Supervised Spatio-Temporal Neighborhood Topology Learning for Action Recognition

Andy J Ma, Pong C Yuen, *Senior Member, IEEE,* Wilman W W Zou and Jianhuang Lai

***Abstract*—Supervised manifold learning has been successfully applied to action recognition, in which class label information could improve recognition performance. However, the learned manifold may not be able to well preserve both the local structure and global constraint of temporal labels in action sequences. To overcome this problem, this paper proposes a new supervised manifold learning algorithm namely supervised spatio-temporal neighborhood topology learning (SSTNTL) for action recognition. By analyzing the topological characteristics in the context of action recognition, we propose to construct the neighborhood topology using both supervised spatial and temporal pose correspondence information. Employing the locality preserving property in LPP, SSTNTL solves the generalized eigenvalue problem to obtain the best projections that not only separating data points from different classes, but also preserving local structures and temporal pose correspondence of sequences from the same class. Experimental results demonstrate that SSTNTL outperforms the manifold embedding methods with other topologies or local discriminant information. Moreover, compared with state-of-the-art action recognition algorithms, SSTNTL gives convincing performance for both human and gesture action recognition.**

***Index Terms*—Manifold learning, action recognition, supervised spatial, temporal pose correspondence, neighborhood topology learning.**

I. Introduction

ACTION recognition is an active research topic in computer vision and pattern recognition, due to a wide range of potential applications, such as intelligent video surveillance, perceptual interface and content-based video retrieval. While many algorithms and systems have been developed in the last decade [1] [2] [3], recognizing actions in videos still remains challenging. In action recognition, a key issue is to extract useful action information from raw video data. So far various approaches have been proposed to extract features from video sequences, such as key frame extraction [4] [5] [6], space-time interest point detection and description [7] [8] [9], key point trajectory based approaches [10] [11] [12], etc. However, recognizing actions using key frames lacks of motion information. And the interest point and trajectory based methods are based on local features, but do not consider the global constraints in space-time volumn.

A. J. Ma, P. C. Yuen and W. W. W. Zou are with the Department of Computer Science, Hong Kong Baptist University, Hong Kong. E-mail: {jhma, pcyuen, wwzou}@comp.hkbu.edu.hk

J. Lai is with the School of Information Science and Technology, Sun Yat-Sen University, Guangzhou 510006, China. He is also with the Guangdong Province Key Laboratory of Information Security, Guangzhou 510006, China. E-mail: stsljh@mail.sysu.edu.cn

Action sequences characterized by poses deforming continuously over time, can be regarded as data points on dynamic action manifolds. In [13] [14], Locality Preserving Projection (LPP) [15] was employed to learn and match the dynamic shape manifolds for human action recognition. Other than LPP, there are many manifold learning methods such as Isometric feature Map (Isomap) [16], Locally Linear Embedding (LLE) [17] and Laplacian Eigenmap (LE) [18], which aim to discover the intrinsic geometrical structure of a data manifold. However, these general manifold learning frameworks do not fully consider the important structure which lies in temporal labels.

In [19], Spatio-Temporal Isomap (ST-Isomap) is proposed to construct the local neighbors emphasizing the similarity between the temporal related blocks. Besides ST-Isomap, the temporal extension to Laplacian Eigenmap (TLE) is proposed in [20] and achieves better performance. Both ST-Isomap and TLE are unsupervised methods. That means class label information does not take into account. In turn, it may not be efficiently applied to supervised dimensionality reduction problem. Recently, supervised manifold embedding methods have been proposed such as Locality Sensitive Discriminant Analysis (LSDA) [21], Local Fisher Discriminant Analysis (LFDA) [22], Marginal Fisher Analysis (MFA) [23], and Supervised Locality Preserving Projection (SLPP) [24]. Although these methods show that label information is important, they do not take temporal cues into account.

In this context, Jia and Yeung [25] proposed a local spatio-temporal discriminant embedding (LSTDE) method to discover the local spatial and local temporal discriminant structures. LSTDE is derived by maximizing the inter-class variance and minimizing the intra-class variance based on the local spatio-temporal information. However, LSTDE does not consider the global constraints of the temporal order.

To overcome the limitations mentioned above, this paper proposes a new method to learn the manifold structure by using supervised spatial and temporal pose correspondence information from the topological aspect of manifold. As we know, topology is an important concept in the definition of a manifold [26]. By analyzing the topological base in existing methods for action recognition, we define a new topology by spatial and class label information, which is used to construct the supervised spatial (SS) neighborhood. However, construction of SS neighbors does not take full advantage of the temporal information. Thus, we further analyze the global constraint of temporal labels in action sequences, and develop the temporal pose correspondence (TPC) neighborhood. The supervised spatial information and

global constraint of temporal labels is fused by taking the union of SS and TPC neighbors. At last, the optimal linear projection functions preserving neighborhood relationship are obtained by solving a generalized eigenvalue problem. Our proposed method not only takes advantage of the supervised spatial distribution, but also temporal pose correspondence information to learn the neighborhood topology. We call it as supervised spatio-temporal neighborhood topology learning (SSTNTL). The contributions of this paper are four-fold.

• We develop a novel method to learn the manifold structure from the aspect of neighborhood topology. Topology is an important concept in manifold definition. For the same set of data, different types of topologies can be defined, and the manifolds therefore are different. By learning the topological base, the topology structure of the manifold is preserved.

• By analyzing the topology for action recognition, we combine the spatial distribution and label information to construct the SS neighborhood topology. Spatial distribution of the data provides the information about the local structure, while class label gives the discriminant information. By combining both information, SS neighborhood topology not only preserves local structure of data points from the same action, but also separates data points from different actions. In addition, we show that temporal adjacent neighborhood is contained in SS neighbors.

• We propose to construct the TPC neighborhood by taking advantage of the global constraint of temporal labels. Different sequences of the same action share similar poses deforming similarly over time. The corresponding poses in different sequences of the same action are thought to be in the same neighborhood, if background and appearance changes do not exist. Thus, we find the temporal pose correspondence by dynamic time warping (DTW) and construct the novel temporal neighborhood by this information. Since the temporal pose correspondence is obtained by the global constraint of temporal labels in action sequences of the same class, TPC neighborhood contains information about the global similarity between action sequences of the same class.

• We develop a new method, namely supervised spatio-temporal neighborhood topology learning (SSTNTL) for action recognition. Fusing SS and TPC neighborhoods, and then employing the locality preserving property in LPP [15], SSTNTL obtains the best projections by solving the generalized eigenvalue problem. Thus, SSTNTL not only learns the supervised spatial structures but also preserves temporal pose correspondence for action recognition.

The rest of this paper is organized as follows. Section II reviews some related works on manifold learning and action recognition. In section III, we present the proposed method for action recognition. Experimental results and conclusion are reported in Sections IV and V, respectively. The preliminary version of this paper which only considers the supervised spatial neighborhood topology learning (SNTL) has been reported in [27].

II. Relative Works

In this section, we first review existing manifold learning methods for action analysis. Then, we further elaborate LPP and SLPP, which are closely related to the proposed method.

A. Manifold Learning for Action Analysis

In the last few years, there has been increasing interest in analyzing actions using manifold embedding methods, since action data may lie on a low-dimensional manifold embedded in the high-dimensional image space [13]. In [28], the gait manifolds under different viewpoints were learned by LLE and used for viewpoint estimation, 3D configuration recovery, new instance synthesis, etc. In [29], manifold representations learned by LE were used for tracking and 3D motion reconstruction.

Along this direction, manifold learning was used for action recognition in [13] [14] [20] [25]. In [13] [14], LPP was used to learn the dynamic shape manifolds for action matching. Detailed introduction of LPP was presented in Section II-B. In order to take advantage of the temporal information, TLE was proposed in [20] for unsupervised dimensionality reduction of time series. TLE follows the LE framework and constructs the neighborhood by the adjacent temporal and repetition temporal information. As demonstrated in [20], TLE ensured the temporal coherence and improved the generalization properties of the embedded low dimensional spaces. Besides LPP and TLE, LSTDE [25] was proposed to discover the local spatio-temporal discriminant structures for human action recognition. LSTDE looks for the projections by iteration from three aspects: minimizing the Euclidean distances between close data points of the same action class, maximizing the Euclidean distances between close data points of different classes, and maximizing the principal angles between video segments of close data points from different classes. Experimental results demonstrated that LSTDE can improve the recognition performance over some representative manifold embedding methods [25].

B. Reviews of Locality Preserving Projection and Its Supervised Version

Locality preserving projection (LPP) [15] is a linear approximation of the nonlinear Laplacian eigenmap (LE) [18]. Suppose $x_1, x_2, \cdots, x_N$ are N samples which lie on a manifold in $\mathbb{R}^d$. The problem of LPP is to find a transformation matrix P which best preserves the neighborhood relationship of the manifold containing $x_1, x_2, \cdots, x_N$. Suppose $x_1, x_2, \cdots, x_N$ are represented by $z_1, z_2, \cdots, z_N$ in $\mathbb{R}^l$ ($l \ll d$), where $z_n = P^T x_n$. The algorithmic procedure of LPP is summarized as follows [15],

1. Constructing the adjacency graph: Let G be a graph with N nodes. An edge is connected nodes i and j, if $x_i \in \mathcal{N}(x_j)$ or $x_j \in \mathcal{N}(x_i)$, where $\mathcal{N}(x_i)$ denotes a small neighborhood of x_i. There are two ways to define $\mathcal{N}(x_i)$: i) k-nearest neighbors (k-NN): $\mathcal{N}(x_i)$ is made up of data points among the k nearest neighbors of x_i and ii) ε-neighborhood: $\mathcal{N}(x_i) = \{x_n | \|x_n - x_i\| < \varepsilon\}$.

2. Choosing the weights: There are two variations to choose weights, i.e. simple minded, $w_{ij} = 1$, or heat kernel, $w_{ij} = e^{-\|x_i - x_j\|^2/t}$, $t \in \mathbb{R}$, if and only if nodes i and j are connected by an edge; $w_{ij} = 0$, otherwise.

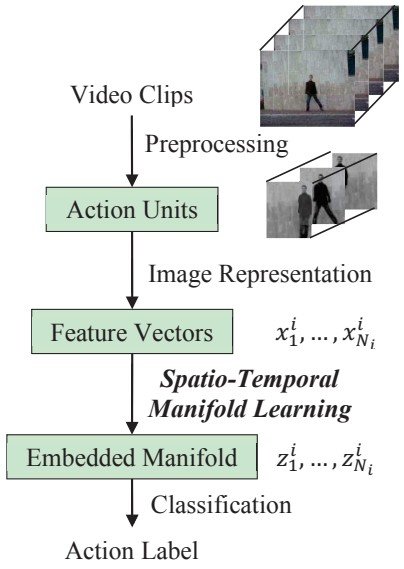


Fig. 1. Manifold learning based action recognition framework.

3. Eigenmap: Compute the eigenvectors and eigenvalues for the generalized eigenvalue problem:

$$XLX^T e = \lambda XDX^T e \tag{1}$$

where $X = (x_1, x_2, \cdots, x_N)$, D is a diagonal matrix with $D_{ii} = \sum_j w_{ij}$ and $L = D - W$ is the Laplacian matrix.

Let column vectors $e_1, e_2, \cdots, e_l$ be the eigenvectors of (1), ordered according to their eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_l$. Thus, the projection matrix and the embedding are given by $P = (e_1, e_2, \cdots, e_l)$ and $z_n = P^T x_n$.

On the other hand, a supervised version of LPP (SLPP) is proposed in [24]. SLPP is based on the framework of LPP. In SLPP, the adjacency graph is constructed by class label information, i.e. an edge is connected nodes i and j, if x_i and x_j are from the same class. Except for this, the other procedure is the same as that in LPP.

III. Supervised Spatio-Temporal Neighborhood Topology Learning

The block diagram of the whole action recognition method is shown in Fig. 1. After preprocessing the videos, a sequence of images containing the regions of interest are obtained. For non-periodic actions, the whole sequences with segmented region of interest are denoted as action units and used for further processing. For periodic actions, one cycle of action is extracted by period detection method and denoted as an action unit similar to those of non-periodic actions. Each action unit is then represented by a set of feature vectors. Denote the feature vectors by chronological order for action unit A as $x_1, \cdots, x_{N_A}$, where N_A is the number of segmented images in A. After that, each feature representation x_n is mapped to the embedded manifold by the proposed spatio-temporal manifold learning method. At last, set-based classifier is employed on the sequence $z_1, \cdots, z_{N_A}$, and the action label is obtained.

As discussed in Section II-B, the major difference between LPP and SLPP is the adjacency graph. In this section, we show that the adjacency graph is intimately related to the topological base from mathematical perspective. As we know, topology is an important concept in the definition of a manifold, so we will first give a topological analysis in the context of action recognition. Then, we propose to construct the neighborhood topology by the supervised spatial as well as temporal pose correspondence information. Finally, employing the locality preserving property in LPP, we propose the SSTNTL for action recognition.

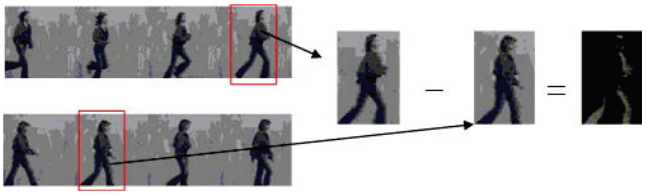


Fig. 2. Similar pose in two different actions.

A. Topological Analysis for Action Recognition

In action recognition, actions are regarded as data points on a manifold. Supposes there are c classes of actions and data points from all the actions lie on a smooth and compact manifold. Different actions may be close to each other on the manifold, because different actions share similar poses. This is illustrated in Fig. 2, which shows two actions, namely "run" and "walk". We can see that poses (frames) indicated with red rectangle in "run" and "walk" actions are similar. As such, certain data points from these two actions will be close together. This means sequences representing these two actions may be close with each other. In this case, the two actions in Fig. 2 can be represented by data points (diamond and circle represent different actions) as shown in Fig. 3(a). Denote the two actions as A_i and A_j. And let $A = A_i \cup A_j$. Usually, the topology defined on A is induced by the Euclidean topology. Denote this topology as τ_e. Since topology is too abstract to understand, we investigate the topological base [30] instead. In mathematics, if $\tau = \overline{\mathscr{B}}$, where $\overline{\mathscr{B}}$ represents a family of sets generated by the family of sets $\mathscr{B}$, i.e. $\overline{\mathscr{B}} = \{U \subset X | \forall x \in U, \exists B \in \mathscr{B}, s.t. x \in B \subset U\}$, $\mathscr{B}$ is called the topological base of the topological space (A, τ). With this representation, τ_e can be written as

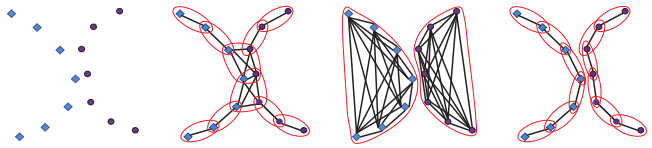
$$\tau_e = \overline{\mathscr{B}_e}, \quad \mathscr{B}_e = \{A \cap B(x_n, \varepsilon) | x_n \in A, \varepsilon > 0\} \tag{2}$$

where $B(x_n, \varepsilon)$ is a ball centered at x_n and with radius ε.

The topology in LPP is constructed with fixed ε in τ_e, which can be written as

$$\tau_{\text{LPP}} = \overline{\mathscr{B}_{\text{LPP}}}, \quad \mathscr{B}_{\text{LPP}} = \{A \cap B(x_n, \varepsilon_{\text{LPP}}) | x_n \in A\} \tag{3}$$

where ε_{LPP} is the parameter for the ε-neighborhood. For suitable parameter ε_{LPP}, the topological base in LPP can be represented by ovals in Fig. 3(b). Based on the topological base, the adjacency graph in LPP is constructed by putting a link between any two points in the same oval, which is also shown in Fig. 3(b). From Fig. 3(b), we can see that close points from two different actions are connected together, so



(a) (b) (c) (d)

Fig. 3. (a) Visualization of two close actions. Visualization of the topological base and adjacency graphs in (b) LPP, (c) SLPP and (d) the proposed supervised spatial method.

projections learned by LPP will map these points into low-dimensional space as close as possible. However, this is not what we want for classification.

Topologies defined on A can have many possible variations. Besides the Euclidean topology, in SLPP, the topology is given by

$$\tau_{\text{SLPP}} = \overline{\mathscr{B}_{\text{SLPP}}}, \quad \mathscr{B}_{\text{SLPP}} = \{A_i, A_j\} \tag{4}$$

The topological base and adjacency graph in SLPP are shown as ovals and edges in Fig. 3(c). Since SLPP only consider the class information, every two points from the same class are connected by an edge on the graph in Fig. 3(c). Therefore, the topology in SLPP hardly contains any local information of the data.

B. Supervised Spatial Neighborhood Topology Construction

According to the analysis in the above section, we can see that the topological base plays an important role in both LPP and SLPP. With spatial distribution and class label information of the data, the topological bases $\mathscr{B}_{\text{LPP}}$ in LPP and $\mathscr{B}_{\text{SLPP}}$ SLPP can be constructed. However, $\mathscr{B}_{\text{LPP}}$ cannot separate data points from the same class, and $\mathscr{B}_{\text{SLPP}}$ hardly contains any local information. In order to ensure that the topological base not only preserves local structure, but also be discriminative, we take the intersection of $\mathscr{B}_{\text{LPP}}$ and $\mathscr{B}_{\text{SLPP}}$ and define the supervised spatial topological base as

$$\mathscr{B}_{\text{SS}} = \{B_{\text{LPP}} \cap B_{\text{SLPP}} | B_{\text{LPP}} \in \mathscr{B}_{\text{LPP}}, B_{\text{SLPP}} \in \mathscr{B}_{\text{SLPP}}\} \tag{5}$$

Since $B_{\text{LPP}} = A \cap B(x_n, \varepsilon_{\text{LPP}})$ and $B_{\text{SLPP}} = A_i$ or A_j, $\mathscr{B}_{\text{SS}}$ in (6) can be rewritten and the novel topology can be defined as

$$\begin{aligned} \tau_{\text{SS}} &= \overline{\mathscr{B}_{\text{SS}}}, \\ \mathscr{B}_{\text{SS}} &= \{A_k \cap B(x_n, \varepsilon_{\text{SS}}) | x_n \in A_k, k = i \text{ or } j\} \end{aligned} \tag{6}$$

The topological base $\mathscr{B}_{\text{SS}}$ with (6) and adjacency graph constructed by $\mathscr{B}_{\text{SS}}$ are shown in Fig. 3(d). In this new topology and adjacency graph, A can be separated as two connected components, A_i and A_j. In addition, close points from the same action are connected by an edge on the graph. Therefore, the proposed supervised spatial topology τ_{SS} not only preserves local structure of data points from the same class but also separates data points from different classes.

Besides spatial and label information, the supervised spatial topology τ_{SS} contains temporal adjacent information as well. As shown in Fig. 4, action sequences are characterized by poses deforming continuously over time. In mathematics, the

Fig. 4. Visualization of the temporal continuity in an action sequence.

Algorithm 1. Constructing $\mathcal{N}_{\text{SS}}$

Input: Training samples $x_1, \cdots, x_N$;
Corresponding labels $y_1, \cdots, y_N$;
Percentage parameter a^{SS};

Output: Supervised spatial neighborhood $\mathcal{N}_{\text{SS}}$;

for $i = 1, \cdots, c$
  Compute number of samples for class i, N_i;
  Calculate k-NN parameter $k_i^{\text{SS}} = a^{\text{SS}} N_i$
endfor;
for $n = 1, \cdots, N$
  Select the $k_{y_n}^{\text{SS}}$ nearest samples respect to x_n from class y_n as $\mathcal{N}_{\text{SS}}(x_n)$;
endfor;
return $\mathcal{N}_{\text{SS}}(x_1), \cdots, \mathcal{N}_{\text{SS}}(x_N)$.

Fig. 5. Algorithm 1. Construction of the Supervised Spatial neighborhood.

temporal continuity means that for suitable ε_{SS}, there exists a positive integer δ, s.t. $\|x_{n+t} - x_n\| \leq \varepsilon_{\text{SS}}$, when $|t| \leq \delta$. By the definition of the supervised spatial topological base, it has

$$x_{n+t} \in B_{\text{SS}} \in \mathscr{B}_{\text{SS}}, \quad \text{when } |t| \leq \delta \tag{7}$$

This equation means that the temporal adjacent neighbors as indicated by red ovals in Fig. 4 are contained in the supervised spatial neighbors. Moreover, the construction of $\mathscr{B}_{\text{SS}}$ avoids the non-trivial problem of selection of the adjacent parameter, which is another advantage of the proposed method.

In practice, it is difficult to choose an optimal ε_{SS} when using ε neighborhood to construct the adjacency graph. Therefore, we construct the graph using k-NN. However, the number of data points from different classes may not be the same, so we choose a percentage parameter a^{SS} instead of k to construct the neighborhood of each data point. The algorithmic procedure to construct the supervised spatial neighborhood $\mathcal{N}_{\text{SS}}(x_n)$ for each x_n is stated in Fig. 5.

C. Temporal Pose Correspondence Neighborhood Topology Construction

Although the supervised spatial neighborhood $\mathcal{N}_{\text{SS}}$ contains temporal adjacent neighbors as mentioned in the above section, the construction of $\mathcal{N}_{\text{SS}}$ still does not take full advantage of the temporal information. If we consider different sequences of the same action "bend" as shown in Fig. 6, it can be observed that different sequences share the similar poses deforming similarly over time. However, if the neighborhood is constructed only by spatial information, the corresponding poses in different sequences of the same actions may not be close to each other, due to the background and appearance changes. Thus, the

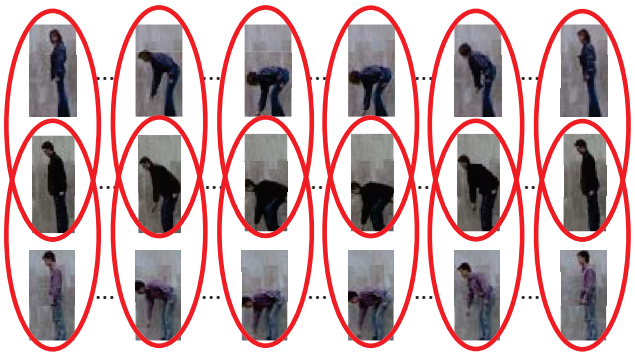


Fig. 6. Topological visualization of the TPC neighborhood.

temporal pose correspondence (TPC) between sequences of the same action may not be discovered by the supervised spatial neighbors. In this context, we propose to construct the neighborhood by the temporal pose correspondence directly.

Denote two action units of the same class as $A_{i_1} = \{x_1^{i_1}, \cdots, x_{N_{i_1}}^{i_1}\}$ and $A_{i_2} = \{x_1^{i_2}, \cdots, x_{N_{i_2}}^{i_2}\}$. Suppose each feature vector for the motion region $x_n^{i_k}$ in the action sequence A_{i_k} is corresponding to an underlying action pose $u_n^{i_k}$ for $k = 1$ or 2. In this context, the pose relationship which may vary in speed can be found out by employing Dynamic Time Warping (DTW) [31] on the underlying pose sequences $u_1^{i_1}, \cdots, u_{N_{i_1}}^{i_1}$ and $u_1^{i_2}, \cdots, u_{N_{i_2}}^{i_2}$. Denote the warping function as

$$\mathscr{F} = [(f_{i_1}(1), f_{i_2}(1)), \cdots, (f_{i_1}(T), f_{i_2}(T))] \tag{8}$$

where $1 \leq f_{i_k}(t) \leq N_{i_k}$, $f_{i_k}(t-1) \leq f_{i_k}(t)$ for $k = 1$ or 2, and T is integer, s.t. $T \geq \max\{N_{i_1}, N_{i_2}\}$. DTW finds the optimal warping function $\mathscr{F}^*$ by solving the following optimization problem

$$\min_{\mathscr{F}} \sum_{t=1}^{T} \omega(t) \| u_{f_{i_1}(t)}^{i_1} - u_{f_{i_2}(t)}^{i_2} \| \tag{9}$$

The distance between corresponding poses $u_{f_{i_1}^*(t)}^{i_1}$ and $u_{f_{i_2}^*(t)}^{i_2}$ must be small, though the distance between original feature vectors $x_{f_{i_1}^*(t)}^{i_1}$ and $x_{f_{i_2}^*(t)}^{i_2}$ may be large due to background and appearance changes. However, underlying poses $u_1^{i_k}, \cdots, u_{N_{i_k}}^{i_k}$ for $k = 1$ or 2 are unknown, so the pose relationship cannot be obtained by directly solving optimization problem (9).

Alternatively, we model the relationship between feature vector $x_n^{i_k}$ and underlying pose $u_n^{i_k}$ as follows. Suppose there is a projection for each sequence, which maps the feature vector subtracted by the background vector to the underlying pose, i.e.

$$u_n^{i_k} = Q_{i_k}(x_n^{i_k} - b_{i_k}) \tag{10}$$

In equation (10), vector b_{i_k} and projection matrix Q_{i_k} model the background and appearance changes in each action unit, respectively. Substituting $u_n^{i_k}$ with (10) into (9), the optimization function becomes

$$\sum_{t=1}^{T} \omega(t) \| Q_{i_1}(x_{f_{i_1}(t)}^{i_1} - b_{i_1}) - Q_{i_2}(x_{f_{i_2}(t)}^{i_2} - b_{i_2}) \| \tag{11}$$

Since vector b_{i_k} and matrix Q_{i_k} for $k = 1$ or 2 are unknown variables in optimization function (11), we minimize the upper bound given by (12) instead.

$$\begin{aligned} &\|Q_{i_1}\| \sum_{t=1}^{T} \omega(t) \| x_{f_{i_1}(t)}^{i_1} - x_{f_{i_2}(t)}^{i_2} \| \\ &+ \|Q_{i_1} - Q_{i_2}\| \sum_{t=1}^{T} \omega(t) \| x_{f_{i_2}(t)}^{i_2} \| \\ &+ \|Q_{i_2} b_{i_2} - Q_{i_1} b_{i_1}\| \sum_{t=1}^{T} \omega(t) \end{aligned} \tag{12}$$

Since $\sum_{t=1}^{T} \omega(t)$ is a constant respect to t and feature vectors $x_1^{i_2}, \cdots, x_{N_{i_2}}^{i_2}$ are normalized, minimizing objective function (12) is equivalent to solving the following optimization problem

$$\min_{\mathscr{F}} \sum_{t=1}^{T} \omega(t) \| x_{f_{i_1}(t)}^{i_1} - x_{f_{i_2}(t)}^{i_2} \| \tag{13}$$

This means that DTW performing on feature sequences $x_1^{i_1}, \cdots, x_{N_{i_1}}^{i_1}$ and $x_1^{i_2}, \cdots, x_{N_{i_2}}^{i_2}$ is equivalent to finding the minimum upper bound of the optimal pose matching (9).

The optimal warping function $\mathscr{F}^*$ obtained by solving (13) with DTW gives the correspondence between similar poses in different sequences of the same action. However, the difference between the corresponding frames may be very large due to the background and appearance changes. Consequently, if corresponding frames of the same pose in different sequences are in the same neighborhood, the manifold structure may be destroyed. Thus, neighboring corresponding frames are selected as the base of the temporal pose correspondence neighborhood topology, i.e.

$$\begin{aligned} \tau_{\text{TPC}} &= \overline{\mathscr{B}_{\text{TPC}}}, \\ \mathscr{B}_{\text{TPC}} &= \{\mathcal{C}(x_n) \cap B(x_n, \varepsilon_{\text{TPC}}) | x_n \in X\} \end{aligned} \tag{14}$$

where X is the universal set and $\mathcal{C}(x_n)$ denotes the set containing poses in different sequences corresponding to x_n. The topological base constructed by the temporal pose correspondence is indicated by the red ovals in Fig. 6.

Similarly, the temporal pose correspondence neighborhood is constructed by k-NN. And a percentage parameter a^{TPC} is used instead of k. Denote M_i is the number of sequences for action i. The algorithmic procedure to construct the temporal pose correspondence neighborhood $\mathcal{N}_{\text{TPC}}(x_n)$ for each x_n is stated in Fig. 7.

D. Supervised Spatial and Temporal Pose Correspondence Neighborhood Topology Learning

As mentioned in Sections III-B and III-C, the topological base $\mathscr{B}_{\text{SS}}$ contains supervised spatial information, while $\mathscr{B}_{\text{TPC}}$ consists of temporal pose correspondence. In order to ensure that the topological base embodies both information, we take

Algorithm 2. Constructing $\mathcal{N}_{\text{TPC}}$

Input: Training sequences $A_1, \cdots, A_M$;
Corresponding labels $y_1, \cdots, y_M$;
Percentage parameter a^{TPC};

Output: Temporal pose correspondence neighborhood $\mathcal{N}_{\text{TPC}}$;

for $i = 1, \cdots, c$
  Find action sequences with label i;
  for $m = 1, \cdots, M_i$
    for $n = m + 1, \cdots, M_i$
      Obtain warping function $\mathcal{F}_{A_{i_m}, A_{i_n}}$ by solving (9) or (10);
    endfor;
  endfor;
endfor;
for $n = 1, \cdots, N$
  Construct corresponding set to x_n, $\mathcal{C}(x_n)$ by $\mathcal{F}$;
  Count the number of elements in $\mathcal{C}(x_n)$, $\#\big(\mathcal{C}(x_n)\big)$;
  Set $k_n^{\text{TPC}} = a^{\text{TPC}} * \#\big(\mathcal{C}(x_n)\big)$;
  Select the k_n^{TPC} nearest samples in $\mathcal{C}(x_n)$ as $\mathcal{N}_{\text{TPC}}(x_n)$;
endfor;
return $\mathcal{N}_{\text{TPC}}(x_1), \cdots, \mathcal{N}_{\text{TPC}}(x_N)$.

Fig. 7. Algorithm 2. Construction of the TPC neighborhood.

the union of $\mathscr{B}_{\text{SS}}$ and $\mathscr{B}_{\text{TPC}}$, and the fused topology namely, supervised spatio-temporal neighborhood topology is defined as

$$\begin{aligned} \tau_{\text{ST}} &= \overline{\mathscr{B}_{\text{ST}}}, \\ \mathscr{B}_{\text{ST}} &= \{B_{\text{ST}} | B_{\text{ST}} \in \mathscr{B}_{\text{SS}} \text{ or } B_{\text{ST}} \in \mathscr{B}_{\text{TPC}}\} \end{aligned} \tag{15}$$

Based on (15), the fused neighborhood of each x_n is given by

$$\mathcal{N}_{\text{ST}}(x_n) = \mathcal{N}_{\text{SS}}(x_n) \cup \mathcal{N}_{\text{TPC}}(x_n) \tag{16}$$

Suppose manifold $\mathscr{M}_{\text{ST}}$ is defined with the neighborhood topology τ_{ST}. In [18], it is shown that the optimal mapping f preserving locality on manifold $\mathscr{M}_{\text{ST}}$ is given by solving the following optimization problem

$$\arg \min_{f,\ s.t.\ \|f\|_{L^2(\mathscr{M}_{\text{ST}})}=1} \int_{\mathscr{M}_{\text{ST}}} \|\nabla f\|^2 \tag{17}$$

In order to avoid the out-of-sample problem, the mapping f is restricted to be linear. In [15], it is shown that the optimal linear projections preserving locality can be obtained by solving the following optimization problem,

$$\arg \min_{e,\ s.t.\ e^T X D X^T e=1} e^T X L X^T e \tag{18}$$

where L and D are the Laplacian and diagonal matrices as defined in Section II-B.

Since the Laplacian L and diagonal matrix D are sparse and with special structure, XLX^T and XDX^T can be computed

Algorithm 3. SSTNTL

Input: Training sequences $A_1, \cdots, A_M$;
Corresponding labels $y_1, \cdots, y_M$;
Parameters $a^{\text{SS}}, a^{\text{TPC}}, l$;

Output: Projection matrix $P = (e_1, \cdots, e_l)$;

Construct $\mathcal{N}_{\text{SS}}$ by Algorithm 1;
Construct $\mathcal{N}_{\text{TPC}}$ by Algorithm 2;
Construct $\mathcal{N}_{\text{ST}}$ by (12);
Set matrices $M_L = 0$ and $M_D = 0$;
for $i = 1, \cdots, c$
  Compute $M_L = M_L + X_i L_i X_i^T$ and $M_D = M_D + X_i D_i X_i^T$;
endfor;
Solve the eigenvalue problem $M_L e = \lambda M_D e$;
Sort the eigenvectors $e_1, \cdots, e_l$ by eigenvalues $\lambda_1 \leq \cdots \leq \lambda_l$;
return $P = (e_1, \cdots, e_l)$.

Fig. 8. Algorithm 3. The algorithmic procedure of SSTNTL.

by the following equations

$$\begin{aligned} XLX^T &= \begin{pmatrix} X_1 \cdots X_c \end{pmatrix} \begin{pmatrix} L_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & L_c \end{pmatrix} \begin{pmatrix} X_1^T \\ \vdots \\ X_c^T \end{pmatrix} \\ &= \sum_{i=1}^{c} X_i L_i X_i^T \end{aligned} \tag{19}$$

$$\begin{aligned} XDX^T &= \begin{pmatrix} X_1 \cdots X_c \end{pmatrix} \begin{pmatrix} D_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & D_c \end{pmatrix} \begin{pmatrix} X_1^T \\ \vdots \\ X_c^T \end{pmatrix} \\ &= \sum_{i=1}^{c} X_i D_i X_i^T \end{aligned} \tag{20}$$

At last, the algorithmic procedure of the proposed SSTNTL method is presented in Fig. 8.

IV. Experiments

In this section, we evaluate the proposed SSTNTL on five publicly available action databases: Weizmann [32], KTH [33], UCF sports [34], Hollywood [8] human action databases and Cambridge-Gesture database [35]. In the following sections, we first give a brief introduction to the experimental settings and classifier in Section IV-A. Then, the results on these five databases are reported in Section IV-B to Section IV-F, respectively. At last, we compare SSTNTL with and without TPC neighbors in Section IV-G.

A. Settings and Classifier

For the video databases used in our experiments, we process the videos as described in Fig. 1. Details about the preprocessing process and image features are discussed separately for each database in the following sections. After obtaining the image features, Principal Component Analysis (PCA) [36] is used for dimensionality reduction to avoid the singular matrix problem and improve the computational efficiency. The

Fig. 9. Example segmented images from videos in Weizmann database

dimension of PCA is determined by keeping around 90% energy.

As shown in Fig. 8, parameters a^{SS}, a^{TPC}, l need to be determined in SSTNTL. Since it is difficult and time-consuming to search the best combination of the three parameters simultaneously, we determine the parameters in a two-stage scheme. First, setting $l = 60$, a^{SS} and a^{TPC} are selected from $\{0.02, 0.04, \cdots, 0.2\}$ and $\{0.1, 0.2, \cdots, 0.9\}$ respectively. Second, the best dimension l is selected from two to the determined PCA dimension with step size two for fixed a^{SS} and a^{TPC}. On the other hand, the parameter selection procedure in other methods is similar to that in SSTNTL. The neighborhood parameter in LPP and LSDA is selected from the same set as a^{SS} in SSTNTL. Since LSTDE is time-consuming with large neighborhood, the neighborhood parameter in LSTDE is selected from $\{0.01, 0.02, 0.03\}$ instead. In order to avoid the extra parameter selection by the heat kernel weighting, the weight matrix is calculated by the simple minded method as mentioned in Section II-B.

After feature extraction, we use a nearest neighbor framework in the classification stage. Let A_q be a query action sequence. The class label of A_q is given by

$$y_q = y_{\arg\min_m d(P^T Z_q, P^T Z_m)} \tag{21}$$

where d measures the distance between embedded feature sequences $Z_q = \{z_1^q, \cdots, z_{N_q}^q\}$ and $Z_m = \{z_1^m, \cdots, z_{N_m}^m\}$. Since median Hausdorff distance gives more robust results as mentioned in [13], we define d as equation (22) in our experiments.

$$\begin{aligned} d(Z_q, Z_m) =& \operatorname*{median}_j \min_i \|z_i^q - z_j^m\| \\ &+ \operatorname*{median}_i \min_j \|z_i^q - z_j^m\| \end{aligned} \tag{22}$$

B. Results on Weizmann Human Action Database

Weizmann database [32] contains 93 videos from nine persons, each performing ten actions, i.e. "bend", "jumping jack", "jump in place on two legs", "jump forward on two legs", "galloping sideways", "skip", "run", "walk", "wave one hand", and "wave two hands". We follow the procedures in [37] [38] to extract region of interest and detect periodic motion cycles by the information saliency curve [39]. Several action units can be detected for one video clip, while one cycle per video is used in our experiments (refer to [37] [38] for details). Fig. 9 shows some example segmented images from the videos representing the ten actions in Weizmann database. The extracted images are normalized into 100×100 pixels and represented by feature vectors with dimension 10000.

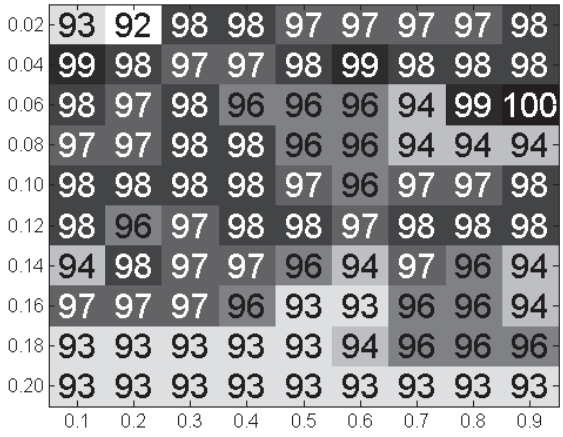


Fig. 10. Recognition accuracy (%) of SSTNTL with different values of a^{SS} and a^{TPC}, and fixed $l = 60$ on Weizmann database

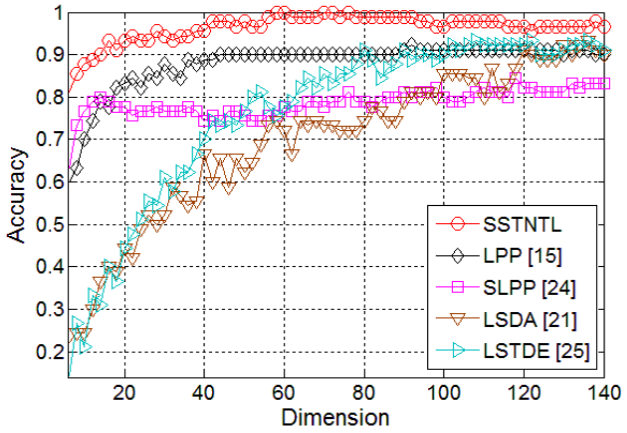


Fig. 11. Recognition accuracy with the best neighborhood parameter and different embedded dimensions on Weizmann database

Nine-fold cross validation protocol is employ to evaluate the proposed method. The average recognition rate is recorded.

The recognition accuracies of SSTNTL with different values of a^{SS} and a^{TPC}, and fixed $l = 60$ on this database are shown in Fig. 10. The darker element of the matrix in Fig. 10 represents higher recognition rate. The columns are the recognition rates with changing a^{SS} and specific a^{TPC}, while rows are recognition rates with changing a^{TPC} and specific a^{SS}. From Fig. 10, we can see that the highest accuracy of 100% is achieved in this database, when $a^{\text{SS}} = 0.06$ and $a^{\text{TPC}} = 0.9$. From the last row in Fig. 10, we observe that the recognition rates do not change when $a^{\text{SS}} = 0.2$ for different a^{TPC}. This may be due to the reason that the supervised spatial neighborhood $\mathcal{N}_{\text{SS}}$ already contains the temporal pose correspondence neighborhood $\mathcal{N}_{\text{TPC}}$, when $a^{\text{SS}} = 0.2$.

Fig. 11 shows the recognition accuracies of different methods with different dimension. From Fig. 11, we can see that the recognition rate of SSTNTL is higher than those of all the other methods in every dimensions. On the other hand, when the dimension is less than 20, SSTNTL, LPP and SLPP outperform LSDA and LSTDE a lot. This result suggests that the neighborhood topology learning methods with clear

Database (Feature)	Weizmann (Gray)	KTH (Gray)	KTH (Gist)	UCF (Gist)	HOHA (Gist)
SSTNTL	**100.0**	**79.6**	**94.4**	**91.3**	**44.5**
LPP [15]	92.2	72.2	89.4	88.6	29.2
SLPP [24]	84.4	70.8	88.4	86.6	27.0
LSDA [21]	93.3	75.0	83.3	84.6	29.4
LSTDE [25]	93.3	75.9	80.6	82.6	26.5

TABLE I
RECOGNITION ACCURACIES (%) OF MANIFOLD EMBEDDING METHODS ON DIFFERENT DATABASES

Method	Accuracy
SSTNTL	**100.0**
LTP [42]	**100.0**
BEL [6]	**100.0**
Sparse Representation [41]	98.9
Effective Codebook [40]	95.4
BoW [7]	90.0
3D Gradients [9]	84.3

TABLE II
RECOGNITION ACCURACY (%) COMPARISON WITH STATE-OF-THE-ART ACTION RECOGNITION SYSTEMS ON WEIZMANN DATABASE

Fig. 12. Example segmented images from videos in KTH database

Method (train/test setting: split)	Accuracy
SSTNTL	94.4
Product Manifold [46]	**96.0**
Neighbor Hierarchy [45]	94.5
Dense Trajectory [12]	94.2
BoD+MKGPC [47]	94.1
Effective Codebook [40]	92.6
Harris 3D + HoF [48]	92.1
Random Forest [49]	91.8
HoG + HoF [8]	91.8
3D Gradients [9]	91.4
LTP [42]	90.1

TABLE III
RECOGNITION ACCURACY (%) COMPARISON WITH STATE-OF-THE-ART ACTION RECOGNITION SYSTEMS ON KTH DATABASE WITH SPLIT SETTING

manifold interpretation are much better than those without clear one, when the dimension is small.

We compare SSTNTL with other manifold embedding methods on this database in the second column of Table I. SSTNTL achieves the best recognition accuracy of 100%, and outperform other embedding methods by 6.7%. On the other hand, we compare SSTNTL with state-of-the-art algorithms[1] in Table II. Our method outperforms space-time interest point based methods [7] [9] [40] as well as sparse representation approach [41]. Since Weizmann database is relatively simple, boosted exemplar learning (BEL) [6], local trinary patterns (LTP) [42] and the proposed method give the perfect performance. While BEL classifies actions based on exemplars and LTP extends the local binary patterns, these two methods do not consider the global constraint of temporal labels. Thus, our method outperforms them on the more challenging databases as shown in Table III, Table IV and Table VII.

C. Results on KTH Human Action Database

There are 25 subjects performing six actions under four scenarios in KTH database [33]. The six actions include "boxing", "hand clapping", "hand waving", "jogging", "running" and "walking". The four different scenarios are outdoors (S1), outdoors with scale variations (S2), outdoors with different clothes (S3) and indoors (S4). Regions of interest and motion cycles are extracted similar to the procedures in Weizmann database (refer to [37] [38] for details). Fig. 12 shows some example segmented images from the videos representing the six actions in KTH database. The extracted images are normalized into 100×100 pixels and represented by two kinds of image features. The first one converts gray-scale images into vectors with dimension 10000, while the second one computes the gist feature [43] on each extracted image. For the gist feature, the parameters about number of orientation per scale and number of blocks are selected from $\{4,8\}$. Three training/testing protocols (refer to [44] for details about the relationship between the performance and evaluation protocol) are used for evaluation and reported as follows.

Following the split setting in [33], the KTH database is divided into training (eight persons), validation (eight persons) and testing (nine persons) sets. Eight-fold cross-validation is performed on the training and validation sets to find optimal parameters and model. We first compare SSTNTL with other manifold embedding methods using gray-scale and gist features in the third and fourth columns of Table I, respectively. From Table I, we can see that SSTNTL outperforms other methods in both features, while the performances with gist are better than those with gray-scale feature. Then, We compare our method with state-of-the-art algorithms under the split setting in Table III. From Table III, we can see that our method outperforms most of the existing methods and is comparable with the hierarchical approach [45]. While Product Manifold (PM) [46] achieves the highest accuracy, the result is obtained by performing PM on action sequences with manually spatio-temporal alignment.

For the second protocol, we evaluate our method under leave-one-person-out (LOO) setting using all the four scenarios. The results are reported in Table IV. As show in IV, our method achieves the second highest accuracy of 96.3%. PM is better than our method by 0.7%. However, the result with PM is obtained by performing spatio-temporal alignment manually. While our method detects the regions of interest automatically, the result is also comparable.

At last, we compare SSTNTL with existing methods under each scenario with leave-one-person-out setting. The results are shown in Table V. SSTNTL outperforms others under scenarios of outdoor, outdoor with clothes variation and indoor. Especially, our method achieves 100% accuracy for the indoor scenario. This convinces that our method is very effective under controlled setting. On the other hand, Tracklet [11] and AFMKL [50] are better than our method under scenario two of outdoor with scale variation. The reason can be explained

[1]Please be noticed that the results of state-of-the-art methods are extracted from their papers and under the same setting.

Method (train/test setting: LOO)	Accuracy
SSTNTL	96.3
Product Manifold [46]	**97.0**
MoSIFT [44]	96.3
TCCA [35]	95.3
BEL [6]	95.3
Tracklet [11]	94.5
BoW [7]	83.3

TABLE IV
RECOGNITION ACCURACY (%) COMPARISON WITH STATE-OF-THE-ART ACTION RECOGNITION SYSTEMS ON KTH DATABASE WITH LOO SETTING

Method	S1	S2	S3	S4	Mean
SSTNTL	**98.0**	88.7	**96.0**	**100.0**	**95.7**
Tracklet [11]	**98.0**	**92.7**	92.0	96.7	94.8
AFMKL [50]	96.7	91.3	93.3	96.7	94.5
HSTM [51]	95.6	87.4	90.7	94.7	92.1

TABLE V
RECOGNITION ACCURACY (%) COMPARISON WITH STATE-OF-THE-ART ACTION RECOGNITION SYSTEMS ON KTH DATABASE UNDER EACH SCENARIO

as follows. The features used in SSTNTL are obtained by motion detection. Since motion detection under scenario two with scale variations is more challenged than that under the other three scenarios, the detected moving regions in scenario two must be less robust. Thus, local feature based methods, Tracklet and AFMKL, without motion detection outperform our method under scenario two.

D. Results on UCF Sports Database

UCF sports database [34] contains ten types of sports actions, including diving (Div), golf swinging (Gol), kicking (Kic), lifting (Lif), horseriding (Rid), running (Run), skateboarding (Ska), swinging at the bench (SwB), swinging at the high bar (SwH), and walking (Wal). There are totally 150 real videos in this database and each action class has different number of training samples. The bounding boxes provided in this database are used to segment the region of interest for the videos. Fig. 13 shows the images representing different actions after segmentation with the bounding boxes. Since results on KTH database show that gist descriptor provides better result than that of the image intensity, we extract the gist feature for each segmented image from the video clips in this database. Following the protocol in [34], the leave-one-sample-out cross validation setting is employed for evaluation.

The last but one column in Table I shows the recognition rates of different manifold embedding methods. The same conclusion can be drawn that SSTNTL outperforms other manifold embedding methods on this database. In Table R2, we further compare the proposed method with state-of-the-art action recognition systems. From Table VI, we can see that SSTNTL gives the highest accuracy 91.3% together with Augmented Features (context and appearance distribution features) multiple kernel learning (AFMKL) method [50]. And our method outperforms dense trajectory [12], space-time interest point (STIP) based methods [45] [48], sparse representation [41], and local trinary patterns (LTP) [42]. This is because the scene representations are discriminative for

Fig. 13. Example segmented images from videos in UCF sport database

Method	Accuracy
SSTNTL	**91.3**
AFMKL [50]	**91.3**
Dense Trajectory [12]	88.2
Neighbor Hierarchy [45]	87.3
Dense HoF [48]	85.6
Sparse Representation [41]	83.8
LTP [42]	79.2

TABLE VI
RECOGNITION ACCURACY (%) COMPARISON WITH STATE-OF-THE-ART ACTION RECOGNITION SYSTEMS ON UCF DATABASE

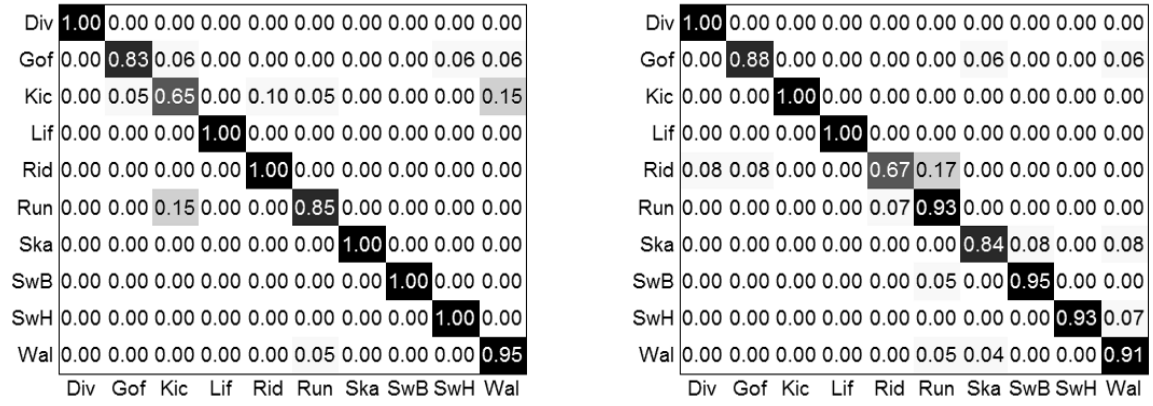
some sport actions. And our method not only captures the temporal variation, but also makes use of the discriminative representations of image frames. Fig. 14 shows the confusion matrices of SSTNTL and AFMKL, which both achieves the best recognition accuracy. From Fig. 14(a), we can see that our method achieves 100% accuracy for six in ten actions, including diving, lifting, riding horse, skateboarding, swinging at the bench and at the high bar. Comparing the confusion matrices in Fig. 14(a) and Fig. 14(b), the performance of our method is better than or equal to that of AFMKL in classifying seven actions.

E. Results on HOHA Database

Hollywood Human Action (HOHA) database [8] consists of eight types of actions, including Answer Phone (AnP), Get out of Car (GoC), Hand Shake (HS), Hug Person (HP), Kiss (Ki), Sit Down (SiD), Sit Up (SiU), and Stand Up (StU). Following the evaluation protocol in [8], 219 and 211 videos with "clean" annotations are used for training and testing, respectively. We manually segment regions of interest as shown in Fig. [8]. Gist features are computed for each segmented image from the video clips in this database as in KTH. Five-fold cross-validation is performed on the training data to select the best parameters for each action.

The average (Avg) values of the per-class precisions by manifold embedding methods on HOHA database are recorded in the last column of Table I. From Table I, we can see that the average precision of SSTNTL is 15% higher than those of other manifold embedding methods on this database. This convinces the effectiveness of SSTNTL in the more challenging database, comparing to other embedding methods.

In Table VII, we compare the proposed method with state-of-the-art action recognition systems in per-class precisions and their average. From Table VII, we can see that the average precision of SSTNTL is higher than those of the baseline method [8] by combining Histogram of Oriented Gradients (HoG) and Optical Flow (HoF), as well as recently proposed descriptors [9] [42] [11]. And our method best classifies actions including Get out of Car, Hand Shake and Sit Up. The performance of our method is comparable with, but lower than



(a) SSTNTL (b) AFMKL [50]

Fig. 14. Confusion matrices on UCF sport database

Fig. 15. Example bounding boxes and images from videos in HOHA database

Method	AnP	GoC	HS	HP	Ki	SiD	SiU	StU	Avg
SSTNTL	40.0	**62.5**	**44.4**	38.1	44.2	30.2	**44.4**	52.4	44.5
BoD + MKGPC [47]	**43.4**	46.8	44.1	**46.9**	**57.3**	**46.2**	38.4	**57.1**	**47.5**
HoG + HoF [8]	32.1	41.5	32.3	40.6	53.3	38.6	18.2	50.5	38.4
LTP [42]	35.1	32.0	33.8	28.3	57.6	36.2	13.1	58.3	36.8
Tracklet [11]	33.0	27.0	20.1	34.5	53.7	27.4	19.0	60.0	34.3
3D Gradients [9]	18.6	22.6	11.8	19.8	47.0	32.5	7.0	38.0	24.7

TABLE VII
RECOGNITION ACCURACY (%) COMPARISON WITH STATE-OF-THE-ART ACTION RECOGNITION SYSTEMS ON HOLLYWOOD DATABASE.

the multiple kernel Gaussian process classifier (MKGPC) [47]. This is attributed to the combination of bag-of-detector (BoD) scene descriptors, HoG, HoF and 3D Gradients for MKGPC, while our method is only based on one kind of feature.

F. Results on Hand Gesture Action Database

Cambridge-Gesture database [35] consists of 900 image sequences of nine hand gesture classes under five kinds of illuminations. The nine hand gesture actions include Flat-Leftward (FL), Flat-Rightward (FR), Flat-Contract (FC), Spread-Leftward (SL), Spread-Rightward (SR), Spread-Contract (SC), V-Shape-Leftward (VL), V-Shape-Rightward (VR), and V-Shape-Contract (VC). Each class and illumination set contains 20 sequences. Example images from this database are shown in Fig. IV-F. Gist feature vectors are extracted for each image as in KTH. Following the experimental protocol in [35] [46], 20 sequences per class with plain illumination

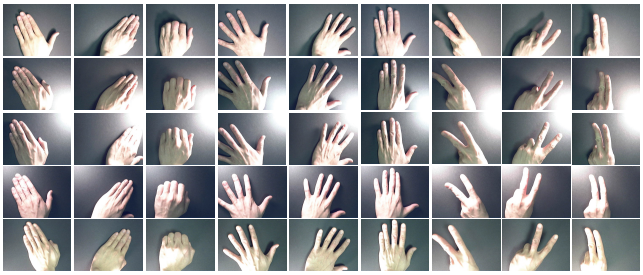
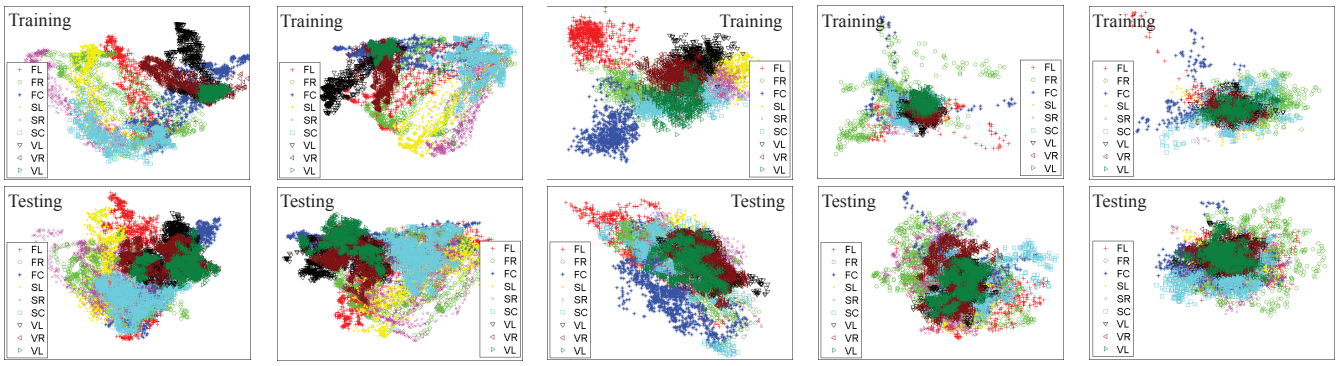


Fig. 16. Example images from sequences in Cambridge-Gesture database

is randomly partitioned into 10 sequences for training and the other 10 sequences for validation, while testing is performed in the data sets with the other four illuminations.

Recognition accuracies for different illumination sets on this database are presented in Table VIII. Compared with the man-



(a) SSTNTL (b) LPP [15] (c) SLPP [24] (d) LSDA [21] (e) LSTDE [25]

Fig. 17. 2D visualization of different methods with the training and testing data on Cambridge-Gesture database (better viewed in color printing)

Method	Set 1	Set 2	Set 3	Set 4	Mean
SSTNTL	**89**	**89**	85	**91**	**89**
LPP [15]	85	84	75	89	83
SLPP [24]	66	65	53	68	63
LSDA [21]	66	67	58	68	65
LSTDE [25]	68	66	69	80	71
TCCA [35]	81	81	78	86	82
PM [46]	**89**	86	**89**	87	88

TABLE VIII
RECOGNITION ACCURACY (%) OF DIFFERENT METHODS ON CAMBRIDGE-GESTURE DATABASE.

Method	Similarity 1	Similarity 2	Accuracy (dim=2)
SSTNTL	**0.941**	**0.698**	**22.2%**
LPP [15]	0.904	0.682	21.1%
SLPP [24]	0.578	0.485	18.3%
LSDA [21]	0.530	0.462	16.7%
LSTDE [25]	0.499	0.462	13.9%

TABLE IX
TWO SIMILARITY MEASURES OF THE 2D EMBEDDINGS FOR TRAINING AND TESTING DATA AND RECOGNITION ACCURACY WITH EMBEDDED DIMENSION TWO ON CAMBRIDGE-GESTURE DATABASE

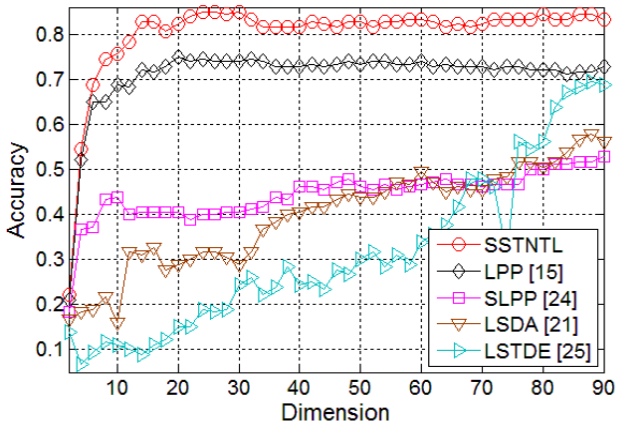
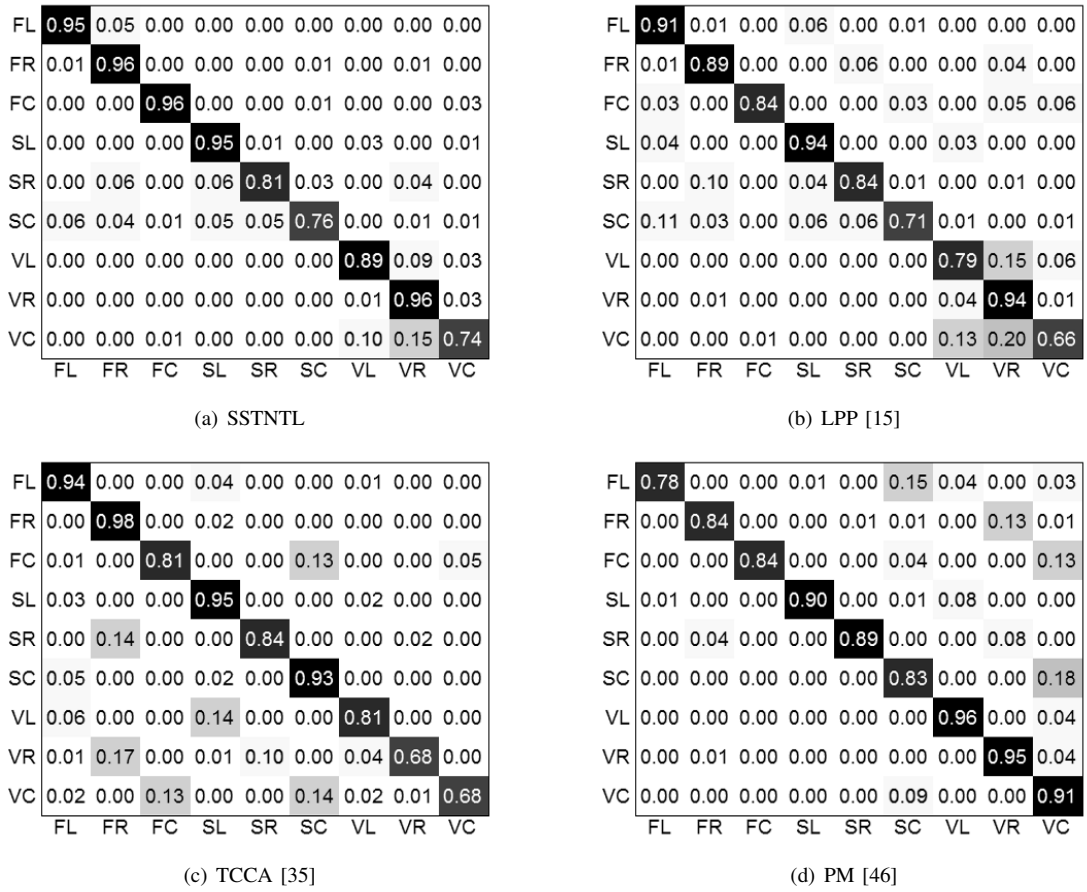


Fig. 18. Recognition accuracy with the best neighborhood parameter and different embedded dimensions on Cambridge-Gesture database.

ifold embedding methods, SSTNTL outperforms the others under the four different illumination sets. This convinces that SSTNTL is better than the topology learning methods with other topologies, as well as the local discriminative algorithms even with temporal information.

In order to further compare the generalization ability of the manifold embedding methods, the 2D visualizations of the training and testing data with illumination set three are shown in Fig. 17. For each method, the 2D embedding of the training data is presented in the upper row, while the one of the testing data is in the lower row. From Fig. 17(c), we can see that the 2D embedding of the testing data deviates to a different degree from that of the training by different method.

It is not easy to compare the difference between the 2D embeddings using visualization. As mentioned in [52], the median Hausdorff distance defined by (22) can be used to quantify the similarity between the 2D embeddings. Denote the training and testing embeddings as $Z^{\text{train}} = \{Z_1^{\text{train}}, \cdots, Z_c^{\text{train}}\}$ and $Z^{\text{test}} = \{Z_1^{\text{test}}, \cdots, Z_c^{\text{test}}\}$, where Z_i^{train} and Z_i^{test} are the embeddings for each class. With these notations, we first measure the shape similarity between the training and testing data as two data sets. And the first similarity index is given by $1/d(Z^{\text{train}}, Z^{\text{test}})$, where d is a distance function defined by (22). With label information, we further calculate the average difference between the training and testing data of each class. And the second index is defined by $c/\sum_{i=1}^{c} d(Z_i^{\text{train}}, Z_i^{\text{test}})$. The similarity quantification scores for these two indexes are shown in the second and third columns in Table IX. The similarity measures in Table IX are consistent with each other to different methods. However, different from the results in Table VIII, SLPP outperforms LSDA and LSTDE in terms of the two similarity measures. In this case, we further show the recognition rate with embedded dimension two and different dimension in the last column of Table IX and Fig. 18, respectively. From Table IX, we can see that when the embedded dimension is two, the recognition rates of the neighborhood topology learning methods are higher than that of LSDA and LSTDE, so the similarity results in Table IX are reasonable. On the other hand, Fig. 18 indicates that SSTNTL, LPP and SLPP outperform LSDA and LSTDE, when the dimension is less than 50, which is similar to the results in Weizmann database. And these results show that the generalization ability of the topology based methods is better than that of the local discriminability based methods, when the reduced dimension is low. And SSTNTL gives the best



(a) SSTNTL (b) LPP [15]

(c) TCCA [35] (d) PM [46]

Fig. 19. Confusion matrix on Hand Gesture confusion matrices

performance in this experiment.

We also compare the proposed method with state-of-the-art gesture action recognition algorithms. The Tensor Canonical Correlation Analysis (TCCA) [35] and Product Manifold (PM) [46] are used for comparison with the manifold embedding methods. From Table VIII, we can see that both PM and SSTNLT get the highest accuracy 89% for illumination set one. The product manifold method outperforms the others for set three, while SSTNTL gives the best performance for sets two and four. Compared with mean recognition rates, SSTNTL achieves the highest mean accuracy of 89% in this database.

At last, the confusion matrices of the best four algorithms are shown in Fig. 19. Compared the diagonal elements in the confusion matrices, SSTNTL outperforms LPP on eight actions, TCCA and product manifold method on five actions, respectively. This means SSTNTL outperforms the others on more than half of the gesture actions in this database. Overall speaking, SSTNTL is also performing well for hand gesture action recognition.

G. Comparing SSTNTL with and without TPC Neighbors

In the last experiment, we compare SSTNTL with and without temporal pose correspondence (TPC) neighbors constructed by DTW method. From Table X, we can see that SSTNTL with TPC neighbors outperforms that without TPC neighbors on different databases. This convince that the neighborhood constructed by DTW help to improve the performance.

Method	Weizmann	KTH	UCF	HOHA	Gesture
With TPC	**100.0**	**94.4**	**91.3**	**44.5**	**89**
Without TPC	95.6	90.3	89.3	32.3	80

TABLE X
RECOGNITION ACCURACIES (%) OF SSTNTL WITH AND WITHOUT TPC NEIGHBORHOOD ON DIFFERENT DATABASES

In order to further show that DTW can find the corresponding poses, we compare the supervised spatial neighborhood and TPC neighborhood on KTH database. Fig. IV-G shows the TPC neighbors which cannot be detected by the supervised spatial information. From Fig. IV-G, we can see that most of the TPC neighbors detected by DTW are similar poses to the referred images on the left hand side. This gives the reason why SSTNTL with TPC neighbors is better.

V. CONCLUSION

In this paper, we have proposed a novel manifold learning method, namely supervised spatio-temporal neighborhood topology learning (SSTNTL) for action classification. Starting from analyzing the topological characteristics in the context of action recognition, we proposed to construct the neighborhood topology from two aspects. First, the spatial distribution containing local structures, as well as the label information

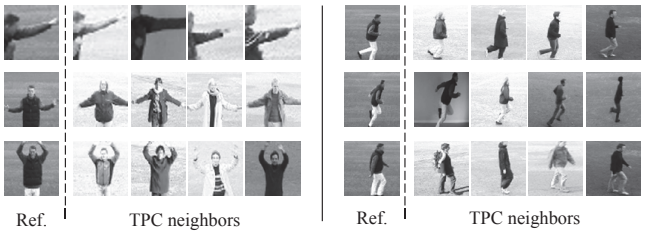


Fig. 20. Example TPC neighbors do not belong to the SS neighborhood

separating data points from different class are used to construct the supervised spatial topology. Second, the temporal pose correspondence neighborhood is constructed by discovering the global constraints of temporal labels in action sequences of the same class. These two neighborhood topologies are fused by taking the union of them. Based on the fused topology, SSTNTL employs the locality preserving property in LPP, and solves the generalized eigenvalue problem to obtain the best projections that not only separating data points from different classes, but also preserving local structures and global constraints of temporal labels.

The proposed algorithm is evaluated on five publicly available action video databases. Experimental results show that SSTNTL outperforms the neighborhood topology learning methods with other topologies, as well as the local discriminant algorithms even with temporal information. On the other hand, by analyzing the 2D visualizations of the training and testing data, we show that the generalization ability of the topology learning methods is better than that of the local discriminability based methods, when the reduced dimension is low. And SSTNTL shows the best generalization ability. In addition, compared with state-of-the-art action recognition algorithms, SSTNTL gives convincing performance for both human and gesture action recognition.

ACKNOWLEDGMENT

This project was partially supported by the Science Faculty Research Grant of Hong Kong Baptist University, National Science Foundation of China Research Grants 61128009 and 61172136. The authors would like to thank the associate editor and reviewers for their helpful comments which improved the quality of this paper. Finally, the authors would like to thank X. Lan's help for manually segmenting the video data in Hollywood Human Action database.

REFERENCES

[1] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473–1488, 2008.

[2] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 28, no. 6, pp. 976–990, 2010.

[3] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Computing Surveys*, vol. 43, no. 3, pp. 16:1–16:43, 2011.

[4] D. Weinland and E. Boyer, "Action recognition using exemplar-based embedding," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1–7, 2008.

[5] S. Baysal, M. C. Kurt, and P. Duygulu, "Recognizing human actions using key poses," *Proc. IEEE Int'l Conf. Pattern Recognition*, pp. 1727–1730, 2010.

[6] T. Zhang, J. Liu, C. X. Si Liu, and H. Lu, "Boosted exemplar learning for action recognition and annotation," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 21, no. 7, pp. 853–866, 2011.

[7] J. C. Niebles, H. Wang, and L. Fei-Fei, "Unsupervised learning of human action categories using spatial-temporal words," *Int'l J. Computer Vision*, vol. 79, no. 3, pp. 299–318, 2008.

[8] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.

[9] A. Kläser, M. Marszałek, and C. Schmid, "A spatio-temporal descriptor based on 3d-gradients," *Proc. British Machine Vision Conference*, 2008.

[10] R. Messing, C. Pal, and H. Kautz, "Activity recognition using the velocity histories of tracked keypoints," *Proc. IEEE Int'l Conf. Computer Vision*, pp. 104–111, 2009.

[11] M. Raptis and S. Soatto, "Tracklet descriptors for action modeling and video analysis," *Proc. European Conf. Computer Vision*, vol. 6311/2010, pp. 577–590, 2010.

[12] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 3169–3176, 2011.

[13] L. Wang and D. Suter, "Learning and matching of dynamic shape manifolds for human action recognition," *IEEE Trans. Image Processing*, vol. 16, no. 6, pp. 1646–1661, 2007.

[14] ——, "Visual learning and recognition of sequential data manifolds with applications to human movement analysis," *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 153–172, 2008.

[15] X. He and P. Niyogi, "Locality preserving projections," *Advances in Neural Information Processing Systems*, 2003.

[16] 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.

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

[18] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," *Advances in Neural Information Processing Systems*, 2001.

[19] O. C. Jenkins and M. J. Matarić, "A spatio-temporal extension to Isomap nonlinear dimension reduction," *Proc. Int'l Conf. Machine Learning*, pp. 441–448, 2004.

[20] M. Lewandowski, J. M. del Rincon, D. Makris, and J.-C. Nebel, "Temporal extension of laplacian eigenmaps for unsupervised dimensionality reduction of time series," *Proc. IEEE Int'l Conf. Pattern Recognition*, pp. 161–164, 2010.

[21] D. Cai, X. He, K. Zhou, J. Han, and H. Bao, "Locality sensitive discriminant analysis," *Proc. Int'l Joint Conf. Artificial intelligence*, 2007.

[22] M. Sugiyama, "Local fisher discriminant analysis for supervised dimensionality reduction," *Proc. Int'l Conf. Machine learning*, 2006.

[23] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40–51, 2007.

[24] Z. Zheng, F. Yang, W. Tan, J. Jia, and J. Yang, "Gabor feature-based face recognition using supervised locality preserving projection," *Signal Processing*, vol. 87, pp. 2473–2483, 2007.

[25] K. Jia and D.-Y. Yeung, "Human action recognition using local spatio-temporal discriminant embedding," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

[26] M. P. do Carmo, *Riemannian geometry*. Birkhauser, 1993.

[27] A. J. Ma, P. C. Yuen, W. Zou, and J.-H. Lai, "Supervised neighborhood topology learning for human action recognition," *IEEE Int'l Conf. Computer Vision Workshops*, pp. 476–481, 2009.

[28] A. Elgammal and C.-S. Lee, "Nonlinear manifold learning for dynamic shape and dynamic appearance," *Computer Vision and Image Understanding*, vol. 106, pp. 31–46, 2007.

[29] C. Sminchisescu and A. Jepson, "Generative modeling for continuous non-linearly embedded visual inference," *Proc. Int'l Conf. Machine learning*, 2004.

[30] J. L. Kelley, *General topology*. Birkhauser, 1975.

[31] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 2, no. 1, pp. 43–49, 1978.

[32] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 2247–2253, 2007.

[33] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local svm approach," *Proc. IEEE Int'l Conf. Pattern Recognition*, 2004.

[34] M. D. Rodriguez, J. Ahmed, and M. Shah, "Action mach a spatio-temporal maximum average correlation height filter for action recognition," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.

[35] T.-K. Kim and R. Cipolla, "Canonical correlation analysis of video volume tensors for action categorization and detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 8, pp. 1415–1428, 2009.

[36] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley, 2000.

[37] C. Liu and P. C. Yuen, "Human action recognition using boosted EigenActions," *Image and Vision Computing*, vol. 28, no. 5, pp. 825–835, 2010.

[38] ——, "A boosted co-training algorithm for human action recognition," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 21, no. 9, pp. 1203–1213, 2011.

[39] C. Liu, P. C. Yuen, and G. Qiu, "Object motion detection using information theoretic spatio-temporal saliency," *Pattern Recognition*, vol. 42, no. 11, pp. 2897–2906, 2009.

[40] L. Ballan, M. Bertini, A. D. Bimbo, L. Seidenari, and G. Serra, "Effective codebooks for human action categorization," *IEEE Int'l Conf. Computer Vision Workshops*, 2009.

[41] T. Guha and R. K. Ward, "Learning sparse representations for human action recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 34, no. 8, pp. 1576–1588, 2012.

[42] L. Yeffet and L. Wolf, "Local trinary patterns for human action recognition," *Proc. IEEE Int'l Conf. Computer Vision*, pp. 492–497, 2009.

[43] A. Oliva and A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope," *Int'l J. Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[44] Z. Gao, M. yu Chen, A. G. Hauptmann, and A. Cai, "Comparing evaluation protocols on the KTH dataset," *Human Behavior Understanding*, pp. 88–100, 2010.

[45] A. Kovashka and K. Grauman, "Learning a hierarchy of discriminative space-time neighborhood features for human action recognition," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 2046–2053, 2010.

[46] Y. M. Lui, J. R. Beveridge, and M. Kirby, "Action classification on product manifolds," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 833–839, 2010.

[47] D. Han, L. Bo, and C. Sminchisescu, "Selection and context for action recognition," *Proc. IEEE Int'l Conf. Computer Vision*, pp. 1933–1940, 2009.

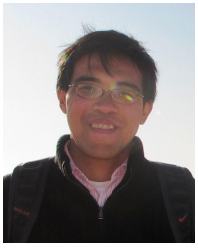
[48] H. Wang, M. M. Ullah, A. Kläser, I. Laptev, and C. Schmid, "Evaluation of local spatio-temporal features for action recognition," *Proc. British Machine Vision Conference*, 2009.

[49] G. Yu, A. Norberto, J. Yuan, and Z. Liu, "Fast action detection via discriminative random forest voting and top-k subvolume search," *IEEE Trans. Multimedia*, vol. 13, no. 3, pp. 507–517, 2011.

[50] X. Wu, D. Xu, L. Duan, and J. Luo, "Action recognition using context and appearance distribution features," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 489–496, 2011.

[51] H. Ning, T. X. Han, D. B. Walther, M. Liu, and T. S. Huang, "Hierarchical space-time model enabling efficient search for human actions," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 19, no. 6, pp. 808–820, 2009.

[52] M.-P. Dubuisson and A. K. Jain, "A modified hausdorff distance for object matching," *Proc. IAPR Int'l Conf. Pattern Recognition*, vol. 1, pp. 566–568, 1994.



Andy J Ma received his B.Sc. degree in mathematics and applied mathematics, and his M.Sc. degree in applied mathematics from Sun Yat-Sen University, Guangzhou, China, in 2007 and 2009, respectively. He is currently working toward his Ph.D. degree in the Department of Computer Science, Hong Kong Baptist University, Kowloon, Hong Kong.

His research interests include pattern recognition, computer vision and machine learning. And he is now focusing on information fusion, domain adaptation, manifold learning and action recognition.

Pong C Yuen (M'93-SM'11) received his B.Sc. degree in Electronic Engineering with first class honours in 1989 from City Polytechnic of Hong Kong, and his Ph.D. degree in Electrical and Electronic Engineering in 1993 from The University of Hong Kong. He joined the Hong Kong Baptist University in 1993 and, currently is a Professor and Head of the Department of Computer Science.

Dr. Yuen was a recipient of the University Fellowship to visit The University of Sydney in 1996. He was associated with the Laboratory of Imaging Science and Engineering, Department of Electrical Engineering. In 1998, Dr. Yuen spent a 6-month sabbatical leave in The University of Maryland Institute for Advanced Computer Studies (UMIACS), University of Maryland at college park. He was associated with the Computer Vision Laboratory, CFAR. From June 2005 to January 2006, he was a visiting professor in GRAVIR laboratory (GRAphics, VIsion and Robotics) of INRIA Rhone Alpes, France. He was associated with PRIMA Group. Dr. Yuen was the director of Croucher Advanced Study Institute (ASI) on biometric authentication in 2004 and the director of Croucher ASI on Biometric Security and Privacy in 2007.

Dr. Yuen has been actively involved in many international conferences as an organizing committee and/or technical program committee member. He was the track co-chair of International Conference on Pattern Recognition (ICPR) 2006 and is the program co-chair of IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS) 2012. Currently, Dr. Yuen is an editorial board member of Pattern Recognition and an associate editor of SPIE Journal of Electronic Imaging.

Dr. Yuen's current research interests include human face recognition, biometric security and privacy, and human activity recognition.

Wilman W. W. Zou received his B.Sc. degree in Mathematics in 2006 and his M.S. degree in Applied Mathematics in 2008 from Sun Yat-sen University, and his Ph.D. degree in Computer Science from Hong Kong Baptist University 2012. His research interests include machine learning, computer vision, image processing and data mining. He joined the Institute of Computational Theoretical Studies with Hong Kong Baptist University as Research Associate. Now his major research focuses on machine learning for computer vision, image processing, educational data mining, adaptive learning.

Jian-Huang Lai received his M.Sc. degree in applied mathematics in 1989 and his Ph.D. in mathematics in 1999 from SUN YAT-SEN University, China. He joined Sun Yat-sen University in 1989 as an Assistant Professor, where currently, he is a Professor with the Department of Automation of School of Information Science and Technology and vice dean of School of Information Science and Technology.

His current research interests are in the areas of digital image processing, pattern recognition, multimedia communication, wavelet and its applications. He has published over 100 scientific papers in the international journals and conferences on image processing and pattern recognition, e.g. IEEE TNN, IEEE TIP, IEEE TSMC (Part B), Pattern Recognition, ICCV, CVPR and ICDM. Prof. Lai serves as a standing member of the Image and Graphics Association of China and also serves as a standing director of the Image and Graphics Association of Guangdong.