compact sets $S_l|_{l=1}^M$. Based on (22), every sub-algorithm Ω_l increases the value of f, so Ω is monotonic with respect to f. Consequently, we can stop the alternating projection optimization procedure when the change of f between two successive iterations is small, i.e., the procedure halts when f achieves its extremum. Or equivalently, $\sum_{l=1}^M \left\| U_l^{(T)}\left(U_l^{(T-1)}\right)^T - I \right\| \le \varepsilon$ (the convergence check criterion in Step 6 in Table 1). ■

For practical applications, it is important to determine the tuning parameter ζ and the dimensions $N_{1*} \times N_{2*} \times \cdots N_{M*}$ of the output tensors automatically. In the t^{th} training iteration and the l^{th} order, we adjust

ζ between Step 4 and Step 5 by setting $\zeta^{(t)}$ equal to the maximum eigenvalue of $\left(W_l^{(t-1)}\right)^{-1} B_l^{(t-1)}$. In the t^{th} training iteration, the l^{th} dimension of the output tensors N_{l^*} is determined by the l^{th} projection matrix $U_l^{(t)}$, so we set a threshold value δ to automatically determine N_{l^*} according to the following inequality:

$$\frac{\lambda_{l;1}^{(t)}}{\sum_{j=1}^{N_l}\lambda_{l;j}^{(t)}} < \frac{\lambda_{l;1}^{(t)}+\lambda_{l;2}^{(t)}}{\sum_{j=1}^{N_l}\lambda_{l;j}^{(t)}} < \cdots < \frac{\sum_{i=1}^{N_{l^*}}\lambda_{l;i}^{(t)}}{\sum_{j=1}^{N_l}\lambda_{l;j}^{(t)}} \le \delta < \frac{\sum_{i=1}^{N_{l^*+1}}\lambda_{l;i}^{(t)}}{\sum_{j=1}^{N_l}\lambda_{l;j}^{(t)}} < \cdots < \frac{\sum_{i=1}^{N_l}\lambda_{l;i}^{(t)}}{\sum_{j=1}^{N_l}\lambda_{l;j}^{(t)}} = 1, \tag{24}$$

where $\lambda_{l;i}^{(t)}$ is the i^{th} eigenvalue of $B_l^{(t-1)} - \zeta^{(t)} W_l^{(t-1)}$ and $\lambda_{l;i}^{(t)} \ge \lambda_{l;j}^{(t)}$ if $i < j$. That is, N_{l^*} is the maximal index, which guarantees $\sum_{i=1}^{N_{l^*}}\lambda_{l;i}^{(t)} \le \delta \sum_{j=1}^{N_l}\lambda_{l;j}^{(t)}$. Or equivalently, N_{l^*+1} is the minimal index, which guarantees $\sum_{i=1}^{N_{l^*+1}}\lambda_{l;i}^{(t)} > \delta \sum_{j=1}^{N_l}\lambda_{l;j}^{(t)}$. Therefore, there is only one parameter, the threshold value δ, which affects the recognition performance. This is the only parameter which needs tuning for recognition tasks. Without this method, we would have to tune a total of $M+1$ parameters, comprising one parameter for each order of the M^{th} order tensors and one parameter for ζ in (19). In our experiments, the averaged gait images are 2^{nd} order tensors (one order for height and the other one for width); the averaged gait images with the Gabor based representation are 4^{th} order tensors (the first two orders for height and width, the third order for direction; and the fourth order for scale) the averaged gait images with the GaborS/GaborD based representation are 3^{rd} order tensors; and the averaged gait images with the GaborSD based representation are 2^{nd} order tensors.

C. The Working Principle of GTDA

If a tensor is scanned into a vector then it is hard to keep track of the information in spatial constraints. For example, two 4-neighbor connected pixels in an image may be separated hugely from each other after the vectorization.

To better characterize or classify natural data, the selected features should preserve as many as possible of the original constraints. When the training samples are limited, these constraints help to give reasonable solutions to classification problems, by reducing the number of unknown parameters. Take the strategies in

the Gaussian distribution estimation as an example[3]: when the training samples are few in number and located in a high dimensional space, then some constraints are imposed on the covariance matrix. A widely used constraint is to require that the covariance matrix be diagonal. Without such constraints, it is impossible to estimate a reasonable model with only a few training samples.

The tensor representation helps to reduce the number of parameters needed to model the data. For example, when a matrix X has the size $N_1 \times N_2$, we need to estimate the projection matrix U with the size $N_1 N_2 \times N_*$ for LDA (N_* is the number of selected features), but we only need to estimate the projection matrices U_1 with the size $N_1 \times N_{1*}$ and U_2 with the size $N_2 \times N_{2*}$ in GTDA. Furthermore, the estimation procedures for the first projection matrix U_1 and the second projection matrix U_2 are independent. The advantage of the independent estimation procedure is the number of the parameters in GTDA is much less than that of LDA.

When choosing between LDA and GTDA, we have the following two results:

1) when the number of the training samples is limited, the vectorization operation always leads to the under sample problem. That is, for a small training set, we need to use GTDA, because LDA will over–fit the data. The vectorization of a tensor into a vector makes it hard to keep track of the information in spatial constraints; and

2) when the number of the training samples is large, GTDA will under–fit the data. In this case, the vectorization operation for the data is helpful because it increases the number of parameters to model the data, i.e., LDA will be a suitable choice.

In summary, the testing error decreases with respect to the increasing number of the training samples. When the number of the training samples is limited, GTDA performs better than LDA. For specific applications, the *K*-fold cross validation [13] can be applied to determine which method is more suitable.

[3] Constraints in GTDA are justified by the form of the data. However, constraints in the example are ad hoc.

D. Complexity Analysis

The time complexity of LDA is $O\left(\left(\prod_{i=1}^{M} N_i\right)^3\right)$ in the training stage, when the samples $\mathbf{X}$ belong to $R^{N_1 \times N_2 \times \cdots \times N_M}$. The time complexity of the alternating projection method based optimization procedure of GTDA is $O\left(T\sum_{i=1}^{M} N_i^3\right)$, where T is the number of iterations to make the optimization procedure of GTDA converge. The space complexity of LDA is $O\left(\left(\prod_{i=1}^{M} N_i\right)^2\right)$ in the training stage. The space complexity of the alternating projection optimization procedure of GTDA is $O\left(\sum_{i=1}^{M} N_i^2\right)$.

The computational complexities of the alternating projection optimization procedure for GTDA with Gabor/GaborD/GaborS/GaborSD representation are listed in Table 2. In Table 2, T_1 (T_2, T_3, and T_4) is the number of iterations to make the optimization procedures of GTDA with Gabor (GaborD, GaborS, and GaborSD) based representations converge. In our experiments, we found that T_1, T_2, T_3, and T_4 are usually similar in value. Based on Table 2, the computational complexities of the alternating projection optimization procedure of GTDA for GaborS/GaborD/GaborSD based representations are reduced compared with that of the Gabor based representation.

Table 2. Computational complexities of the alternating projection method based optimization procedure of GTDA with Gabor/GaborD/GaborS/GaborSD representations.

	Time Complexity	Space Complexity
Gabor gaits in $R^{N_1 \times N_2 \times 5 \times 8}$	$O\left(T_1\left(637 + N_1^3 + N_2^3\right)\right)$	$O\left(89 + N_1^2 + N_2^2\right)$
GaborD gaits in $R^{N_1 \times N_2 \times 5}$	$O\left(T_2\left(125 + N_1^3 + N_2^3\right)\right)$	$O\left(25 + N_1^2 + N_2^2\right)$
GaborS gaits in $R^{N_1 \times N_2 \times 8}$	$O\left(T_3\left(512 + N_1^3 + N_2^3\right)\right)$	$O\left(64 + N_1^2 + N_2^2\right)$
GaborSD gaits in $R^{N_1 \times N_2}$	$O\left(T_4\left(N_1^3 + N_2^3\right)\right)$	$O\left(N_1^2 + N_2^2\right)$

VI. Experimental Results

This Section first briefly describes the USF HumanID gait database [34]. We then compare the performance of our algorithms with several other established algorithms for human gait recognition.

A. HumanID Gait Database: Gallery and Probe Data Sets

We carried out all of our experiments upon the USF HumanID outdoor gait (people–walking–sequence) database of version 2.1. The database was built for vision–based gait recognition, and it is widely used. It consists of 1,870 sequences from 122 subjects (people). For each of the subjects, there are the following covariates: change in viewpoint (Left or Right), change in shoe type (A or B), change in walking surface (Grass or Concrete), change in carrying condition (Briefcase or No Briefcase), and elapsed time (May or November) between sequences being compared. There is a set of 12 pre-designed experiments for algorithm comparisons. For algorithm training, the database provides a gallery collected in May with the following covariates: grass, shoe type A, right camera, and no briefcase. The gallery also includes a number of new subjects collected in November. This gallery dataset has 122 individuals. For algorithm testing, 12 probe sets are constructed according to the 12 experiments. Detailed information about the probe sets is given in Table 3. More detailed information about USF HumanID is described in [34].

Table 3. Twelve probe sets for challenge experiments.

Experiment (Probe)	# of Probe Sets	Difference between Gallery and Probe Set
A (G, A, L, NB, M/N)	122	View
B (G, B, R, NB, M/N)	54	Shoe
C (G, B, L, NB, M/N)	54	View and Shoe
D (C, A, R, NB, M/N)	121	Surface
E (C, B, R, NB, M/N)	60	Surface and Shoe
F (C, A, L, NB, M/N)	121	Surface and View
G (C, B, L, NB, M/N)	60	Surface, Shoe, and View
H (G, A, R, BF, M/N)	120	Briefcase
I (G, B, R, BF, M/N)	60	Briefcase and Shoe
J (G, A, L, BF, M/N)	120	Briefcase and View
K (G, A/B, R, NB, N)	33	Time, Shoe, and Clothing
L (C, A/B, R, NB, N)	33	Time, Shoe, Clothing, and Surface

Fig. 6 shows examples of the average gait images. The averaged gait image is the mean image (pixel by pixel) of silhouettes over a gait cycle within a sequence. A gait cycle is a series of stances: from full-stride-stance and heels-together-stance, to full-stride-stance. As suggested in [30], the whole sequence is partitioned into a series of sub–sequences according to the gait period length N_{Gait}. Then the binary images within one cycle (a sub–sequence) are averaged to acquire a set of average silhouette images AS_i, i.e. $AS_i \left|_{i=1}^{\lfloor T/N_{Gait} \rfloor}\right. = \left(\sum_{k=iN_{Gait}}^{k=(i+1)N_{Gait}-1} S(k) \right) \Big/ N_{Gait}$. The averaged gait image is robust against any errors in individual frames, so we choose the averaged gait image to represent a gait cycle, thus one sequence yields several averaged gait images and the number of averaged gait images depends on the number of gait cycles in this sequence. In the following experiments, averaged gait images are utilized as the original data for the gait recognition problem. Some further averaged gait images from the gallery set are also shown in Fig. 1, which demonstrates that the averaged gait images can be used for gait recognition, because different people have different averaged gait images.

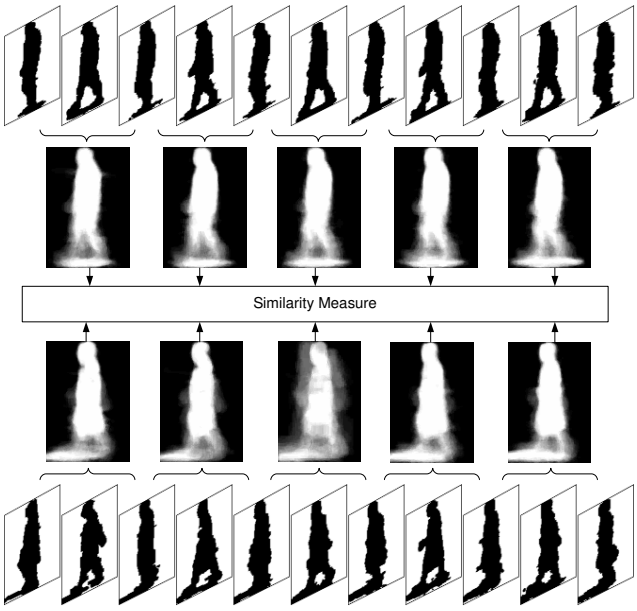

Fig. 6. The averaged gait extraction and the similarity measure.

The dissimilarity measure in gait recognition is the same as in [30]. The distance between the gallery sequence and the probe sequence is

$$\mathrm{Dist}\left(AS_P^{Method}, AS_G^{Method}\right) = \mathrm{Median}_{i=1}^{N_p}\left(\min_{j=1}^{N_G}\left\|AS_P^{Method}(i) - AS_G^{Method}(j)\right\|\right), \quad (22)$$

where $AS_P^{Method}(i)|_{i=1}^{N_P}$ is the i^{th} projected AS in the probe data and $AS_G^{Method}(j)|_{j=1}^{N_G}$ is the j^{th} projected AS in the gallery. The right hand side of (22) is the median of the Euclidean distances between the averaged silhouettes from the probe and the gallery. It is suggested as a suitable measure for gait recognition by Liu and Sarkar in [30].

There are only two parameters in all proposed methods, one for GTDA and one for LDA. In detail, one parameter is the threshold value σ for GTDA as described in (24). In all experiments, we vary σ from 0.85 to 0.995 with step 0.005. In 2DLDA, a similar strategy is used, i.e., σ is used to determine the dimensions of the projected subspace in each order. The other parameter is the number of the selected dimensions in LDA. In all experiments relevant to LDA, we vary the number of dimensions from 1 to 121 with step 1. To speed up all experiments, we down sample the original averaged gait images from 128×88 to 64×44 in all proposed methods. These are indicated by the note "H" in Tables 4 and 5. We also show some experimental results based on the original averaged gait images with the size 128×88.

To examination the effectiveness of the automatic selection of ζ in GTDA based recognition, we also manually tune the parameters to achieve further improvements by changing the selected dimensions for each mode and the tuning multiplier ζ defined in (15). This is indicated by the note "M" in Tables 4 and 5. Although manually tuning parameters improves the performance, it is time consuming.

B. Performance Evaluation

Sarkar et al. [34] evaluated the performance of the baseline algorithm on the HumanID challenge database using the rank one/five recognition rates: 1) the rank one recognition rate is the percentage of the number of the correct subjects in the first place of all retrieved subjects and 2) the rank five recognition rate is the percentage of the number of the correct subjects in any of the first five places of all retrieved subjects. Twelve experiments have been designed, namely experiment A to experiment L as shown in Table 3. The baseline algorithm reports the rank one recognition rates of the twelve experiments of increasing difficulty

from 78% as the easiest to 3% as the hardest by examining the effects of the introduced five covariates (under different combinations).

Table 4. Rank one recognition rate for human gait recognition.

Rank One	A	B	C	D	E	F	G	H	I	J	K	L	Avg
Probe Size	122	54	54	121	60	121	60	120	60	120	33	33	—
Baseline	73	78	48	32	22	17	17	61	57	36	3	3	40.9572
HMM	89	88	68	35	28	15	21	85	80	58	17	15	53.5365
IMED	75	83	65	25	28	19	16	58	60	42	2	9	42.8695
IMED+LDA	88	86	72	29	33	23	32	54	62	52	8	13	48.6357
LDA	87	85	76	31	30	18	21	63	59	54	3	6	48.1983
LDA+Sync	83	94	61	50	48	22	33	48	52	34	18	12	48.0355
LDA+Fusion	91	94	81	51	57	25	29	62	60	57	9	12	55.8257
2DLDA	89	93	80	28	33	17	19	74	71	49	16	16	50.9823
2DLDA+LDA	89	91	82	33	33	23	25	67	78	50	19	19	52.6409
GTDA (H)	85	88	73	24	25	15	14	53	49	45	4	7	42.9916
GTDA (M & H)	86	88	73	24	25	17	16	53	49	45	10	7	43.7035
GTDA	85	88	71	19	23	15	14	49	47	45	7	7	41.5992
Gabor+GTDA (H)	84	86	73	31	30	16	18	85	85	57	13	10	52.5052
GaborD+GTDA (H)	88	88	71	28	28	12	19	87	75	59	7	10	51.7359
GaborD+GTDA	81	88	65	21	23	8	13	92	83	55	13	10	49.2610
GaborS+GTDA (H)	89	89	69	31	33	13	16	79	76	56	13	13	51.4322
GaborS+GTDA	82	86	67	22	30	8	14	92	88	62	10	7	50.9990
GaborSD+GTDA (H)	87	89	71	23	28	8	14	82	69	51	4	13	48.2109
GaborSD+GTDA	81	82	69	17	26	7	14	91	78	60	101	10	48.8518
GTDA+LDA (H)	94	95	88	35	42	23	30	65	61	58	16	19	54.5543
GTDA+LDA	95	95	86	39	44	25	30	61	68	67	16	19	56.5167
Gabor+GTDA+LDA (H)	89	93	80	45	49	23	30	81	85	53	10	19	57.7296
GaborD+GTDA+LDA (H)	93	93	84	34	40	23	32	90	80	63	16	19	58.9102
GaborD+GTDA+LDA	89	93	84	27	35	17	26	93	88	67	16	22	57.5511
GaborS+GTDA+LDA (H)	93	95	88	39	47	28	33	82	82	63	19	19	60.2390
GaborS+GTDA+LDA	91	93	86	32	47	21	32	95	90	68	16	19	60.5804
GaborSD+GTDA+LDA (H)	92	93	78	30	38	21	26	82	75	55	16	19	54.8685

Tables 4 and 5 report all the experiments, which compare the proposed algorithms with the existing algorithms. The item “Avg” in Tables 4 and 5 means the averaged recognition rates of all probes (A-L), i.e., the ratio of correctly recognized subjects to the total number of subjects in all probes. The columns labeled A to L are exactly the same tasks as in the baseline algorithm. In both tables, the first rows give the performance of Baseline [34], HMM[19], IMED [41], IMED+LDA, LDA[16], LDA+Sync [16], LDA+Fusion [16], 2DLDA [43], and 2DLDA+LDA[43], respectively; while the performance of the new algorithms upon the same gallery set and probe set are fully reported on all the comparison experiments, which are namely, GTDA (H), GTDA (M & H), GTDA, Gabor+GTDA (H), GaborD+GTDA (H), GaborD+GTDA,

GaborS+GTDA (H), GaborS+GTDA, GaborSD+GTDA (H), GaborSD+GTDA, GTDA+LDA (H), GTDA+LDA, Gabor+GTDA+LDA (H), GaborD+GTDA+LDA (H), GaborD+GTDA+LDA, GaborS+GTDA+LDA (H), GaborS+GTDA+LDA, and GaborSD+GTDA+LDA (H), respectively. Finally, the last columns of both tables report the average performance of the corresponding algorithm on all the probe sets.

Table 5. Rank five recognition rate for human gait recognition.

Rank Five	A	B	C	D	E	F	G	H	I	J	K	L	Avg
Probe Size	122	54	54	121	60	121	60	120	60	120	33	33	—
Baseline	88	93	78	66	55	42	38	85	78	62	12	15	64.5397
HMM	—	—	—	—	—	—	—	—	—	—	—	—	—
IMED	91	93	83	52	59	41	38	86	76	76	12	15	65.3132
IMED+LDA	95	95	90	52	63	42	47	86	86	78	21	19	68.5950
LDA	92	93	89	58	60	36	43	90	81	79	12	12	67.3674
LDA+Sync	92	96	91	68	69	50	55	80	78	69	39	30	70.8528
LDA+Fusion	94	96	93	85	79	52	57	89	86	77	24	21	76.1754
2DLDA	97	93	93	57	59	39	47	91	94	75	37	34	70.9530
2DLDA+LDA	97	100	95	58	57	50	50	86	94	77	43	40	72.8507
GTDA (H)	98	95	95	57	54	34	42	75	80	69	22	16	65.0532
GTDA (M & H)	100	97	95	57	54	34	45	75	80	70	25	25	66.1472
GTDA	100	97	95	52	52	34	45	47	71	70	25	25	64.7015
Gabor+GTDA (H)	96	95	89	59	63	33	49	94	92	76	19	40	70.3205
GaborD+GTDA (H)	96	95	88	59	49	27	35	95	97	84	28	28	69.0898
GaborD+GTDA	96	91	82	45	45	23	32	96	94	78	31	37	65.4134
GaborS+GTDA (H)	98	97	93	60	52	34	37	93	95	79	31	25	70.0605
GaborS+GTDA	96	91	84	45	54	23	37	96	95	79	22	31	66.0741
GaborSD+GTDA (H)	95	93	88	54	47	27	30	89	88	71	28	28	64.8361
GaborSD+GTDA	96	91	82	43	54	23	33	98	94	82	28	34	66.3319
GTDA+LDA (H)	100	99	97	66	68	50	57	89	85	81	40	31	75.3267
GTDA+LDA	100	99	97	67	69	50	57	90	90	84	40	37	76.5365
Gabor+GTDA+LDA (H)	95	97	93	70	71	44	56	94	95	80	31	34	75.1451
GaborD+GTDA+LDA (H)	98	99	95	62	68	44	50	96	99	87	37	43	76.0731
GaborD+GTDA+LDA	98	99	93	52	59	37	49	99	99	88	34	43	73.5846
GaborS+GTDA+LDA (H)	98	99	97	68	68	50	56	95	99	84	40	40	77.5762
GaborS+GTDA+LDA	98	99	95	58	64	41	52	98	99	87	31	37	74.9008
GaborSD+GTDA+LDA (H)	99	99	93	57	61	40	47	89	90	78	40	37	71.6534

From the comparison results in Tables 4 and 5, it is clear that the averaged recognition rate of the twelve probes, our new methods (GTDA+LDA, Gabor+GTDA+LDA, GaborD+GTDA+LDA, and GaborS+GTDA +LDA) outperform the previous state-of-the-art algorithms (top part in both tables), e.g., the HMM algorithm, which is stable in modeling the gait cycles, and the IMED algorithm, which is demonstrated to improve the conventional LDA.

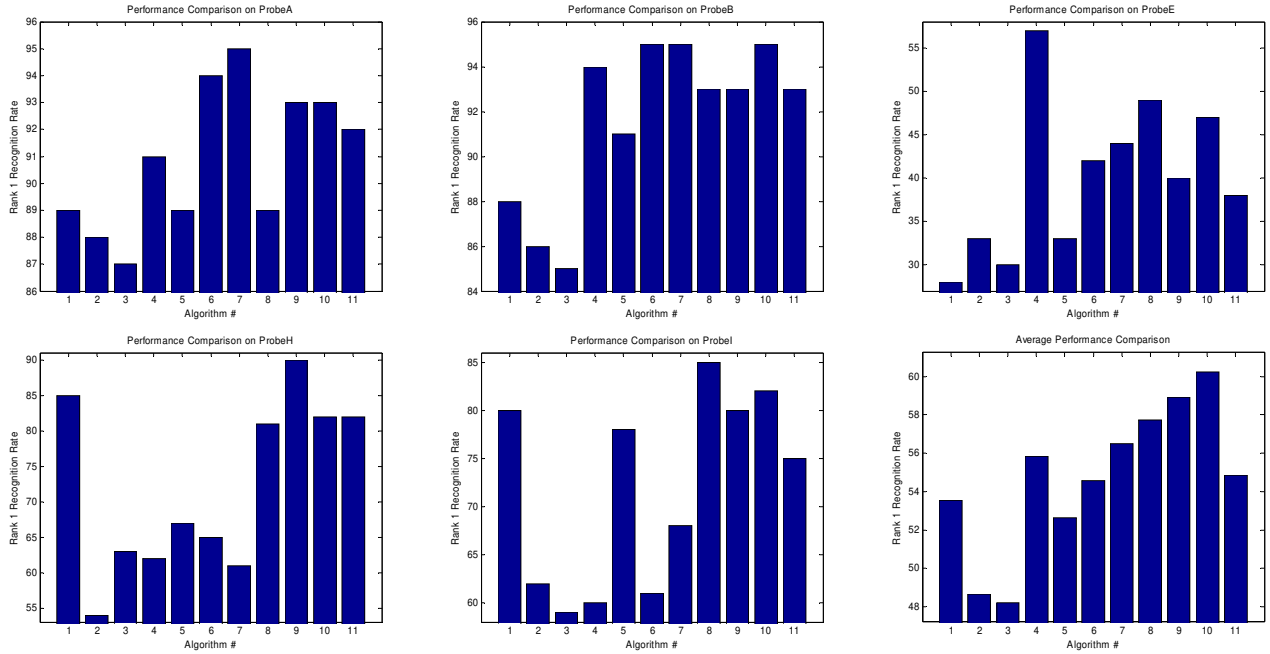

Fig. 7. Recognition performance comparison for rank one evaluation. From top-left to bottom-right, in each of the six subfigures (Probes A, B, E, H, and I and the average performance), there are eleven bars, which correspond to the performance of HMM, IMED+LDA, LDA, LDA+Fusion, 2DLDA+LDA, GTDA+LDA (H), GTDA+LDA, Gabor+GTDA+LDA(H), GaborD+GTDA+LDA(H), GaborS+GTDA+LDA(H), and GaborSD+GTDA+LDA(H), respectively.

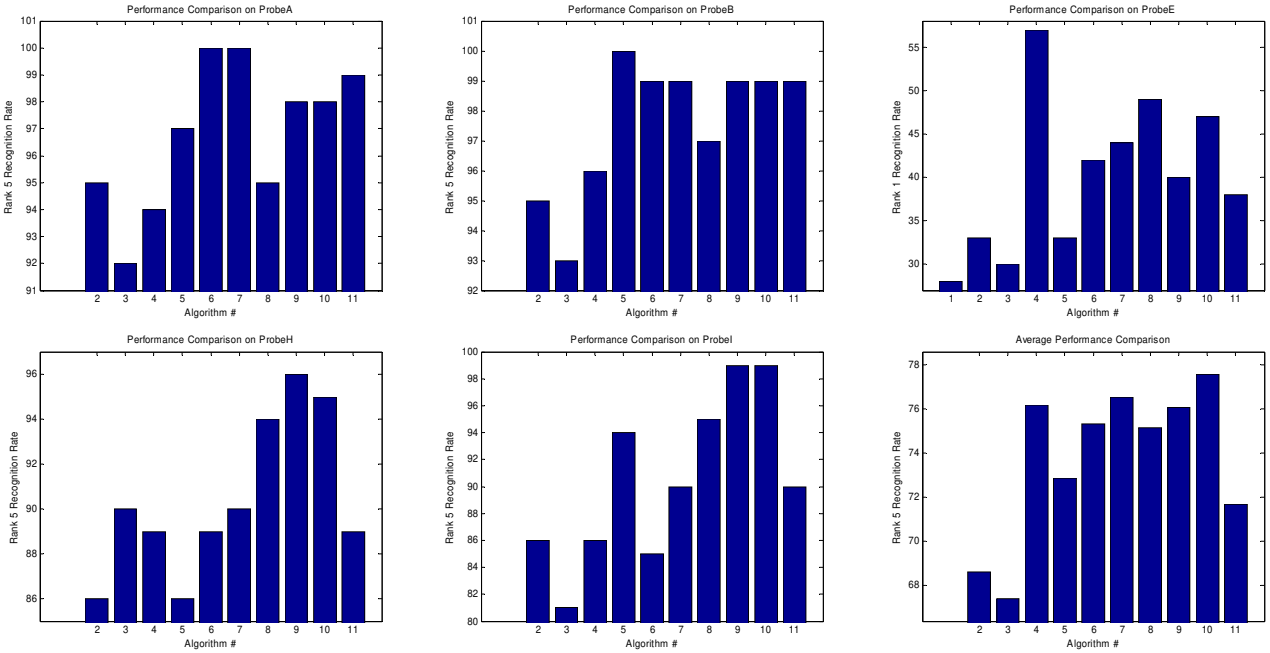

Fig. 8. Recognition performance comparison for rank five evaluation. From top-left to bottom-right, in each of the six subfigures (Probes A, B, E, H, and I and the average performance), there are ten bars, which correspond to the performance of IMED+LDA, LDA, LDA+Fusion, 2DLDA+LDA, GTDA+LDA(H), GTDA+LDA, Gabor+GTDA+LDA(H), GaborD+GTDA+LDA(H), GaborS+GTDA+LDA(H), and GaborSD+GTDA+LDA(H), respectively.

From Tables 4 and 5, we find our proposed methods are not very sensitive to the change of the size of the averaged gait images, because the recognition rates are slightly decreased when the averaged gait images are down sampled from 128×88 to 64×44. Manually tuning the parameter ζ in GTDA in (15) will slightly improve the averaged recognition rate. Furthermore, the performances for probes D-G and K-L are not satisfactory. Therefore, further studies are required to make them applicable. Finally, the performances of different methods have the following relationship: Baseline < IMED < LDA < IMED+ LDA < 2DLDA+LDA < LDA+Fusion < GTDA+LDA < GaborD+GTDA+LDA < Gabor+GTDA+LDA < GaborS+GTDA+LDA.

In addition, it is worth emphasizing that the effects of the covariates are also reflected in the experimental results. In general, in Tables 4 and 5, for the proposed GaborS/GaborD+GTDA+LDA, it shows that:

- *Viewpoint* and *shoe* changes have little impact on the recognition rate, this point is demonstrated by column A-C, in which the rank one recognition rates are around 92%;
- Apart from the *viewpoint* and the *shoe* covariates, if *briefcase* is also considered, the recognition tasks become more difficult and as a result, in columns H-J the performance is around 87%. The Gabor based representations are helpful to improve the recognition rates for these probes, because the difference around the briefcase region between a gallery sample and a probe (H-J) sample is significantly reduced, as shown in Fig. 5;
- Instead of the *briefcase* covariate, if the *viewpoint* and the *shoe* issues are studied together with the *surface* covariate, the recognition tasks become hard. This effect leads to the worse performance around 35% in columns D-G;
- The most difficult task is the *elapsed* time covariate. Much work should be done to improve the performance on the tasks K and L although our proposed algorithms also report better performance around 17% compared with previous efforts in most cases.

To further illustrate the advantage of the new algorithms, in Figs. 7 and 8, we compare the new algorithms with the state-of-the-art work. The bottom-right sub-figure reports average performance of HMM,

IMED+LDA, LDA, LDA+Fusion, 2DLDA+LDA, GTDA+LDA (H), GTDA+LDA, Gabor+GTDA+LDA (H), GaborD+GTDA+LDA (H), GaborS+GTDA+LDA (H), and GaborSD+GTDA+LDA (H). More detailed performance comparisons are in the other five sub-figures, for which the experiments were carried out on probes A, B, E, H, and I.

The state-of-the-art schemes for human gait recognition, such as HMM, IMED+LDA, and LDA+Fusion can also be utilized to enhance Gabor/GaborS/GaborD/GaborSD+GTDA+LDA based approaches: 1) HMM can be used to obtain more accurate cycles from a sequence. With accurate cycles, the recognition rate can be further improved; 2) IMED can be regarded as a kind of transformation to further improve GTDA data, especially as IMED has already been demonstrated to benefit LDA. Therefore, it is expected to improve GTDA; and 3) LDA+Fusion scheme benefits the gait recognition task from the data set point of view in terms of sample number and stability.

C. Convergence Examination

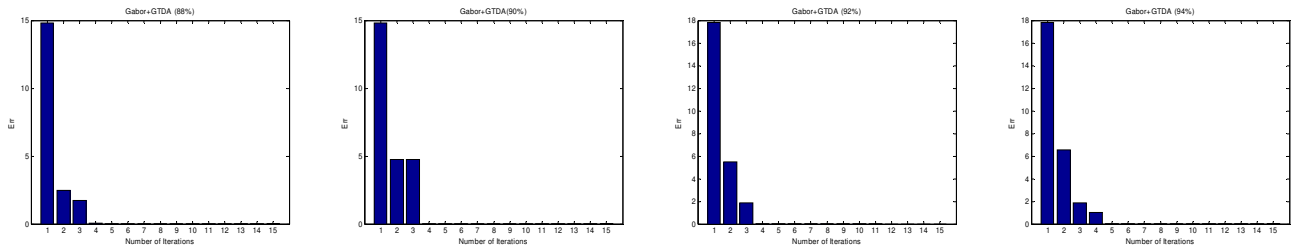

Fig. 9. Experimental based convergence justification for the alternating projection method for GTDA. The x-coordinate is the number of the training iterations and the y-coordinate is the Err value defined in Step 6 in Table 1. From left to right, these four sub-figures show how Err changes with the increasing number of training iterations in different threshold values (88%, 90% 92% and 94%) defined in (24) with predefined ζ.

From Fig. 9, it can be seen that only 3 to 5 iterations are usually required for the convergence of the alternating projection method based optimization procedure of GTDA with predefined ζ, because the Err values approach zero rapidly. In contrast, the traditional 2DLDA does not converge during the training procedure, which can be seen from the first figure in [43].

VII. Conclusion

A human being's walking manner or gait can reflect the walker's physical characteristics and psychological state, and therefore the features of gait can be employed for individual recognition. This paper focuses on the representation and pre-processing of appearance-based models for human gait sequences. Two major novel representation models are presented, namely, Gabor gait and tensor gait, and some extensions of them are made to further enhance their abilities for recognition tasks. Gabor gait is based on the well-known Gabor functions, which have been demonstrated to benefit visual information processing and recognition in general. In this paper, three different approaches using Gabor functions are developed to reduce the computational complexities in calculating the representation, in training classifiers, and in testing. They are the sum of Gabor functions over directions for gait representation (GaborD), the sum of Gabor functions over scales for gait representation (GaborS), and the sum of Gabor functions over both directions and scales for gait representation (GaborSD). Tensor gait is also introduced to represent these Gabor gaits. To further take the feature selection into account, the size of the tensor gait is reduced by the general tensor discriminant analysis (GTDA), which is based on a low rank approximation of the original data. Apart from preserving discriminative information, GTDA has another advantage - it significantly reduces the effects of under sampling on classification. In contrast with previous work on tensor discriminative analysis, the alternating projection optimization procedure of GTDA converges. The developed Gabor gait methods and GTDA methods are combined with LDA for gait recognition. Experiments show that the new algorithms achieve better recognition rates than previous algorithms reported in the literature.

Acknowledgement

The authors would like to thank the associate editor Dr. Trevor Darrell and the anonymous reviewers for their constructive comments on two earlier versions of this manuscript.

REFERENCES

[1] P. Belhumeur and J. Hespanha and D. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, 1997.

[2] C. BenAbdelkader and L. S. Davis, "Detection of People Carrying Objects: a Motion-based Recognition Approach," *Proc. IEEE Int'l Conf. Automatic Face and Gesture Recognition*, pp. 378-383, 2002.

[3] A. Bobick and A. Johnson, "Gait Recognition using Static Activity–specific Parameters," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 423–430, Kauai, HI, 2001.

[4] J. E. Boyd, "Synchronization of Oscillations for Machine Perception of Gaits," *Computer Vision and Image Understanding*, vol. 96, no. 1, pp. 35–59, 2004.

[5] L. F. Chen, H.Y. Liao, M. T. Ko, J. C. Lin, and G. J. Yu, "A New LDA–based Face Recognition System which Can Solve the Small Sample Size Problem," *Pattern Recognition*, vol. 33, no. 10, pp. 1,713–1,726, 2000.

[6] R. T. Collins, R. Bross, and J. Shi, "Silhouette–based Human Identification from Body Shape and Gait," *Proc. IEEE Int'l Conf. Automatic Face and Gesture Recognition*, pp. 351–356, Washington DC, 2002.

[7] D. Cunado, M. Nixon, and J. Carter, "Automatic Extraction and Description of Human Gait Models for Recognition Purposes," *Computer Vision and Image Understanding*, vol. 90, no. 1, pp. 1–41, 2003.

[8] R. Cutler and L. Davis, "Robust Periodic Motion and Motion Symmetry Detection," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pp. 615–622, Hilton Head, SC, 2000.

[9] J. Cutting and D. Proffitt, "Gait Perception as an Example of How We May Perceive Events," *Intersensory Perception and Sensory Integration*, vol. 2, pp. 249–273, New York, 1981.

[10] J. G. Daugman, "Two–Dimensional Spectral Analysis of Cortical Receptive Field Profile," *Vision Research*, vol. 20, pp. 847–856, 1980.

[11] J. G. Daugman, "Uncertainty Relation for Resolution in Space, Spatial Frequency, and Orientation Optimized by Two–Dimensional Visual Cortical Filters," *Journal of the Optical Society of America*, vol. 2, no. 7, pp. 1,160–1,169, 1985.

[12] J. W. Davis and A. F. Bobick, "The Representation and Recognition of Human Movement using Temporal Templates," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 928–934, San Juan, Puerto Rico, 1997.

[13] R. O. Duda, P. E. Hart, and D. G. Stork, Pattern Classification (2nd). Wiley-Interscience, 2000.

[14] D. Dunn, W. E. Higgins, J. Wakeley, "Texture Segmentation Using 2–D Gabor Elementary Functions," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 2, pp. 130–149, 1994.

[15] K. Fukunaga. Introduction to Statistical Pattern Recognition (2nd). Academic Press, Boston 1990.

[16] J. Han and B. Bhanu, "Statistical Feature Fusion for Gait–Based Human Recognition," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 842–847, Washington, DC, 2004.

[17] I. Haritaoglu, R. Cutler, D. Harwood, and L. Davis, "Backpack: Detection of people carrying objects using silhouettes," *Computer Vision and Image Understanding*, vol. 6, no. 3, pp. 385-397, 2001.

[18] G. Johansson, "Visual Motion Perception," Scientific American, vol. 232, pp. 76–88, 1975.

[19] A. Kale, A. Sundaresan, A. N. Rajagopalan, N. P. Cuntoor, A. K. Roy–Chowdhury, V. Kruger, and R. Chellappa, "Identification of Humans using Gait," *IEEE Trans. Image Processing*, vol. 13, no. 9, pp. 1,163–1,173, 2004.

[20] L. D. Lathauwer, Signal Processing Based on Multilinear Algebra, Ph.D. Thesis, Katholike Universiteit Leuven, 1997.

[21] L. Lee, G. Dalley, and K. Tieu, "Learning Pedestrian Models for Silhouette Refinement," *Proc. IEEE Int'l Conf. Computer Vision*, vol. 1, pp. 663–670, Nice, France, 2003.

[22] L. Lee and W. E. L. Grimson, "Gait Analysis for Recognition and Classification," *Proc. IEEE Int'l Conf. Automatic Face and Gesture Recognition*, pp. 155–162, Washington, DC, 2002.

[23] T. S. Lee, "Image Representation Using 2D Gabor Wavelets," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, pp. 959–971, 2003.

[24] J. J. Little and J. E. Boyd, "Recognizing People by Their Gait: the Shape of Motion," *Videre*, vol. 1, no. 2, pp. 1–32, 1998.

[25] C. Liu, "Gabor–based Kernel PCA with Fractional Power Polynomial Models for Face Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 572–581, 2004.

[26] C. Liu and H. Wechsler, "Enhanced Fisher Linear Discriminant Models for Face Recognition," *Proc. IEEE Int'l Conf. Pattern Recognition*, vol.2, pp. 1,368–1,372, Brisbane, Australia, 1998.

[27] C. Liu and H. Wechsler, "Gabor Feature Based Classification Using the Enhanced Fisher Linear Discriminant Model for Face Recognition," *IEEE Trans. Image Processing*, vol. 11, no. 4, pp. 467–476, 2002.

[28] F. Liu, and R. W. Picard, "Finding Periodicity in Space and Time," *Proc. IEEE Int'l Conf. Computer Vision*, pp. 376–383, Bombay, India, 1998.

[29] Y. Liu, R.T. Collins, and Y. Tsin, "Gait Sequence Analysis using Frieze Patterns," *Proc. of Europe Conf. Computer Vision*, vol. 2, pp. 657-671, 2002.

[30] Z. Liu and S. Sarkar, "Simplest Representation yet for Gait Recognition: Averaged Silhouette," *Proc. IEEE Int'l Conf. Pattern Recognition*, vol. 4, pp. 211-214, 2004.

[31] S. Marcelja, "Mathematical Description of the Responses of Simple Cortical Cells," *Journal of the Optical Society of America*, vol. 70, no. 11, pp. 1297–1300, 1980.

[32] T. Moeslund and E. Granum, "A Survey of Computer Vision–based Human Motion Capture," *Computer Vision and Image Understanding*, vol. 81, no. 3, pp. 231–268, 2001.

[33] M. Murray, A. Drought, and R. Kory, "Walking Pattern of Normal Men," *Journal of Bone and Joint Surgery*, vol. 46–A, no. 2, pp. 335–360, 1964.

[34] S. Sarkar, P. Phillips, Z. Liu, I. Vega, P. Grother, and K. Bowyer, "The HumanID Gait Challenge Problem: Data Sets, Performance, and Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 162–177, 2005.

[35] D. L. Swets and J. Weng, "Using Discriminant Eigenfeatures for Image Retrieval," *IEEE Trans. Pattern Analysis and Machine Intelligences*, vol. 18, no. 8, pp. 831-836, 1996.

[36] R. Tanawongsuwan and A. Bobick, "Gait Recognition from Time–normalized Joint–angle Trajectories in the Walking Plane," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 726–731, Kauai HI, 2001.

[37] R. Tanawongsuwan and A. Bobick, "Modelling the Effects of Walking Speed on Appearance–based Gait Recognition," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 783–790, Washington, DC, 2004.

[38] D. Tao, X. Li, X. Wu, and S. J. Maybank, "Human Carrying Status in Visual Surveillance," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol.2, pp. 1,670–1,677, NY, 2006.

[39] M. A. O. Vasilescu, D. Terzopoulos, "Multilinear Subspace Analysis for Image Ensembles," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol.2, pp. 93–99, Madison, WI, 2003.

[40] L. Wang, T. Tan, H. Ning, and W. Hu, "Silhouette Analysis–based Gait Recognition for Human Identification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1,505–1,518, 2003.

[41] L. Wang, Y. Zhang, J. Feng, "On the Euclidean Distance of Images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1,334–1,339, 2005.

[42] X. Wang and X. Tang, "Dual-Space Linear Discriminant Analysis for Face Recognition," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 564–569, Washington, DC, 2004.

[43] J. Ye, R. Janardan, and Q. Li, "Two–Dimensional Linear Discriminant Analysis," *Neural Information Processing Systems*, pp. 1,569–1,576, Vancouver, Canada, 2005.

[44] H. Yu and J. Yang, "A Direct LDA Algorithm for High–dimensional Data with Application to Face Recognition," *Pattern Recognition*, vol. 34, no. 12, pp. 2,067–2,070, 2001.