Learning and Matching of Dynamic Shape Manifolds for Human Action Recognition

Liang Wang and David Suter

***Abstract*—In this paper, we learn explicit representations for dynamic shape manifolds of moving humans for the task of action recognition. We exploit locality preserving projections (LPP) for dimensionality reduction, leading to a low-dimensional embedding of human movements. Given a sequence of moving silhouettes associated to an action video, by LPP, we project them into a low-dimensional space to characterize the spatiotemporal property of the action, as well as to preserve much of the geometric structure. To match the embedded action trajectories, the median Hausdorff distance or normalized spatiotemporal correlation is used for similarity measures. Action classification is then achieved in a nearest-neighbor framework. To evaluate the proposed method, extensive experiments have been carried out on a recent dataset including ten actions performed by nine different subjects. The experimental results show that the proposed method is able to not only recognize human actions effectively, but also considerably tolerate some challenging conditions, e.g., partial occlusion, low-quality videos, changes in viewpoints, scales, and clothes; within-class variations caused by different subjects with different physical build; styles of motion; etc.**

***Index Terms*—Action recognition, dimensionality reduction, human motion analysis, locality preserving projections (LPP).**

I. Introduction

VISUAL analysis of human movements concerns the detection, tracking and recognition of people, and, more generally, the understanding of human activities, from image sequences [3]. In particular, the recognition of human activities has a wide range of promising applications such as smart surveillance, perceptual interfaces, interpretation of sport events, etc. Although there has been much work on human motion analysis over the past two decades (see reviews [1]–[3]), activity understanding still remains challenging. In terms of higher-level analysis, previous studies generally fall under two major categories of approaches, i.e., template matching based approaches and state-space approaches [3]. The former usually characterizes the spatiotemporal distribution generated by the motion in its continuum, e.g., Bobick and Davis [24] proposed temporal templates for the representation and recognition of aerobics actions. The latter generally defines each static posture in the action as a state, and recognizes it through considering temporal variations of those poses using state-space models, e.g., Yamato *et al.* [30] combined the mesh features of 2-D human blobs with the HMMs to identify tennis behaviors. Our approach is a form of matching and is, thus, closer to the first category.

An important question in action recognition is how to extract and represent useful information from raw video data. There have been various approaches to extract features, e.g., key frame extraction [25], the computation of optical flow [4]–[6], space-time gradients and local descriptors [19], [21], [22], feature tracking [7]–[18], etc. However, the key frame lacks motion information, and the usage of exemplar poses becomes impractical as the number of activities increases. The studies based on computing local gradients or intensity-based features can be unreliable in cases of low-quality video, smooth surfaces, motion discontinuities and singularities. Feature tracking is also complex due to the large variability in the articulation of the human body, fast motions, self-occlusions, changes of appearance, etc.

Shape and kinematics are two important cues in human movement analysis [26]. It is difficult to accurately extract kinematics from real videos using current imperfect vision techniques. Alternatively, focusing on shape, human action can be regarded as a temporal process in which human silhouettes continuously change over time. If the extracted feature in each frame characterizes the human silhouette, temporal variations of these features will implicitly characterize motion kinematics. Recently, there has been some work showing that human silhouette plays a considerable role in the activity understanding [23]–[29]. It is an interesting idea to use purely binary silhouette (or shape) information without any explicit body models for action recognition. We pursue this idea in this paper.

A. Motivation and Overview of Approach

Our study aims to establish an effective action recognition method using analysis of spatiotemporal silhouettes measured during the activities, based on the idea that spatiotemporal variations of human silhouettes encode not only spatial information about body poses at certain instants, but also dynamic information about global body motion and the motions of local body parts. It appears to be feasible to use features that can be obtained from space-time shapes for exploring the action properties. In contrast to feature tracking, extracting space-time shapes

Manuscript received May 11, 2006; revised January 29, 2007. This work was supported by the ASC Centre for Perceptive and Intelligent Machines in Complex Environments (PIMCE). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Manuel Samuelides.

L. Wang was with Monash University, Melbourne, Victoria, 3800, Australia. He is now with the Department of Computer Science and Software Engineering, The University of Melbourne, Melbourne, 3010, Australia (e-mail: lwwang@csse.unimelb.edu.au).

D. Suter is with the Department of Electrical and Computer Systems Engineering, Monash University, Melbourne, Victoria, 3010, Australia (e-mail: d.suter@eng.monash.edu.au).

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

Digital Object Identifier 10.1109/TIP.2007.896661

is also easier to implement using current vision technologies, especially in the case of stationary cameras.

The deformations of the human silhouette during an activity are subject to certain physical and temporal constraints. The silhouettes can be regarded as points in a high-dimensional visual space, and these points may generally be expected to lie on a low-dimensional manifold embedded in the high-dimensional image space. This motivates the analysis of human actions in a low-dimensional subspace rather than the ambient space. In addition to principal component analysis (PCA) [34] and linear discriminant analysis (LDA) [35], some newer frameworks for dimensionality reduction have been introduced, e.g., isometric feature mapping (Isomap) [37], local linear embedding (LLE) [36], locality preserving projections (LPP) [39], etc. There are many impressive results concerning how to discover the intrinsic features of the data manifold, but there have been relatively fewer reports on practical applications of manifold learning for complex human action recognition.

Based on the above considerations, this paper explores the applicability of analysis of dynamic shapes of moving objects for action recognition. To characterize the properties of human actions in a more compact manner, the associated sequences of dynamic silhouettes are used to learn the activity space using LPP [39]. To match activity trajectories in the low-dimensional embedding space, two kinds of motion similarity measures are used. Finally, supervised pattern classification techniques are applied for action recognition in a nearest-neighbor framework. Although the method is simple in essence, the experimental results are encouraging.

B. *Contributions*

The main contributions of this paper are summarized as follows.

- We develop a simple but effective method for human action recognition using dynamic shape analysis. The proposed method does not directly analyze the dynamics of motions, but derives a compact trajectory description to reflect the characteristics of motion patterns.
- Performance evaluation is carried out on a recently reported database [27], with size similar or larger to those of most action databases currently in use, in terms of the number of actions, subjects and videos, and good results with considerable robustness are obtained, which demonstrates that silhouettes are indeed informative for characterization and recognition of human motions.
- Our experimental results show that LPP can discover the intrinsic structure of action manifolds and can, thus, provide more compact feature representations. This extends to video-based human action recognition, the relative success of LPP demonstrated over PCA and LDA in image-based face recognition [43].
- We provide the quantitative and qualitative comparative experiments to examine the proposed method, as well as comparison of different dimensionality reduction methods. In contrast, a large number of papers in the literature only reported the recognition results on individual limited-size databases, but they seldom made informed comparisons among different algorithms.
- The proposed method has several desirable properties: a) it is easier to comprehend and implement, without the requirements of explicit feature tracking and complex probabilistic modeling of motion patterns; b) being based on binary silhouette analysis, it naturally avoids some problems arising in most previous methods, e.g., unreliable 2-D or 3D tracking, expensive and sensitive optical flow computation, etc; and c) it obtains good results on a large and challenging database and exhibits considerable robustness.

C. *Organization*

Section II reviews related work on action recognition and the dimension reduction methods. Section III details visual inputs and the method of activity subspace learning. Section IV describes action recognition. A large number of experimental and comparative results are presented and discussed in Section V, prior to a summary in Section VI.

II. Related Work

A. *Human Action Recognition*

There has been much work on human motion/action recognition in the recent literature [4]–[33]. We will briefly review those methods in order to put ours into context. For clarity, we roughly divide the existing work into three major categories, based on the used low-level feature cues, as follows:

1) Feature Tracking Based Methods: Many traditional methods of activity recognition are based on feature tracking in either 2-D or 3-D space [1], [2]. The earliest study on feature tracking for motion perception is probably due to Johansson's experiments with moving light displays (MLD) [31]. Song *et al.* [20] used spatial arrangements of the tracked points to distinguish between walking and biking. Rao and Shah [14] used the trajectory of a tracked hand to differentiate between actions such as opening a cabinet or picking up an object. In [18], an action was represented by 40 curves derived from the tracking results of five body parts using a cardboard people model. In addition to 2-D features, some approaches used 3-D information to establish motion descriptors based on positions, angles and velocities of body parts [7]–[9], [12], [13], though accurate 3-D tracking is quite difficult for unrestricted activities. For example, Ali and Aggarwal [12] used the angles of inclination of the torso, the lower and upper parts of legs as features to recognize activity.

2) Intensity or Gradient Based Methods: Some researchers used intensity or gradient-based features for motion recognition. Zelnik-Manor and Irani [21] used marginal histograms of spatiotemporal gradients at multiple temporal scales to cluster video events. The work by Polana and Nelson [6] used the normal flow for periodic and nonrigid motion recognition. Efros *et al.* [5] proposed a descriptor based on blurred optical flow measurements, and applied it to recognize actions on ballet, tennis and football datasets. There has also been significant interest in approaches that exploit local descriptors on interest points in static images or videos. Schuldt *et al.* [22] constructed video representations in terms of local space-time features and integrated such representations with a SVM classification scheme for action recognition. Dollar *et al.* [19]

proposed to characterize behaviors through spatiotemporal feature points, in which a behavior was described in terms of the types and locations of feature points present. However, these studies usually rely on the assumption that one can reliably detect a sufficient number of stable interest points in videos.

3) Silhouette-Based Methods: Since human actions can be characterized as motion of a sequence of human silhouettes over time, silhouette-based methods are becoming popular [23]–[29]. Kellokumpu *et al.* [23] proposed a human activity recognition method from sequences of postures. A SVM was used for posture classification and then the discrete HMMs were used for activity recognition. In [28], Sminchisescu *et al.* recognized human motions based on discriminative conditional random field (CRF) and maximum entropy Markov models (MEMM), using image descriptors combining shape context and pairwise edge features extracted on the silhouette. Blank *et al.* [27] utilized properties of the solution to the Poisson equation to extract features from the space-time shapes, e.g., local space-time saliency, action dynamics, shape structure and orientation. They showed that these features were useful for action recognition, detection and clustering.

As stated earlier, feature tracking is complex due to the large variability in the shape and articulation of the human body. In particular, perfect limb tracking is not yet well solved. When using image measurements in terms of optical flow, gradients or intensity-based features, the recognition results depend greatly on the recording conditions. In contrast, human silhouette extraction from videos is easy for current vision techniques, especially in the imaging setting with fixed cameras. So, the method that we present here directly relies on moving silhouettes.

B. Methods of Dimensionality Reduction

In many areas, such as artificial intelligence and information processing, one is often confronted with intrinsically low-dimensional data lying in a very high-dimensional space. Accordingly, much work on dimensionality reduction has been proposed to solve the problem of "curse of dimensionality." Both linear PCA and LDA usually fail to discover the underlying structure if the observed images lie on a nonlinear manifold hidden in the high-dimensional image space. Alternatively, some nonlinear techniques have been carried out. As a global embedding algorithm, Isomap [37] presumes that isometric properties should be preserved in both the observation space and the intrinsic embedding space in the affine sense. LLE [36] and Laplacian eigenmaps (LE) [38] focus on the preservation of local neighbor structure. Recently, He and Niyogi proposed LPP [39], based on linear projective maps that arise by solving a variational problem that optimally preserves the neighborhood structure of the data set. Despite being linear, LPP shares many of the data representation properties of nonlinear techniques. Yet it is computationally more tractable, and more crucially, is defined everywhere rather than just on the training data points. This paper will, thus, use LPP to obtain low-dimensional embedding of dynamic shapes.

A few researchers have recently explored nonlinear dimensionality reduction methods for different vision tasks such as 3-D pose recovery [40], [42], face and expression recognition [43]–[45] and visual tracking [41], [42]. However, research on the manifold learning of human activities for recognition is still limited. A work closely related to this paper is that of [32], in which PCA was used to obtain low-dimensional action representations. Our method is different from that work in a few major aspects: a) *Visual input*. The response of the infinite impulse response (IIR) filter on original frames is used to construct the feature images in [32]. Instead, we just use binary silhouettes. These are easier to extract and less sensitive to the low color contrast and texture changes of clothes. b) *Embedding method*. In the case of articulated objects, action measurements are inherently nonlinear across the whole action. Linear PCA will not be able to effectively discover the underlying structure of actions. Although LPP is linear, it is of particular applicability in the special case where the observed data is a nonlinear manifold embedded in the ambient space [39]. c) *Similarity measures*. In addition to the median Hausdorff distance, we use the normalized spatiotemporal correlation for measuring the motion similarity, which counters their claim that frame-to-frame correlation appears to be not efficient for matching action trajectories. d) *Robustness test*. No experiments are provided in [32] to explain the methods' robustness. We compare the method of [32] with ours on the same dataset, and demonstrate the better robustness of LPP than PCA on some walking sequences with various challenging conditions.

III. Activity Manifold Learning

It is a formidable task to learn the complete structure of the activity in the high-dimensional image space. Our idea is to embed the human actions into a lower dimensional feature space. We choose LPP for this goal based on the following considerations: a) LPP explicitly models the manifold structure by an adjacency graph, which provides an efficient subspace learning algorithm to discover the intrinsic structure of the action space; b) LPP shares some of the data representation properties of nonlinear techniques such as LLE, e.g., locality preserving characteristic; c) LPP is obtained by finding the optimal linear approximations to eigenfunctions of the Laplace Beltrami operator [39]. This linearity naturally leads to low computation complexity and is, thus, more efficient for practical applications; and d) although a few nonlinear methods (e.g., Isomap, LLE) do yield impressive results on some benchmark artificial datasets, they are defined only on the training data points and how to evaluate the maps on new test data remains unclear [43]. In contrast, LPP may be simply applied to any new data point.

A. Brief Introduction to LPP

The problem of linear dimensionality reduction can be generally formalised as follows. Given a set of high-dimensional data points $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ in $\mathbf{R}^h$, find a transformation matrix $\boldsymbol{E}$, so that these n data points can be represented by a set of low-dimensional points $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n$ in $\mathbf{R}^l$ $(l \ll h)$, where $\boldsymbol{y}_i = \boldsymbol{E}^T \boldsymbol{x}_i$.

Based on [39], the algorithmic procedure of LPP is simply summarized as follows:

1) *Constructing the adjacency graph*: Let G denote a graph with n nodes. An edge will be put between the nodes i and j if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are "close," where "close" can be defined by ε-neighbourhoods, $\boldsymbol{\varepsilon} \in \mathbf{R}$ (i.e., $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 < \varepsilon$ in

$\mathbf{R}^h$, where ε defines the radius of the local neighborhood), or K-nearest neighbours, $K \in \mathbf{N}$ (i.e., $\boldsymbol{x}_i$ is among the K-nearest neighbours of $\boldsymbol{x}_j$ or $\boldsymbol{x}_j$ is among the K-nearest neighbours of $\boldsymbol{x}_i$).

2) *Choosing the weights*: The weights evaluate the local structure of the data space. $\boldsymbol{W}$ is a sparse and symmetric $n \times n$ matrix with the weight w_{ij} of the edge joining the nodes i and j, and 0 if there is no such edge. As well, two variations for weighting the edges can be used, i.e., heat kernel, $w_{\boldsymbol{ij}} = e^{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/\boldsymbol{t}}$, $t \in \mathbf{R}$, or more simply, $w_{ij} = 1$ if and only if the vertices i and j are connected.

3) *Eigenmaps*: The optimal projection preserving the locality can be solved by minimizing the following objective function based on the standard spectral graph theory [39]

$$\min \sum_{i,j} (\boldsymbol{y}_i - \boldsymbol{y}_j)^2 w_{ij}. \tag{1}$$

This minimization problem is to ensure that if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are close, then $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ are close as well. Let $\boldsymbol{e}$ denote a transformation vector. The objective function can be modified [39]

$$\begin{aligned}\frac{1}{2}\sum_{\boldsymbol{ij}}(\boldsymbol{y}_i-\boldsymbol{y}_j)^2 w_{\boldsymbol{ij}} &= \frac{1}{2}\sum_{ij}\left(\boldsymbol{e}^T\boldsymbol{x}_i-\boldsymbol{e}^T\boldsymbol{x}_j\right)^2 w_{\boldsymbol{ij}}\\ &= \sum_i \boldsymbol{e}^T\boldsymbol{x}_i d_{ii}\boldsymbol{x}_i^T\boldsymbol{e} - \sum_{ij}\boldsymbol{e}^T\boldsymbol{x}_i w_{ij}\boldsymbol{x}_i^T\boldsymbol{e}\\ &= \boldsymbol{e}^T\boldsymbol{X}(\boldsymbol{D}-\boldsymbol{W})\boldsymbol{X}^T\boldsymbol{e}\\ &= \boldsymbol{e}^T\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^T\boldsymbol{e}\end{aligned} \tag{2}$$

where $\boldsymbol{D}$ is a diagonal matrix whose entries are column (or row) sums of symmetric W, i.e., $d_{\boldsymbol{jj}} = \sum_i w_{ij}$, $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$ is the Laplacian matrix, and $\boldsymbol{X}$ is the data matrix $[\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n]$. Matrix $\boldsymbol{D}$ provides a natural measure on the data points. The larger the value of d_{ii} (corresponding to $\boldsymbol{y}_i$), the more "important" $\boldsymbol{y}_i$ is. Therefore, a constraint is imposed [39]

$$\boldsymbol{y}^T\boldsymbol{D}\boldsymbol{y} = 1 \Rightarrow \boldsymbol{e}^T\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T\boldsymbol{e} = 1. \tag{3}$$

Accordingly, the minimization problem reduces to find

$$\arg \min_{\substack{\boldsymbol{e}\\ \boldsymbol{e}^T\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T\boldsymbol{e}=1}} \boldsymbol{e}^T\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^T\boldsymbol{e}. \tag{4}$$

That is, to find the solution of the generalized eigenvalue and eigenvector problem [39]

$$\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^T\boldsymbol{e} = \lambda\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T\boldsymbol{e}. \tag{5}$$

Let the column vectors $\boldsymbol{e}_0, \boldsymbol{e}_1, \ldots, \boldsymbol{e}_{l-1}$ be the solutions of (5), ordered according to their eigenvalues $\lambda_0 < \lambda_1 < \ldots < \lambda_{l-1}$. Thus, the embedding of each data point is represented by

$$\boldsymbol{y_i} = \boldsymbol{E}^T\boldsymbol{x_i} \tag{6}$$

where $\boldsymbol{E}$ represents the embedding function $\boldsymbol{E} = [\boldsymbol{e}_0, \boldsymbol{e}_1, \cdots, \boldsymbol{e}_{l-1}]$. The obtained projections are actually the optimal linear approximation to the eigenfunctions of the Laplace Beltrami operator on the manifold. For more details on the derivation and justification of LPP, the reader may refer to [39].

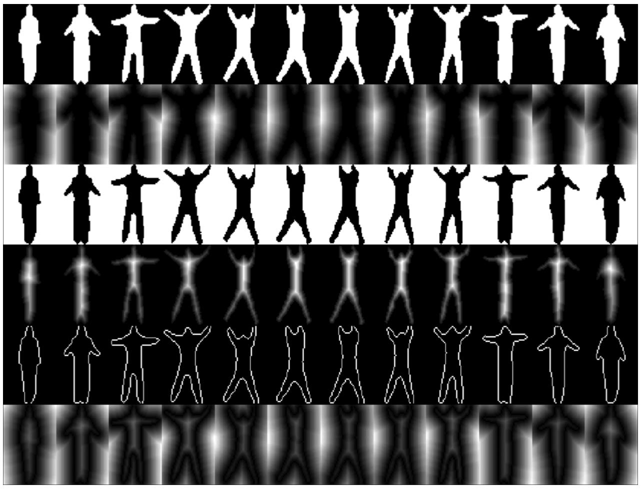


Fig. 1. Visual input representations of a jumping jack action. From top to bottom: $\boldsymbol{I}$, DT-I, $\sim \boldsymbol{I}$, DT-II, edge of $\boldsymbol{I}$, and DT-III, respectively.

B. Representations of Visual Inputs

Effective feature representation is important in the domain of pattern recognition. One basic assumption here is that a sequence of regions of interest (ROI) (i.e., the silhouettes of a moving human) can be obtained from the original video (see Fig. 1). These foreground images are then centred and normalized on the basis of keeping the aspect ratio property of the moving silhouette so that they contain as much foreground as possible and are all of equal dimensions. Considering that establishing correspondences between landmarks on the silhouettes is not always feasible because of the temporal changes in topology and self-occlusions, the resulting normalized images, namely raw silhouette representations, are directly used as visual inputs for action subspace learning.

Alternatively, we also represent each shape instance as an implicit function by using the distance transform (DT) technique [47]. The result of the transformation is a grayscale image that looks similar to the input image. For each pixel in the binary image $\boldsymbol{I}$, the distance transform assigns a number that is the distance between that pixel and the nearest nonzero pixel of $\boldsymbol{I}$ (i.e., the pixels with white color in Fig. 1). There are several different sorts of distance transforms depending upon which distance metric is being used to determine the distance between pixels. We just use the simple Euclidean distance in our experiments. The resulting representations naturally impose smoothness on the distances between the shapes. Note that all these derived representations depend essentially on the silhouettes, thus indirectly reflecting deformations of the dynamic shapes. Fig. 1 shows examples of different representations of visual input, including raw silhouette $\boldsymbol{I}$ and three different DTs corresponding to $\boldsymbol{I}$, reversion of $\boldsymbol{I}$ and the edge map of $\boldsymbol{I}$, named as DT-I, DT-II, and DT-III, respectively.

C. *Subspace Learning*

The original image representations are both noisy and expensive to analyze. We adopt LPP to embed activities into a lower dimensional subspace for more compact representations. Consider c classes (i.e., different actions) where each class represents a sequence of silhouettes (or derived DT representations). For each frame with the resolution of $M \times N$, the image is simply converted into an h-dimensional $(h = M \times N)$ vector $\boldsymbol{v}$ in a raster-scan manner. Let $\boldsymbol{v}_{i,j}$ be the jth input frame in the ith class and N_i the number of such inputs in the ith class. The total number of training samples is $N_t = N_1 + N_2 + \ldots + N_c$, and the whole training data set can be represented by

$$\begin{aligned}\boldsymbol{X} &= [\boldsymbol{v}_{1,1},\, \boldsymbol{v}_{1,2},\, \ldots,\, \boldsymbol{v}_{1,N1},\, \boldsymbol{v}_{2,1},\, \ldots,\, \boldsymbol{v}_{cNc}]\\ &= [\boldsymbol{x}_1,\, \boldsymbol{x}_2,\, \ldots,\, \boldsymbol{x}_{Nt}]\end{aligned} \tag{7}$$

where each column of $\boldsymbol{X}$ is an h-dimensional data point. For each class of action, multiple sequences may be freely added to the training data without altering the following learning procedure.

To construct an affinity matrix $\boldsymbol{W}$, the neighbourhood of each point is directly determined by its K-nearest neighbour points based on the distance measured in the input space. To measure the distance between two data points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, we compute the cosine of the angle between the two vectors

$$S_{ij} = \cos(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{\boldsymbol{x}_i \cdot \boldsymbol{x}_j}{(|\boldsymbol{x}_i| \cdot |\boldsymbol{x}_j|)}. \tag{8}$$

Two points that need to be emphasized are: 1) we do not use the ε-neighborhood to construct the adjacency graph, because it is often difficult to choose an optimal ε in real-world applications, while the K nearest-neighbor graph can be constructed more easily and stably [43], and 2) we do not use temporal information to determine the neighbors of each shape for obtaining an embedding that preserves the geometry of the manifold. We also experiment with the supervised form of LPP (named as S-LPP) by integrating the class information when constructing the affinity graph. That is, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ will be directly connected if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ belong to the same class.

To avoid an extra parameter selection (i.e., for heat kernel function) in the learning process, we simply use the 0–1 weighting scheme to set the weight of the edge, i.e.,

$$w_{ij} = \begin{cases} 1, & \text{the nodes } i \text{ and } j \text{ are connected}\\ 0, & \text{otherwise.}\end{cases} \tag{9}$$

Then we solve the "eigenmaps" problem of (5) to obtain the embedding function $\boldsymbol{E}$. The embedding results of the original data are given by $\boldsymbol{Y} = \boldsymbol{E}^T\boldsymbol{X}$. Each data point $\boldsymbol{v}$ is embedded into a point $\boldsymbol{p}$ in the low-dimensional subspace. A sequential movement of a certain action is accordingly mapped into a trajectory in such a parametric space. Fig. 2 gives two illustrations of action trajectories in 3-D space for visualization, where the size of the marker increases over time, reflecting its temporal progression.

IV. Activity Classification

Action recognition can be solved through measuring motion similarities between the reference motion patterns and test samples in the low-dimensional embedding space. Assume that two action sequences are respectively mapped into $\boldsymbol{A}_1$ $(l \times T_1)$ and $\boldsymbol{A}_2$ $(l \times T_2)$, where l is the reduced dimensionality, and T_1 and T_2 are the durations of these two complete actions respectively. Note that the same activities can have different temporal durations due to speed changes (but have the same moving path), and different activities may have significantly different temporal durations. We select two kinds of distance metrics to measure the motion similarity. Before computing the similarity, each column vector $\boldsymbol{p}$ (corresponding to a frame in the sequence) in the matrixes $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ is normalized, i.e., $\hat{\boldsymbol{p}} = \boldsymbol{p}/\|\boldsymbol{p}\|$.

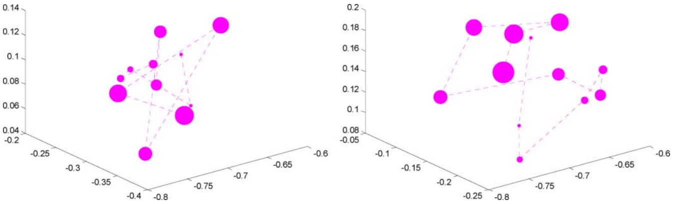


Fig. 2. Examples of action trajectories in 3-D subspace: (*left*) skip and (*right*) jump.

A. *Motion Similarity*

1) Similarity-I: Normalized Spatiotemporal Correlation: Action is a kind of spatiotemporal motion pattern, so we may use spatiotemporal correlation to capture its spatial structural and temporal transitional characteristics by

$$d_1^2 = \min_{s,b} \sum_{t=1}^{T} \left\| A_1'(t) - A_2'(st+b) \right\|^2 \tag{10}$$

where $\boldsymbol{A}_1'$ and $\boldsymbol{A}_2'$ are warped (temporally) versions from $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$, and s and b explain time stretching and shifting respectively. We warp each action trajectory matrix into the same temporal duration T by the bicubic interpolation technique. The piecewise bicubic interpolation produces a much smoother surface than the bilinear interpolation since the value of an interpolated point is a combination of the values of the sixteen closest points, which can be a key advantage for applications like image processing. Accordingly, b will be selected within $[0, 1, \ldots, T-1]$. To reduce the influence of the interpolation in real operations, we may simply select T as max (T_1, T_2), which will at least ensure that each time only one matrix needs to be warped. Assume that $T_1 > T_2$, $\boldsymbol{A}_1$ will keep unchanged, and $\boldsymbol{A}_2$ will be temporally warped [i.e., $s = T_2/T_1$, as represented in (10)]. Otherwise, the roles of $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ will be swapped in (10). Note that the converted matrixes have the same dimension, so the shifting here is circular, not linear. The computation manner of Similarity-I usually requires knowing the temporal duration of each one action for such an approximate frame-to-frame matching. If the length of the input test is much less (or more) than one duration (a whole action as reference), the recognition accuracy will naturally degrade.

2) Similarity-II: Median Hausdorff Distance: The action duration is not necessarily always easy to estimate (and segment) in real videos. However, the computed trajectory of each sequence depends on the duration and temporal shift of the action. A distance measure that can handle changes in duration and temporal shifts is, thus, ideal. The Hausdorff distance measure provides a means of determining the resemblance of one point set to another, by examining the fraction of points in one set that lie near points in the other set (and vice versa). We use a variant of the Hausdorff metric, i.e., the median value of the minimum

$$S(\boldsymbol{A}_1, \boldsymbol{A}_2) = \underset{i}{\text{median}} \left(\min_j \left(\|\boldsymbol{A}_1(i) - \boldsymbol{A}_2(j)\| \right) \right). \quad (11)$$

Since the Hausdorff distance is oriented, to ensure symmetry, we modify the final distance measure to

$$d_2 = S(\boldsymbol{A}_1, \boldsymbol{A}_2) + S(\boldsymbol{A}_2, \boldsymbol{A}_1). \quad (12)$$

The smaller the distance measure is, the more similar the two actions are. Note that the computation manner of Similarity-II implicitly includes the temporal constraints between frame-based observation vectors, i.e., the closest points between two point sets ideally have the same temporal order in similar motion classes. Note that we find from our experiments that the median Hausdorff distance exhibits similar results to the mean Hausdorff distance on the dataset we use [27], though the "median" is generally thought to be more robust than the "mean," especially when outliers exist.

B. Classifier

Action classification is performed in a nearest neighbour framework. Let TA represent a test action sequence and R_i represent the ith reference action sequence. We will classify this test as the class c that can minimize the similarity distance between the test sequence and all reference patterns, i.e.,

$$c = \underset{i}{\arg\min}\, d(TA, R_i) \quad (13)$$

where d is the similarity measure d_1 or d_2 defined above. No doubt, a more sophisticated classifier could be employed, but the interest here is to evaluate the genuine discriminatory powers of the used features.

V. Experimental Results

Extensive experiments have been carried out to verify the effectiveness of the proposed method. The following describes the details of the experiments and the results. Note that the evaluation reported here is generally in terms of the percentage of the correctly recognized actions among all tests.

A. Evaluation Dataset

Due to the lack of a common evaluation database, of a reasonable size, in the domain of human action recognition; most researchers usually evaluate their methods on individual databases with a different number and category of actions (see Table V for a simple summary of some action databases currently in use). In this paper, we use a recent database reported in [27]. To the best of our knowledge, this database is one of the few, reasonably sized (in terms of the number of subjects, actions and videos), concurrent action databases available in the public domain. It consists of 81 low-resolution videos (180 × 144, 25 fps) from nine different people, each performing nine natural actions. These actions include bending (bend), jumping jack (jack), running (run), walking (walk), jumping-forward-on-two-legs (jump), jumping-in-place-on-two-legs (pjump), galloping-sideways (side), waving-one-hand (wave1), and waving-two-hands (wave2). Together with one more recently added action of skipping (skip), the dataset in our experiments in total includes ten actions and 90 videos. These actions are either periodic (e.g., run and walk) or nonperiodic actions (e.g., bend), and either stationary (e.g., wave1 and wave2) or nonstationary motions along both horizontal (e.g., skip), and vertical directions (e.g., jack). Please refer to http://www.wisdom.weizmann.ac.il/~vision/SpaceTimeActions.html.

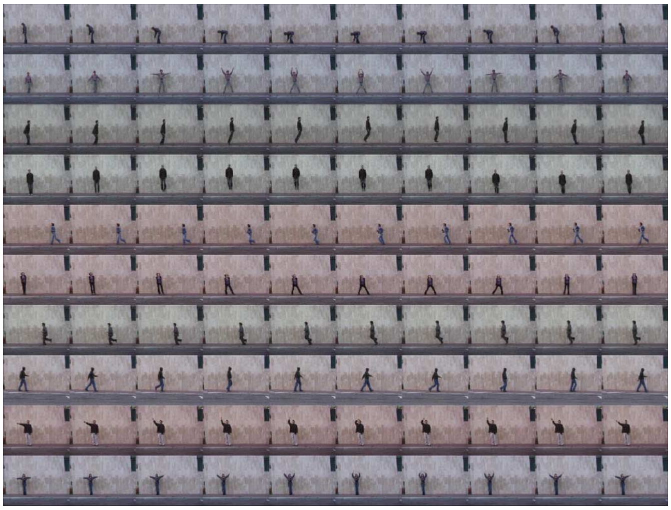


Fig. 3. Some example images of actions. From top to bottom: bend, jack, jump, pjump, run, side, skip, walk, wave1, and wave2, respectively.

Sample images of each action video are shown in Fig. 3, from which we can see that some actions are somewhat similar in the sense that the limbs have similar motion paths and forms; and this high degree of similarity makes discrimination more challenging. In addition, people have different physical characteristics and perform activities differently both in motion styles and speeds. Different people are asked to perform the same actions in this dataset, thus providing more realistic data for the test of the method's versatility.

B. Data Processing

We directly use the foreground masks from [27], though they are not very perfect. These masks are obtained by background subtraction in color-space. It should be noted that, whether the actions in this dataset are in essence periodic or not, people are asked to perform them multiple times in a repetitive manner (except for bending). Each action video generally includes 2∼4 complete action cycles. This property allows us to easily compute each action's duration in these videos by considering them as semantically periodic motions (for bending, the real length of the video is simply selected as its duration). It needs to be emphasized that, for Similarity-I, we need to compute the action's duration for accurate matching; however, Similarity-II has no such requirement (see Section V-D).

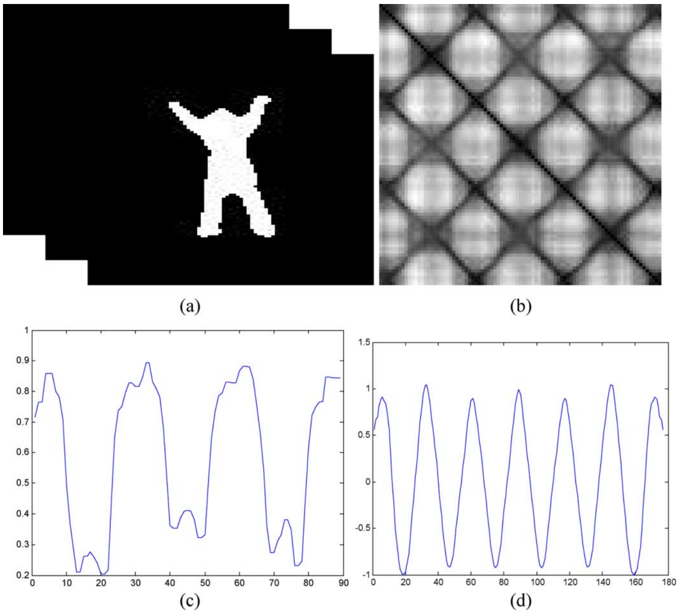


Fig. 4. Periodicity analysis. (a) Input sequence, (b) self-similarity S, (c) a column vector $\boldsymbol{z}$ of S, and (d) its autocorrelation.

To estimate the action cycle of each silhouette video [Fig. 4(a)], object O_ts self-similarity is computed at times t_1 and t_2 based on the similarity measure of the absolute correlation [48]

$$S_{t_1,t_2} = \min_{|dx,dy|<r} \sum_{(\boldsymbol{x},\boldsymbol{y})\in B_{t1}} |O_{t_1}(x+dx,y+dy) - O_{t_2}(x,y)| \tag{14}$$

where B_{t1} is the bounding box of the object O_{t1}, and a small search radius r is introduced to account for the segmentation error. For periodic motions, S will be also periodic [Fig. 4(b)], where dark regions show more similarity. Periodic motions will have dark lines parallel to the diagonal of S. To determine if an object exhibits periodicity, we can analyze the column vectors $\boldsymbol{z}$ of S [i.e., a certain fixed t_1 and all t_2, Fig. 4(c)]. We first linearly detrended of this column vector to $\hat{\boldsymbol{z}}$ by subtracting its mean value and then divided by its standard variance, then compute its autocorrelation [Fig. 4(d)] by

$$\boldsymbol{R}_{\hat{\boldsymbol{z}}\hat{\boldsymbol{z}}}(m) = \frac{1}{N-|m|}E\{\hat{\boldsymbol{z}}_{n+m}\hat{\boldsymbol{z}}_n\}. \tag{15}$$

Finally, we compute its first-order derivative to find peak positions by seeking the positive-to-negative zero-crossing points. We estimate the real duration as the average distance between each pair of consecutive peaks. A more accurate duration can be estimated by averaging the results of multiple t_1s. This process has been demonstrated to be computationally feasible with respect to current mask results.

We select two complete cycles from the middle part for each original action video for the following experiments. Note that, when selecting them, we do not, and need not, temporally assign onset and ending for each class of action. The resulting dataset includes in total 171 sequences $(9 \times 2 \times 9 + 9 \times 1 \times 1)$, i.e., each person has 1 sequence for bending and two sequences for each of nine other actions, each of which includes an action with a complete duration (cycle). We center and normalize all silhouette images in these sequences into the same dimension (i.e., 64 × 48 pixels), and convert them into 3072-dimensional input representations in a manner described in Section III. A considerable portion of such visual inputs is used to learn the action subspace. Fig. 5 shows two examples of action distribution where only 3-D space is used for visualization, in which the points with same color are from the same action. From Fig. 5, we can see that, the same actions have satisfactory clustering, and different actions are also distinguishable. Note that jump, run and skip are relatively closer in the embedding space due to their high similarities.

C. Results and Analysis

1) Identification Mode: In identification mode, the classifier determines which class a given measurement belongs to in the nearest-neighbor framework. For a small number of examples, we compute an overall unbiased estimate of the true recognition accuracy using the leaving-one-out cross-validation method. Each time, we first leave one sequence out. Considering that repeated performances of the same action performed by the same human vary quite slightly, the same action sequence taken from the same original video is also removed, following [27], while other actions of the same subject remain. Then, we train on all the remaining sequences, and classify the omitted element according to its similarity differences with respect to the rest of the examples. Thus, if this left-out sequence is classified correctly, it must exhibit high similarity to a sequence from a different person performing the same action. Fig. 6 shows correct classification rates (CCR) of action recognition.

From Fig. 6, the following basic conclusions can be drawn. 1) Dynamic shape manifolds are indeed informative (enough to correctly classify human actions performed by people with different body build and different motion styles). 2) Generally, the supervised LPP obtains better results than unsupervised LPP (naturally because it integrates class label information in the training process, thus increasing the discrimination ability). 3) Overall, DT-III performs best among all input representations, but it only slightly outperforms DT-I. This may be due to the fact it imposes more smoothness of the distance values reflecting shape changes within or outside of the silhouette shape. 4) DT-II performs worst, yet it has about 95% correct classification rates for both similarity measures. 5) The selection of K within 5∼20 does not make big difference on the results, which means that it is very easy to select K to achieve stable classification rates. 6) Similarity-II performs somewhat better than Similarity-I. On the one hand, the median Hausdorff distance is more robust than the frame-to-frame correlation; however, data interpolation in Similarity-I may bring a negative effect on the results.

2) Verification Mode: For completeness, we also estimate the FAR (false acceptance rate) and FRR (false reject rate) via the leave-one-out rule in verification mode. For the verification mode, the pattern classifier is asked to verify whether a new

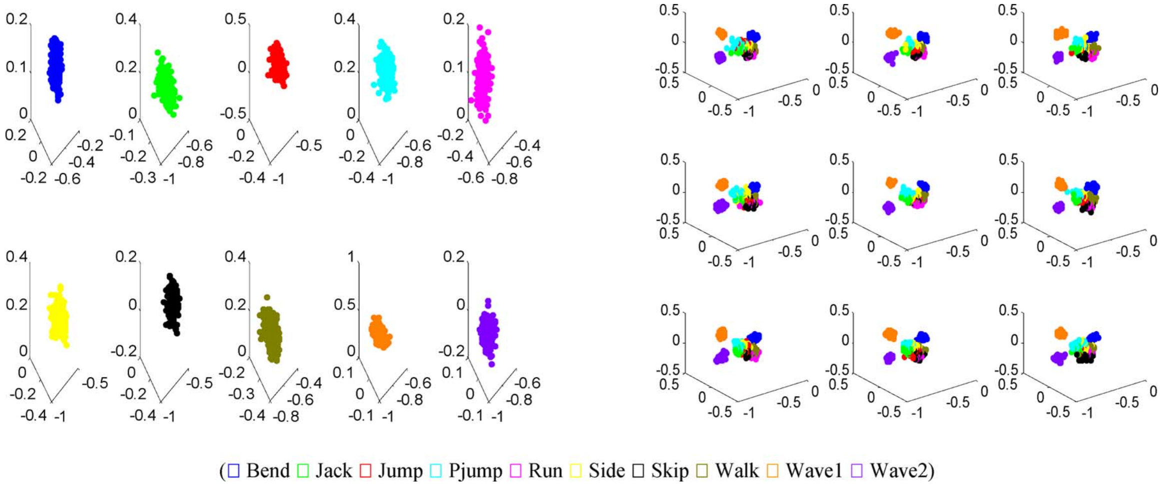


Fig. 5. Visualizations of action manifolds: (*left*) each of ten actions from nine subjects and (*right*) ten actions from each of nine different subjects.

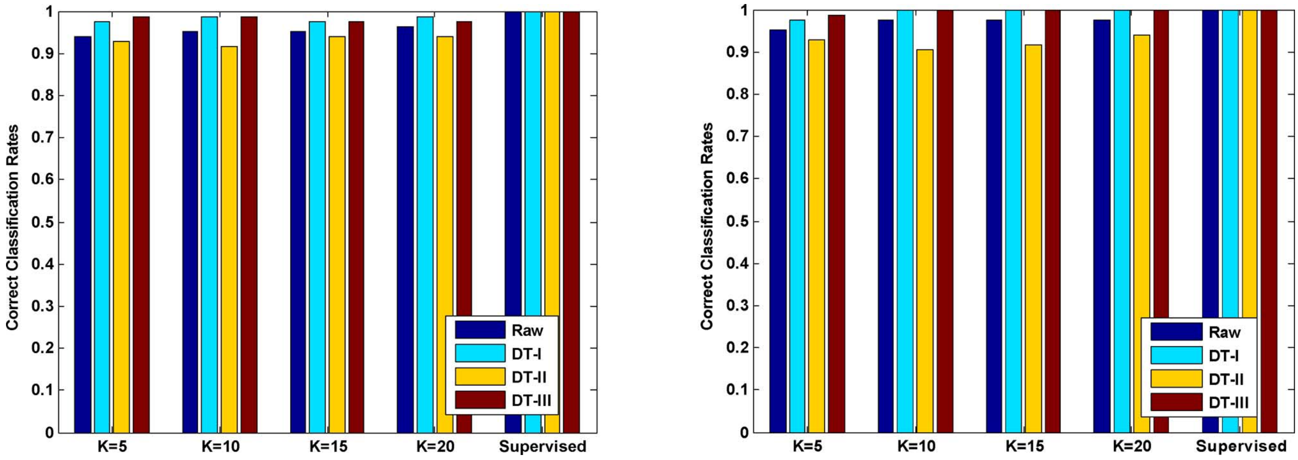


Fig. 6. Action classification results: (left) similarity-I and (right) similarity-II.

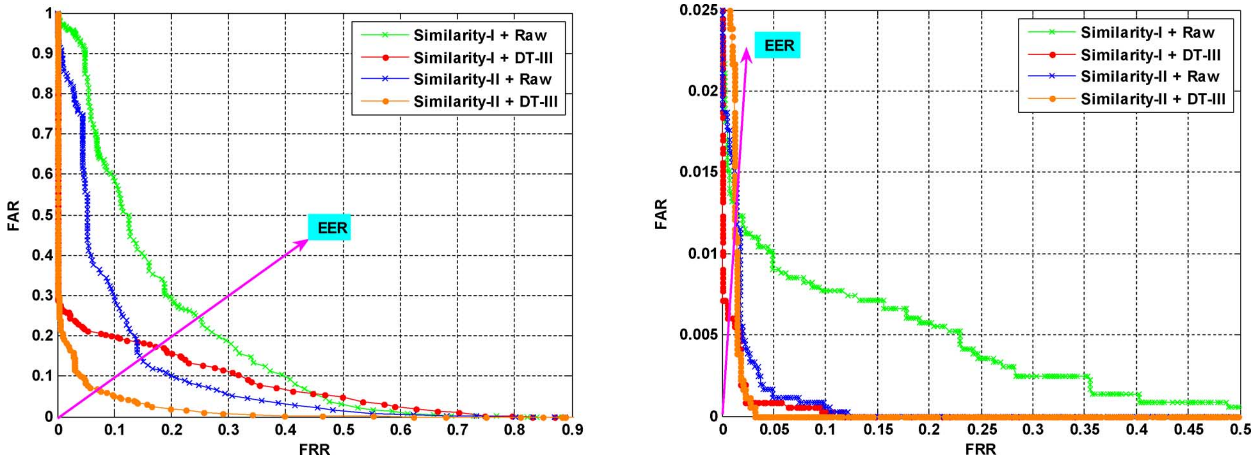


Fig. 7. ROC curves: unsupervised situation with (left) $K = 20$ and (right) supervised situation.

measurement really belongs to certain claimed class. Note that, each time there is only one genuine attempt and nine imposters since the left-out sequence is known to belong to one of the ten action classes. Fig. 7 shows the ROC (receiver operating characteristics) curves using both raw and DT-III representations, from which some similar conclusions (to the identification mode) can be drawn, e.g., the supervised LPP performs better than the unsupervised one, i.e., S-LPP has much lower EERs (equal error rate).

3) Reduced Dimension: Another important aspect is to examine the relationship between the reduced dimensions and the recognition rates. Fig. 8 shows the classification performance

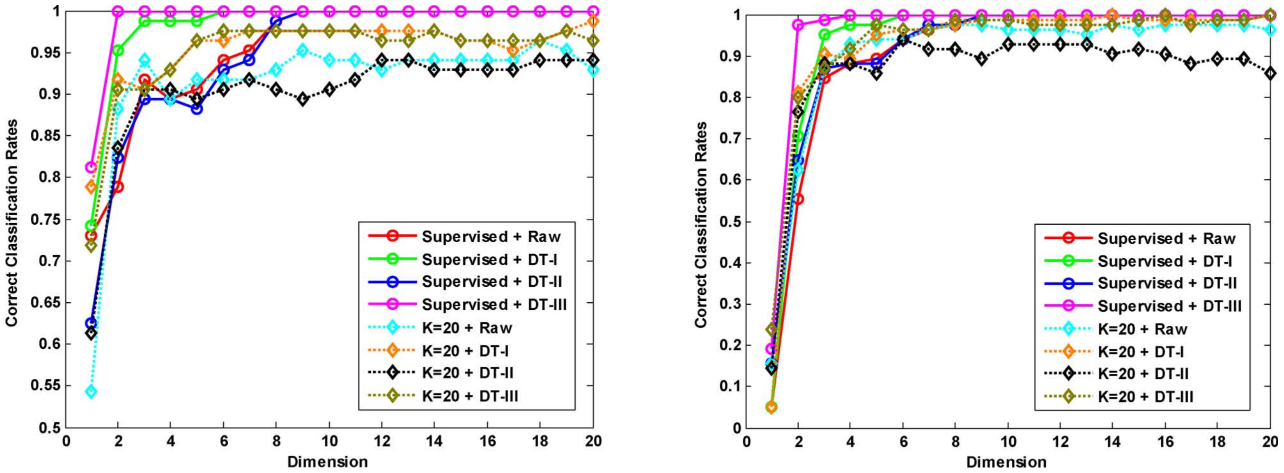
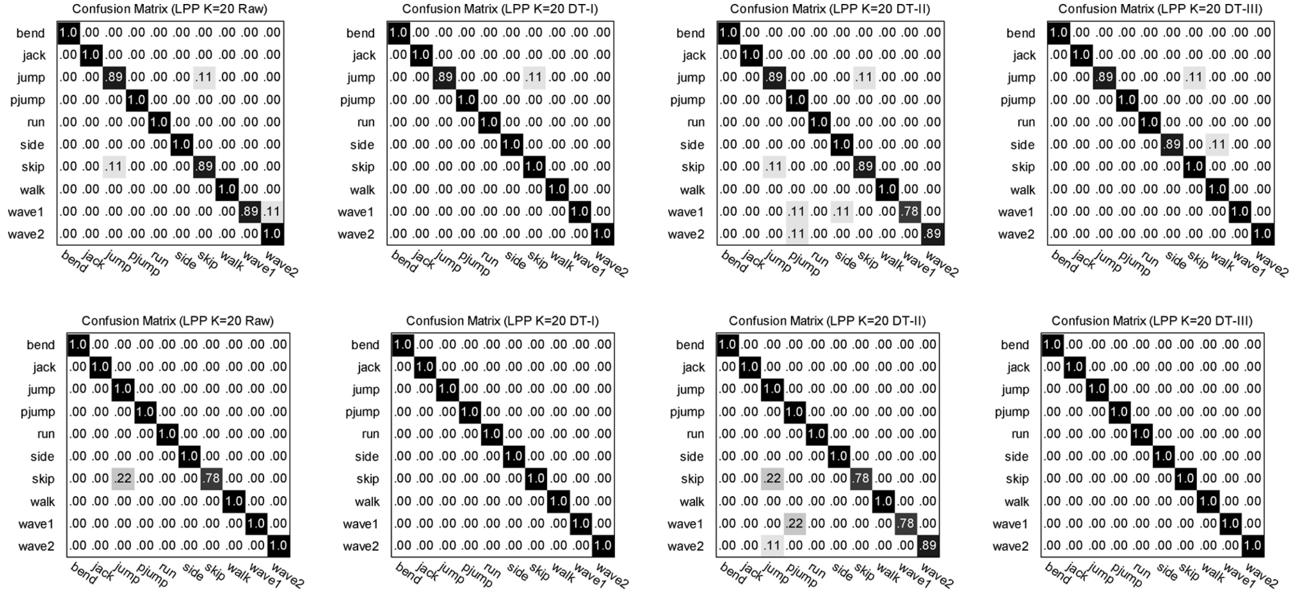


Fig. 8. Recognition rate versus the reduced dimension l: (left) Similarity-I and (right) Similarity-II.



Confusion Matrix (LPP K=20 Raw)

	bend	jack	jump	pjump	run	side	skip	walk	wave1	wave2
bend	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00
jack	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00
jump	.00	.00	.89	.00	.00	.00	.11	.00	.00	.00
pjump	.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00
run	.00	.00	.00	.00	1.0	.00	.00	.00	.00	.00
side	.00	.00	.00	.00	.00	1.0	.00	.00	.00	.00
skip	.00	.00	.11	.00	.00	.00	.89	.00	.00	.00
walk	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00
wave1	.00	.00	.00	.00	.00	.00	.00	.00	.89	.11
wave2	.00	.00	.00	.00	.00	.00	.00	.00	.00	1.0

Confusion Matrix (LPP K=20 DT-I)

	bend	jack	jump	pjump	run	side	skip	walk	wave1	wave2
bend	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00
jack	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00
jump	.00	.00	.89	.00	.00	.00	.11	.00	.00	.00
pjump	.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00
run	.00	.00	.00	.00	1.0	.00	.00	.00	.00	.00
side	.00	.00	.00	.00	.00	1.0	.00	.00	.00	.00
skip	.00	.00	.00	.00	.00	.00	1.0	.00	.00	.00
walk	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00
wave1	.00	.00	.00	.00	.00	.00	.00	.00	1.0	.00
wave2	.00	.00	.00	.00	.00	.00	.00	.00	.00	1.0

Confusion Matrix (LPP K=20 DT-II)

	bend	jack	jump	pjump	run	side	skip	walk	wave1	wave2
bend	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00
jack	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00
jump	.00	.00	.89	.00	.00	.00	.11	.00	.00	.00
pjump	.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00
run	.00	.00	.00	.00	1.0	.00	.00	.00	.00	.00
side	.00	.00	.00	.00	.00	1.0	.00	.00	.00	.00
skip	.00	.00	.11	.00	.00	.00	.89	.00	.00	.00
walk	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00
wave1	.00	.00	.00	.11	.00	.11	.00	.00	.78	.00
wave2	.00	.00	.00	.11	.00	.00	.00	.00	.00	.89

Confusion Matrix (LPP K=20 DT-III)

	bend	jack	jump	pjump	run	side	skip	walk	wave1	wave2
bend	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00
jack	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00
jump	.00	.00	.89	.00	.00	.00	.11	.00	.00	.00
pjump	.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00
run	.00	.00	.00	.00	1.0	.00	.00	.00	.00	.00
side	.00	.00	.00	.00	.00	.89	.00	.11	.00	.00
skip	.00	.00	.00	.00	.00	.00	1.0	.00	.00	.00
walk	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00
wave1	.00	.00	.00	.00	.00	.00	.00	.00	1.0	.00
wave2	.00	.00	.00	.00	.00	.00	.00	.00	.00	1.0

Confusion Matrix (LPP K=20 Raw)

	bend	jack	jump	pjump	run	side	skip	walk	wave1	wave2
bend	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00
jack	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00
jump	.00	.00	1.0	.00	.00	.00	.00	.00	.00	.00
pjump	.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00
run	.00	.00	.00	.00	1.0	.00	.00	.00	.00	.00
side	.00	.00	.00	.00	.00	1.0	.00	.00	.00	.00
skip	.00	.00	.22	.00	.00	.00	.78	.00	.00	.00
walk	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00
wave1	.00	.00	.00	.00	.00	.00	.00	.00	1.0	.00
wave2	.00	.00	.00	.00	.00	.00	.00	.00	.00	1.0

Confusion Matrix (LPP K=20 DT-I)

	bend	jack	jump	pjump	run	side	skip	walk	wave1	wave2
bend	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00
jack	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00
jump	.00	.00	1.0	.00	.00	.00	.00	.00	.00	.00
pjump	.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00
run	.00	.00	.00	.00	1.0	.00	.00	.00	.00	.00
side	.00	.00	.00	.00	.00	1.0	.00	.00	.00	.00
skip	.00	.00	.00	.00	.00	.00	1.0	.00	.00	.00
walk	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00
wave1	.00	.00	.00	.00	.00	.00	.00	.00	1.0	.00
wave2	.00	.00	.00	.00	.00	.00	.00	.00	.00	1.0

Confusion Matrix (LPP K=20 DT-II)

	bend	jack	jump	pjump	run	side	skip	walk	wave1	wave2
bend	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00
jack	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00
jump	.00	.00	1.0	.00	.00	.00	.00	.00	.00	.00
pjump	.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00
run	.00	.00	.00	.00	1.0	.00	.00	.00	.00	.00
side	.00	.00	.00	.00	.00	1.0	.00	.00	.00	.00
skip	.00	.00	.22	.00	.00	.00	.78	.00	.00	.00
walk	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00
wave1	.00	.00	.00	.22	.00	.00	.00	.00	.78	.00
wave2	.00	.00	.11	.00	.00	.00	.00	.00	.00	.89

Confusion Matrix (LPP K=20 DT-III)

	bend	jack	jump	pjump	run	side	skip	walk	wave1	wave2
bend	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00
jack	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00
jump	.00	.00	1.0	.00	.00	.00	.00	.00	.00	.00
pjump	.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00
run	.00	.00	.00	.00	1.0	.00	.00	.00	.00	.00
side	.00	.00	.00	.00	.00	1.0	.00	.00	.00	.00
skip	.00	.00	.00	.00	.00	.00	1.0	.00	.00	.00
walk	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00
wave1	.00	.00	.00	.00	.00	.00	.00	.00	1.0	.00
wave2	.00	.00	.00	.00	.00	.00	.00	.00	.00	1.0

Fig. 9. Confusion matrixes of action classification: (top) Similarity-I and (bottom) Similarity-II.

vs. the reduced dimensions up to 20, from which we can see that: 1) the recognition rate increases rapidly during the first few values of l. As the dimension l increases, the recognition rate is usually improved as expected and finally approaches a basically stable level; 2) the S-LPP generally needs a lower dimension than the unsupervised one to obtain good results; and 3) using a smaller dimension l is computationally more efficient but may result in a lower recognition rate. Fortunately, our results have shown that LPP has remarkably reduced the dimensionality, and the proposed method generally does not need a high dimension (e.g., about 8) to obtain very satisfactory results.

4) Confusion Matrix: For the unsupervised LPP, there exist a few false classifications. To analyze which action sequences (and why) are incorrectly classified, we show confusion matrixes with respect to different similarity measures and visual inputs in Fig. 9. The elements of each row in the confusion matrix represent the probability that certain kind of action is classified as other kinds of actions.

From Fig. 9, it can be seen that most actions have perfect classification, and only quite a small number of actions, especially skip and jump, wave and pjump, are easily confused. In addition to high similarities among most silhouette shapes in these actions (with local similar moving patterns), poor foreground segmentation may contribute to these confusions. From the experiments, we also observe that the correct classification of all these confused actions is generally within the first five best choices.

D. Robustness Test

To further evaluate performance of the proposed method, we construct several experiments for testing robustness with respect to periodicity, silhouette quality, and other challenging factors.

1) Length of Test Sequences: To examine the influence of the test sequence length on the recognition results, we assess 90 original videos by applying the classifier with Similarity-II on randomly selected subsequences of each action, each of which includes varying portions of cycles. Note that all reference action patterns include one cycle (i.e., a complete action). The classification results are shown in Fig. 10 (left), from which we can see that 1) the recognition accuracy is satisfactory, even for

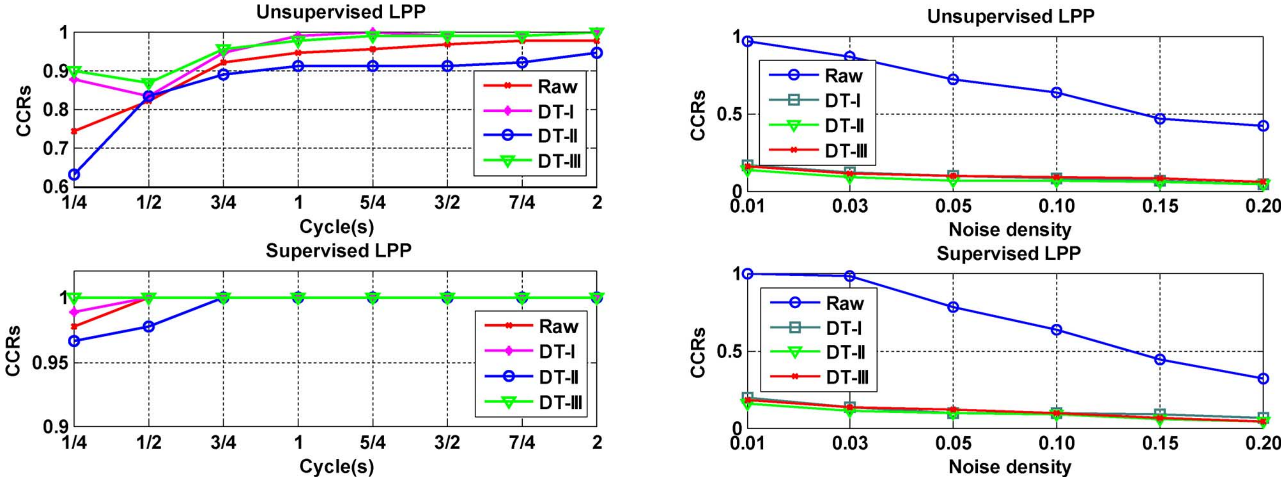


Fig. 10. Performance evaluation: (left) CCRs versus the length of test sequences and (right) CCRs versus synthetic noise.

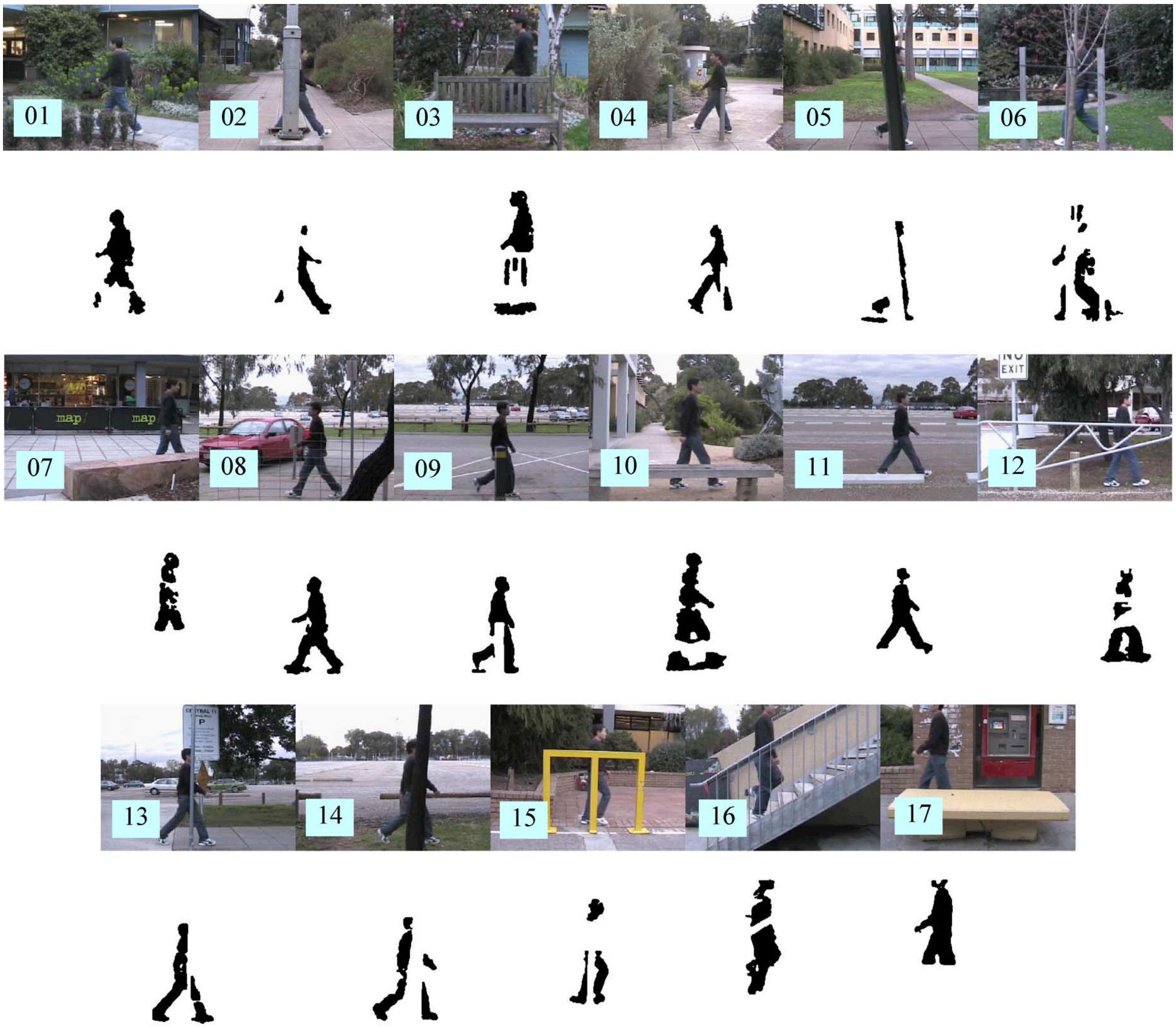


Fig. 11. Example images and the associated corrupted silhouettes due to occlusions.

partial matching with less than one cycle, 2) S-LPP is less subject to the test sequence length than LPP, and 3) one cycle is enough to provide the steady results in all cases.

2) Corrupted Silhouettes by Synthetic Noise: The silhouette masks used for the above experiments are relatively noise-free. A simple method to check sensitivity to noise is to add various amounts of synthetic noise to all the silhouette images to simulate corrupted silhouettes. Since the silhouette image is binary, we use "salt & pepper" noise for this experiment. A parameter, the noise density, is used to represent the percentage of the af-

TABLE I
PERFORMANCE EVALUATION ON REAL CORUPTED SILHOUETTE SEQUENCES

Supervised LPP plus Similarity-II (S: Successful and F: Failed)																	
No.	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17
Raw	F	S	F	S	S	S	S	S	S	F	S	F	S	S	F	F	S
DT-I	S	F	F	S	F	F	S	S	S	F	S	F	F	F	F	F	F
DT-II	F	F	F	F	F	F	F	S	S	S	S	F	S	F	F	F	F
DT-III	S	F	F	S	F	F	F	S	S	F	S	F	F	F	F	F	S

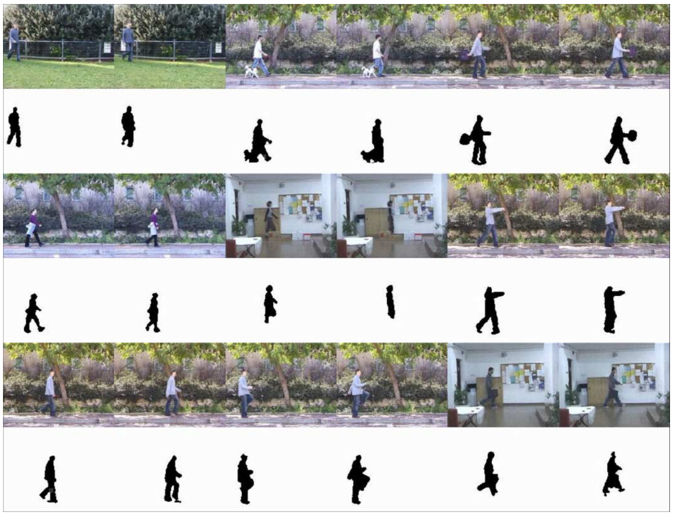


Fig. 12. Test walking sequences. From left to right and from top to bottom: diagonal walk, walk with a dog, walk and swinging a bag, walk in a skirt, walk with the legs occluded partially, sleepwalking, limping, walk with knees up, and walk when carrying a briefcase, respectively.

fected pixels in the whole image. The classification results are shown in Fig. 10 (right), from which we can see that 1) raw silhouette representation can tolerate a considerable amount of noise (e.g., 5%), and 2) all distance transformed representations are very sensitive to such kind of isolated noise. This is mainly because, unlike raw silhouette representation, such form of noise greatly affects much more pixels in the whole image when computing the distance values of each pixel (i.e., the noise influence is significantly enlarged).

3) Corrupted Silhouettes by Real Occlusions: The synthetic "salt & pepper" noise cannot necessarily reflect actual low-quality silhouettes in real world. Current motion detection methods often perform postprocessing on the background-subtracted images based on morphological operators and connected component analysis to remove isolated noise and small-area cluttered motions. To further evaluate the performance of the method on the actually corrupted silhouettes, we specially collect some sequences in which human silhouettes are corrupted to different extents, by different forms and degrees of occlusions, as well as unconstrained environments.

Since the walking action is one of the most common motions in real life, and reliable analysis of it is very important for other related research such as gait recognition, we captured 17 walking sequences for this experiment. Some example images and the associated silhouettes are shown in Fig. 11, and the recognition results are summarized in Table I. Note that here we just use a simple Gaussian-based background model in the RGB color space for silhouette extraction (shadows around feet, and low-contrast between background and foreground, influence the silhouette quality). We only select from the video a subsequence where the occlusion occurs from video for the test. From Table I, it can be seen that our method can considerably tolerate low-degree or high-degree but momentary occlusions, especially for raw silhouette representation (all DTs are still sensitive compared with raw silhouette representations). For significantly corrupted silhouettes due to long and severe occlusions, our method mostly fails.

4) Other Challenging Factors: We also address the performance of our method with respect to other challenging factors such as viewpoints, different clothes and motion styles, etc. Here we use 10 walking sequences captured in various different scenarios in front of nonuniform background [27] for this experiment. Some example images and the associated silhouette segmentation results are shown in Fig. 12. In contrast to corrupted silhouettes in Fig. 11, the silhouettes here embody deformations of human shapes, compared with normal walking pattern.

We report the test results using the raw and DT-III representations together with Similarity-II. The above experiments have already shown that the matching using the sequence with multiple repeated action cycles; and the matching using the cycle-segmented sequences, have similar correct classification rates when using Similarity-II (i.e., the point-set matching properties of the Hausdorff distance makes it able to handle different time durations and shifting). We directly use these whole walking sequences without segmenting into different cycles in this experiment. Each reference action type is matched in these walking sequences to find the best matched action type. Table II summarizes the results, from which it can be seen that, except for three (for raw representation) or two (for DT-III) sequences, all other test sequences are correctly classified as the "walk" action. This shows that the proposed method has relatively low sensitivity to considerable changes in scale, viewpoint (30∼40 degrees) and clothes, high irregularity in walking forms, etc.

E. Comparisons

1) Comparison I—Different Dimension Reduction Methods: The purpose of this comparison is to find which method is more effective for subspace learning of dynamic shapes of human actions. Here we select two linear methods (i.e., PCA and LDA), two nonlinear ones (i.e., LLE and LE), and 171 sequences in Section V-B for this experiment. PCA [34] is an eigenvector method designed to model linear variation in high-dimensional data. LDA [35] searches for the projection axes on which the data points of different classes are far from each other while

TABLE II
RESULTS OF ROBUSTNESS TEST EXPERIMENTS

Test sequences	Conditions	Classification results Raw (1st and 2nd best matching)	 DT-III (1st and 2nd best matching)
Diagonal walk	Scale and Viewpoint	Walk (pjump)	Walk (pjump)
Walking with a dog	Non-rigid deformation	**Run** (skip)	**Run** (skip)
Swinging bag	Rigid deformation	**Skip** (walk)	Walk (bend)
Walking in a skirt	Clothes	Walk (side)	Walk (jump)
Occluded legs	Partial occlusion	Walk (jump)	Walk (jump)
Sleepwalking	Walking style	**Side** (skip)	**Jump** (walk)
Limping man	Walking style	Walk (jump)	Walk (jump)
Knees up	Walking style	Walk (jump)	Walk (jump)
Carrying briefcase	Carried object	Walk (skip)	Walk (skip)
Normal walk	Background	Walk (skip)	Walk (skip)

TABLE III
COMPARISON OF CCRS USING DIFFERENT DIMENSION REDUCTION METHODS PLUS SIMILARITY-II

Methods Visual input	Correct Classification Rates (%) (Leave-one-out rule) PCA	 LDA	 S-LPP	 LPP	 LE	 LLE
Raw	98.83 (Dim=15)	98.83 (Dim=9)	100.0 (Dim=9)	97.66 (Dim=8, K=20)	94.15 (Dim=12, K=15)	92.98 (Dim=13, K=10)
DT-I	98.83 (Dim=22)	100.0 (Dim=8)	100.0 (Dim=6)	100.0 (Dim=14, K=20)	100.0 (Dim=13, K=15)	95.32 (Dim=18, K=10)
DT-II	92.98 (Dim=18)	97.66 (Dim=6)	100.0 (Dim=9)	94.15 (Dim=6, K=20)	92.98 (Dim=18, K=15)	91.81 (Dim=20, K=10)
DT-III	97.66 (Dim=30)	98.83 (Dim=6)	100.0 (Dim=4)	100.0 (Dim=16, K=20)	98.83 (Dim=18, K=15)	96.49 (Dim=19, K=10)
Statistical significance Visual input	Average CCR [Confidence intervals] (%) PCA	 LDA	 S-LPP	 LPP	 LE	 LLE
Raw	91.72 [90.73, 92.71]	94.66 [94.02, 95.30]	99.98 [99.94, 100.0]	92.38 [91.51, 93.25]	88.60 [87.66, 89.54]	85.64 [84.47, 86.81]
DT-III	93.44 [92.50, 94.38]	94.48 [93.63, 95.33]	100.0 [100.0, 100.0]	97.62 [97.03, 98.21]	93.32 [92.53, 94.11]	88.84 [87.89, 89.79]

requiring data points of the same class to be close to each other. For LLE [36], assuming that each data point and its neighbors lie on a locally linear patch of the manifold, each point can be reconstructed as linear combinations of its local neighbors. The objective is to find the construction weights that minimize the global reconstruction errors. In LE [38], the embedding maps for the data come from the approximation to the Laplace–Betrami operator defined on the entire manifold. For each method, we carefully determine the optimal parameters by experiments, in order to provide fair comparison. We report the best results using all 171 sequences (with Similarity-II and the leave-one-out rule) in Table III; and visualizations using the raw representation are shown in Fig. 13. In order to observe the statistical significance of the results among various algorithms, we also establish confidence intervals with respect to CCRs. Each time, we randomly sample 100 sequences from the whole dataset without replacement as a subset for testing, and the remaining 71 sequences (including all ten classes) for training. Such testing procedure is repeated 1000 times. The associated results with 95% confidence intervals using both raw and DT-III inputs are also listed in Table III.

From Fig. 13 and Table III, some conclusions can be drawn with respect to the reduced dimensions, CCRs, statistical significance, and the clustering abilities in the embedding space: 1) LPP and S-LPP generally perform better, and DT-III representation is superior to raw representation. 2) Amongst the two kinds of supervised methods, S-LPP outperforms LDA. This is probably because LPP can discover the nonlinear structure of activity manifolds more effectively. 3) As a supervised method, LDA performs a little better than PCA and the unsupervised LLE, but similar to the unsupervised LPP and LE. 4) Interestingly, nonlinear LE and LLE performs not as well as those linear methods. Possible explanations are that the practical data have high curvature both in the observation space and in the embedded space, and that parameter adjustment in LE or LLE-based manifold learning and extrapolation (which is relatively hard to do well) probably brings considerable influences on the results. 5) For any one representation, PCA, LE and LLE usually need a larger dimension (compared with supervised LDA and S-LPP), which naturally increases the computation complexity. 6) Unlike the experiments using the leave-one-out cross-validation rule (in which each test action has multiple reference templates with the same action class for matching, i.e., eight templates for bending and 16 templates for other nine actions), there are relatively less templates for each action to be tested in the estimation of statistical significance, thus leading to a little lower average CCRs for each dimension reduction method. 7) Although action measurements are inherently nonlinear across the whole action, linear PCA and LDA provide good discrimination rates, just somewhat lower than LPP overall (though LPP has been demonstrated to be far superior to PCA and LDA in static face recognition [43]). This is probably because that the action is not considered as one entity but a sequence of entities, thus the introduction of temporal relation to some extent increases their discriminating powers. However, compared with S-LPP, PCA and LDA need an overall higher dimensionality to obtain good results, thus leading to higher computational cost. 8) In particular, LPP and S-LPP have relatively better visual

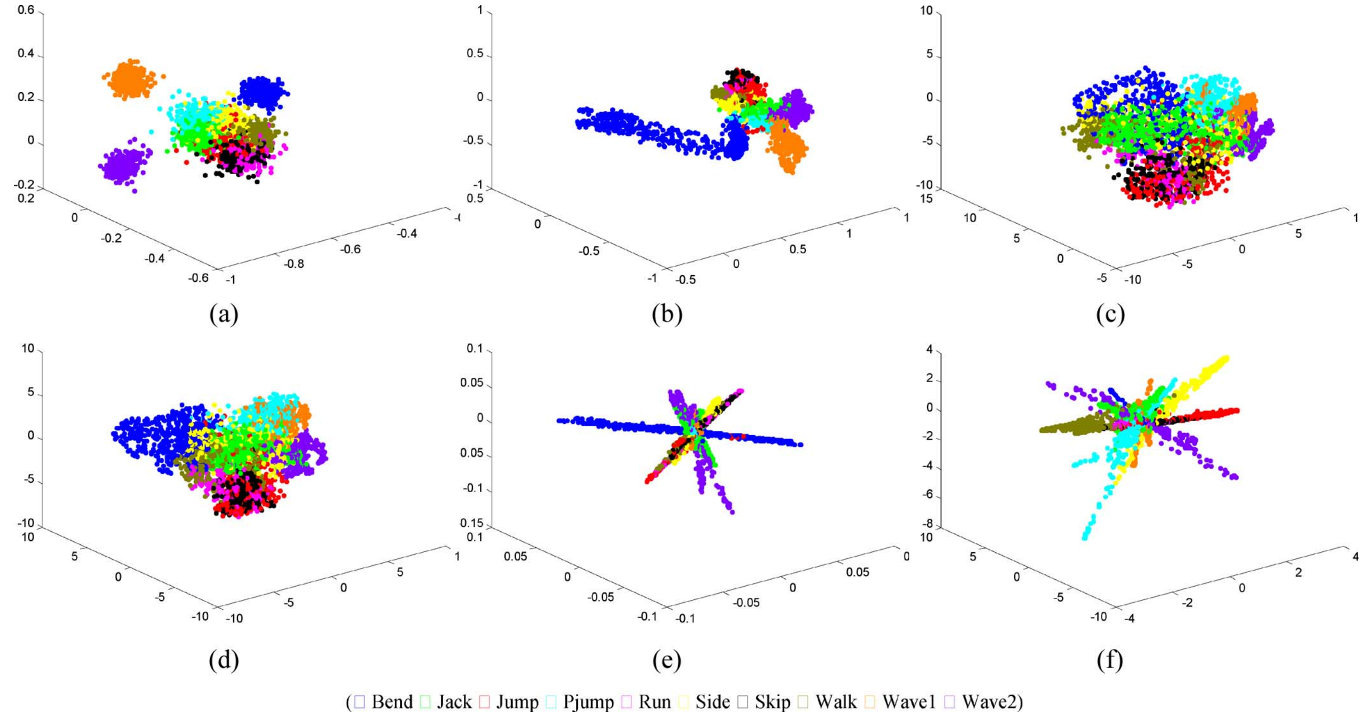


Fig. 13. Three-dimensional visualization of projections of all action sequences using different dimension reduction methods with the raw input. (a) S-LPP; (b) LPP; (c) PCA; (d) LDA; (e) LE; (f) LLE.

TABLE IV
COMPARISION OF OUR METHOD WITH [27] AND [32]

	Action Classification		
Methods	Test dataset	Best accuracy	Brief comments of methods
[32], AVSS'03	8 actions: walk, run, skip, line-walk, hop, march, side-walk, side-skip, 168 sequences, 21 subjects	92.8%	Filtered images together with PCA
[27], ICCV'05	9 actions (no skip), 81 videos, 9 subjects -549 cubes by a sliding window with time overlapping	99.6%	Solution of the Poisson equation of space-time shapes
Our method	10 actions (with skip), 90 videos, 9 subjects -171 sequences by cycle analysis and segmentation	100%	Dynamic shapes together with LPP
[32], AVSS'03	Same as ours	88.9%	-
	Robustness test (4) with respect to the challenging factors		
Methods	Test sequences	The number of sequences correctly classified as 'walk' action	
[27], ICCV'05	10 walking sequences with different conditions	9	
Our method	Same above	8	
[32], AVSS'03	Same above	1	

clustering effect. This is because, by trying to preserve neighborhood structure in the embedding, LPP implicitly emphasizes the natural clusters in the data. This is a key ideal property for classification tasks. Although other methods may obtain similar results, their clustering effects are very poor, which naturally decreases the performance as the number of actions increases. Analysis of statistical significance also demonstrates the advantages of LPP and SLPP over other methods.

2) Comparison II—Comparison With Methods of Masoud and Papanikolopoulos and of Blank et al.: It is more meaningful and fair to make comparison of different algorithms on the same dataset. Here, we compared a related method described in [32], which uses linear PCA on the filtered images for obtaining low-dimensional activity description. The evaluation of [32] was carried out on a test dataset of eight actions and 168 sequences, and achieved a best recognition rate of 92.8% using the nearest centroid manifold distance, with the reduced dimension of 50; or 82% using the nearest manifold distance (as we use in this paper). Due to the unavailability of their database we are unable to test the proposed algorithm on their dataset. Therefore, we re-implement their method on the datasets [27] we used for both classification and the robustness test. The best recognition rate is 88.9% (see Table IV), which is far lower than our method's. In particular, with respect to robustness test, their method can only classify normal walking. This further demonstrates that PCA is sensitive to outliers and noise, and unsuitable to learn nonlinear activity manifolds. In Table IV, we also directly cite the best results reported in [27] using the same dataset (without the skipping action there, however, it has been found in our experiments that "skip" is easily confused for classification). Our results are comparable to those of Blank *et al.*, but our feature selection and extraction are apparently simpler than theirs.

3) Comparison III—Compendium of Results Reported: The lack of a common database (in face recognition, the FERET

TABLE V
SUMMARY OF SOME REPRESENTATIVE METHODS OF HUMAN MOTION RECOGNITION

Methods	Test dataset	Accuracy	Basic ideas
[23], MVA'05 Silhouette-based	15 basic actions (raise one or both hands, wave one or both hands, lower one or both hands, bend down, get up, raise foot, lower foot, sit down, stand up, squat, raise from squat, X-hopping), 5 persons	90%	SVM for posture classification and dHMMs for activity recognition
[24], PAMI'01 Silhouette-based	18 aerobic exercises, 1 subject training, 1 subject test	83.3%	Moment-based features of MEI and MHI templates of binary images
[28], ICCV'05 Silhouette-based	Walking, running, bending and picking, dancing, jump forward, turns, etc.	85%	CRF-based training/ recognition using 50d histogram of images features
[29] PHI'05 Silhouette-based	11 actions (lift right or left arm ahead or sideways, lift both arms ahead then sideways, drop or lift both arms sideways, lift right or left leg bend knee, lift right leg firm, jump) , 2 persons	93.9%	Fourier descriptors of motion history volumes from 6 calibrated cameras
[17], CVPR'05 Contour-based	12 actions, 28 sequences (2 dancing, 1 falling, 2 tennis strokes, 7 walking, 1 running, 2 kicking, 2 sit-down, 3 stand-up, 2 surrender, 2 hands-down, 4 aerobics actions)	85.7%	Differential geometric surface properties of STV derived from frame-to-frame contour correspondence
[13], PAMI'02 Model-based	8 activities (jumping, kneeling, picking up or putting down an object, running, sitting down, standing up, and walking), 5 people, 40 test videos	100%	A set of pose and velocity vectors for the major body parts (hands, legs and torso) plus indexing of hash tables
[7] 3DPVT'02 Model-based	3 periodic actions (stepping at same place, walking and running), 6 subjects, and 1 non-periodic reaching action in 8 different directions using right arm, 5 subjects.	80%	Movelet codewords representing shape, motion and occlusions of the 10 parts of human body model
[5], ICCV'03 Optical flow	Ballet, 16 classes, 2 males and 2 females; Tennis, 6 actions (swing, move left or right, move left or right and swing, stand), 2 subjects; Football, 8 actions (run left or right 45, run left or right, walk left or right, walk or run in/out)	87.5% 64.3% 65.4%	Motion descriptor based on blurred optical flow measurements plus normalized correlation
[22], ICPR'04 Local descriptor	6 actions (walking, jogging, running, boxing, hand waving and hand clapping), 25 subjects, a training set (8 persons), a validation set (8 persons) and a test set (9 persons).	76%	Local measurements in terms of spatiotemporal interest point, together with SVM classification
[19], VS-PETS'05 Local descriptor	The same as [22]	81%	Spatiotemporal interest point detector, and behaviour descriptor based on a histogram of the cuboid types
[15], ICCV'05 Trajectory-based	8 actions, 18 sequences (3 running, 3 bicycling, 3 sitting-down, 2 walking, 2 picking-up, 1 waving hands, 1 forehand stroke, 1 backhand stroke)	100%	13 landmark's trajectories together with dynamic epipolar geometry matching
[18], CVIU'99 Trajectory-based	4 actions (walking, marching, line-walking, and kicking while walking), 66 test sequences	82%	Trajectories of five body parts to obtain principle curves of actions

database is an example serving that comment well) and standardized evaluation methodology has been an apparent limitation in the development of action recognition algorithms. Although a large number of papers reported good recognition results on individual databases with different numbers and various categories of actions, they seldom made informed comparison among various different methods due to the real difficulties of making such quantitative comparison. Here, we simply list some representative studies in terms of their test datasets, approximate accuracies, and basic principles in Table V. To some extent, these methods reflect the latest and best work in human motion or action recognition.

From Table V, we can see that our method and its performance demonstrated here are comparable to others with respect to recognition rates with robustness. Moreover, we have demonstrated these qualities on a comparable dataset in terms of size and complexities. The simplicity and reliability of the extracted features, upon which our method is constructed, is also very competitive (that is, we do not use extraordinarily complex or high quality feature extraction). Yet our method has several advantages: 1) our method is very easy to understand and implement. It only analyzes binary shapes, without the requirement of video alignment and explicit 2-D or 3-D tracking; 2) our method avoids feature tracking, the computation of optical flow, and the extraction of gradient or intensity based features (and, hence, their complexities and brittleness); 3) our method avoids the problems of the computation complexity, and the difficulty of parameter selection, introduced in some approaches based on motion modeling using dynamic probabilistic networks such as HMMs, CRF and so on; and 4) our method has also the potential to cope with low quality video data, where other methods, especially those based on intensity features only, will fail.

VI. SUMMARY AND FUTURE WORK

Human action recognition has gained increasing interest in the computer vision community. Compared with other extensively studied topics such as human detection, tracking and recognition, human activity understanding is in its infancy due to the complexity and variety of actions. This paper has proposed a simple but effective method for human action recognition. The major core is based on low-dimensional embedding representations of dynamic silhouettes obtained from the action videos. Extensive experimental results have validated the powerful abilities of the proposed method.

Although the experiments have demonstrated that our methodology works effectively, further evaluation on a larger database, with multivaried actions, subjects and scenarios, needs to be carried out. Both shape and kinematics information derived from actions play important roles in human motion analysis [26]. Fusion of two cues is, thus, preferable for improving the accuracy and reliability. Most current work on action and motion interpretation remains rooted in view-dependent representations. Although there have been some attempts for this problem [14], [16], [29], they usually use the epipolar geometry between the views of two or more cameras to perform view invariant recognition. How to extract view-invariant motion features still remains challenging. We also plan to test our algorithm with a spatiotemporal extension to Isomap [46], or to augment other dimension reduction methods with temporal relation for dealing with the problem of manifold learning of dynamic data.

Acknowledgment

The authors would like to thank M. Blank at the Weizmann Institute for providing the dataset. They would also like to thank Associate Editor Prof. M. Samuelides for considering the paper and anonymous reviewers for their thorough review and constructive comments.

References

[1] D. Gavrila, "The visual analysis of human movement: A survey," *Comput. Vis. Image Understand.*, vol. 73, no. 1, pp. 82–98, 1999.

[2] C. Cedras and M. Shah, "Motion-based recognition: A survey," *Image Vis. Comput.*, vol. 13, no. 2, pp. 129–155, 1995.

[3] L. Wang, W. Hu, and T. Tan, "Recent developments in human motion analysis," *Pattern Recognit.*, vol. 36, no. 3, pp. 585–601, 2003.

[4] M. Black, "Explaining optical flow events with parameterized spatio-temporal models," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1999, vol. 1, pp. 1326–1332.

[5] A. Efros, A. Berg, G. Mori, and J. Malik, "Recognizing action at a distance," in *Proc. Int. Conf. Computer Vision*, 2003, vol. 2, pp. 726–733.

[6] R. Polana and R. Nelson, "Detection and recognition of periodic, nonrigid motion," *Int. J. Comput. Vis.*, vol. 23, no. 3, pp. 261–282, 1997.

[7] X. Feng and P. Perona, "Human action recognition by sequence of movelet codewords," in *Proc. Int. Symp. 3D Data Processing Visualization and Transmission*, 2002, pp. 717–723.

[8] Y. Sheikh and M. Shah, "Exploring the space of an action for human action recognition," in *Proc. Int. Conf. Computer Vision*, 2005, vol. 1, pp. 144–149.

[9] R. Green and L. Guan, "Quantifying and recognizing human movement patterns from monocular video images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 2, pp. 179–190, Feb. 2004.

[10] C. Bregler, "Learning and recognizing human dynamics in video sequences," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1997, pp. 568–574.

[11] A. Bissacco, A. Chiuso, Y. Ma, and S. Soatto, "Recognition of human gaits," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2001, vol. 2, pp. 52–57.

[12] A. Ali and J. Aggarwal, "Segmentation and recognition of continuous human activity," in *Proc. Int. Workshop on Detection and Recognition of Events in Video*, 2001, pp. 28–35.

[13] J. Ben-Arie, Z. Wang, P. Pandit, and S. Rajaram, "Human activity recognition using multidimensional indexing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 8, pp. 1091–1104, Aug. 2002.

[14] C. Rao and M. Shah, "View-invariance in action recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2001, vol. 2, pp. 316–321.

[15] A. Yilmaz and M. Shah, "Recognizing human actions in videos acquired by uncalibrated moving cameras," in *Proc. Int. Conf. Computer Vision*, 2005, vol. 1, pp. 150–157.

[16] V. Parameswaran and R. Chellappa, "View invariants for human action recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2003, vol. 2, pp. 610–613.

[17] A. Yilmaz and M. Shah, "Action sketch: A novel action representation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2005, vol. 1, pp. 984–989.

[18] Y. Yacoob and M. Black, "Parameterized modeling and recognition of activities," *Comput. Vis. Image Understand.*, vol. 73, no. 2, pp. 232–247, 1999.

[19] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," presented at the Int. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005.

[20] Y. Song, L. Goncalves, and P. Perona, "Unsupervised learning of human motion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 7, pp. 814–827, Jul. 2003.

[21] L. Zelnik-Manor and M. Irani, "Event-based analysis of video," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2001, vol. 2, pp. 123–130.

[22] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach," in *Proc. Int. Conf. Pattern Recognition*, 2004, vol. 3, pp. 32–36.

[23] V. Kellokumpu, M. Pietikainen, and J. Heikkila, "Human activity recognition using sequences of postures," presented at the IAPR Conf. Machine Vision Applications, 2005.

[24] A. Bobick and J. Davis, "The recognition of human movement using temporal templates," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 3, pp. 257–267, Mar. 2001.

[25] S. Carlsson and J. Sullivan, "Action recognition by shape matching to key frames," presented at the Int. Workshop on Models Versus Exemplars in Computer Vision, 2001.

[26] A. Veeraraghavan, A. Roy-Chowdhury, and R. Chellappa, "Role of shape and kinematics in human movement analysis," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2004, vol. 1, pp. 730–737.

[27] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Action as space-time shapes," in *Proc. Int. Conf. Computer Vision*, 2005, vol. 2, pp. 1395–1402.

[28] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas, "Conditional models for contextual human motion recognition," in *Proc. Int. Conf. Computer Vision*, 2005, vol. 2, pp. 1808–1815.

[29] D. Weinland, R. Ronfard, and E. Boyer, "Motion history volumes for free viewpoint action recognition," presented at the IEEE Workshop Modeling People and Human Interaction, 2005.

[30] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time sequential images using hidden Markov model," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1992, pp. 379–385.

[31] G. Johansson, "Visual perception of biological motion and a model for its analysis," *Percept. Psychophys.*, vol. 14, pp. 201–211, 1973.

[32] O. Masoud and N. Papanikolopoulos, "Recognizing human activities," in *Proc. Int. Conf. Advanced Video and Signal Based Surveillance*, 2003, pp. 157–162.

[33] C. Fanti, L. Zelnik-Manor, and P. Perona, "Hybrid models for human motion recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2005, vol. 1, pp. 1166–1173.

[34] I. T. Jolliffe*, Principal Component Analysis*. New York: Springer-Verlag, 1986.

[35] 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.

[36] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, pp. 2323–2326, 2000.

[37] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, pp. 2319–2323, 2000.

[38] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Proc. Int. Conf. Advances in Neural Information Processing Systems*, 2001, pp. 585–591.

[39] X. He and P. Niyogi, "Locality preserving projections," presented at the Int. Conf. Advances in Neural Information Processing Systems, 2003.

[40] A. Elgammal and C.-S. Lee, "Inferring 3D body pose from silhouettes using activity manifold learning," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2004, vol. 2, pp. 681–688.

[41] Q. Wang, G. Xu, and H. Ai, "Learning object intrinsic structure for robust visual tracking," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2003, vol. 2, pp. 227–233.

[42] C. Sminchisescu and A. Jepson, "Generative modeling for continuous non-linearly embedded visual inference," in *Proc. Int. Conf. Machine Learning*, 2004, pp. 140–147.

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

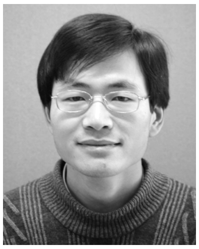
[44] C. Shan, S. Gong, and P. McOwan, "Appearance manifold of facial expression," in *Proc. Int. Workshop on Human Computer Interaction*, 2005, pp. 221–230.

[45] Y. Chang, C. Hu, and M. Turk, "Probabilistic expression analysis on manifolds," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2004, vol. 2, pp. 520–527.

[46] O. C. Jenkins and M. J. Mataric, "A spatiotemporal extension to Isomap nonlinear dimension reduction," in *Proc. Int. Conf. Machine Learning*, 2004, pp. 441–448.

[47] H. Breu, J. Gil, D. Kirkpatrick, and M. Werman, "Linear time Euclidean distance transform algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 5, pp. 529–533, May 1995.

[48] R. Cutler and L. Davis, "Robust real-time periodic motion detection, analysis, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 781–796, Aug. 2000.



Liang Wang received the B.Sc. degree in electrical engineering and the M.Sc. degree in video processing and multimedia communication from Anhui University, Hefei, China, in 1997 and 2000, respectively, and the Ph.D. degree in pattern recognition and intelligent systems from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2004.

From July 2004 to September 2005, he was with Imperial College London, London, U.K. From October 2005 to January 2007, he was with Monash University, Melbourne, Australia. Currently, he is with the University of Melbourne. He has published more than 30 papers on major international journals and conferences. His major research interests include computer vision, pattern recognition, digital image processing, analysis, machine learning, etc.

David Suter received the B.Sc. degree in applied mathematics and physics, the Grad. Dip. Comp., and the Ph.D. in computer science.

He was a Lecturer at La Trobe from 1988 to 1991 and a Senior Lecturer (1992), Associate Professor (2001), and Professor (2006–present) at Monash University, Melbourne, Australia. Currently, he is an Associate Editor for the *International Journal of Computer Vision* and for *Machine Vision and Applications* (having previously served as an Associate Editor for the *International Journal of Image and Graphics*). His research activities focus on topics such as motion estimation from images (including optic flow), structure from motion, image segmentation, biomedical image analysis, human motion capture and animation, visual tracking, activity detection and classification, face recognition, and the construction of building models from laser scan, and image data.