Pattern Recognition 45 (2012) 3611–3624

Contents lists available at SciVerse ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Human action recognition based on graph-embedded spatio-temporal subspace

Chien-Chung Tseng [a], Ju-Chin Chen [b], Ching-Hsien Fang [a], Jenn-Jier James Lien [a,]*

[a] *Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan 70101, Taiwan, ROC*

[b] *Department of Computer Science and Information Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung 80778, Taiwan, ROC*

ARTICLE INFO

Article history:
Received 8 January 2011
Received in revised form
17 February 2012
Accepted 2 April 2012
Available online 14 April 2012

Keywords:
Human action recognition
Adaptive locality preserving projection
Large margin nearest neighbor

ABSTRACT

Human action recognition is an important issue in the pattern recognition field, with applications ranging from remote surveillance to the indexing of commercial video content. However, human actions are characterized by non-linear dynamics and are therefore not easily learned and recognized. Accordingly, this study proposes a silhouette-based human action recognition system in which a three-step procedure is used to construct an efficient discriminant spatio-temporal subspace for *k*-NN classification purposes. In the first step, an Adaptive Locality Preserving Projection (ALPP) method is proposed to obtain a low-dimensional spatial subspace in which the linearity in the local data structure is preserved. To resolve the problem of overlaps in the spatial subspace resulting from the ambiguity of the human body shape among different action classes, temporal data are extracted using a Non-base Central-Difference Action Vector (NCDAV) method. Finally, the Large Margin Nearest Neighbor (LMNN) metric learning method is applied to construct an efficient spatio-temporal subspace for classification purposes. The experimental results show that the proposed system accurately recognizes a variety of human actions in real time and outperforms most existing methods. In addition, a robustness test with noisy data indicates that our system is remarkably robust toward noise in the input images.

1. Introduction

Human action recognition has attracted significant interest in the computer vision community in recent decades and has spurred the development of a wide variety of applications, including video surveillance human–computer interaction and the analysis of sporting events. However, automatic human action recognition is highly challenging due to the non-stationary background of most video content, the ambiguity of the human body shape among different actions, and the existence of intra-class variations in the appearance, physical characteristics, and motion style of different human subjects.

One of the most important issues in realizing human action recognition systems is how to extract discriminant features from a video sequence. A suitable feature extraction and selection method can help system improve the recognition performance [1]. Moreover, the importance of feature extraction is not only for human action recognition, but for face recognition, age and gender classification [1–3]. In recent years, the feature of sparse representation selected via the optimization process attracts a lot of attention because it can provide promising recognition performance for faces under occlusions or other variants [2]. In addition, joint boosting [1] and the difference of Gaussian filter followed by Radon transform [3] are powerful approaches for feature selection. According to [4,5], most existing feature-based human action recognition systems are based on optical flow [6,7], space–time gradients [8,9], feature tracking models [10–15], or sparse spatio-temporal interest points [16–21]. However, space–time gradient methods and feature tracking models are highly sensitive to the quality of the input video and variations in the articulation of the human body or lighting conditions, respectively. Furthermore, the performance of recognition systems based on sparse interest points is inevitably limited due to the loss of global structure information [22]. Accordingly, the feasibility of performing human action recognition based on the human silhouette has attracted increasing interest in recent years [4,5,23–28]. Compared to the feature extraction methods proposed in [6–21], silhouette-based methods enable the construction of a sequence of space–time patterns that encode not only the spatial information of the body shape but also the temporal information of the global and local body parts [5].

In a video sequence containing human actions, the human silhouette in each frame can be represented by a vector in high-dimensional space and expected intrinsically to lie in a low-dimensional space embedded within this high-dimensional space [4]. Manifold learning methods, such as Isometric Feature

* Corresponding author. Tel.: +886 6 2757575; fax: +886 6 2747076.
E-mail address: jjlien@csie.ncku.edu.tw (J.-J. James Lien).

0031-3203/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.patcog.2012.04.002

Mapping (Isomap) [29], Locally Linear Embedding (LLE) [30], or Laplacian Eigenmaps [31], provide the means to identify the intrinsic geometrical structure of a database and thus facilitate the analysis of human action motions in compact low-dimensional space. For example, Elgammal and Lee [32] utilized the LLE method to infer 3D body poses from human silhouettes. Similarly, Wang and Suter [5] used a linear approximation to the LE method referred to as the Locality Preserving Projection (LPP) method [33] to establish low-dimensional feature representation for human silhouettes. LPP can extract the low-dimensional features of human silhouettes as a manifold by preserving both the intrinsic geometry and the local structure of the data via an adjacency undirected graph that incorporates the neighborhood information of the database [33]. However, LPP lacks clear rules for preserving linearity, and the rules for building the adjacency graph are not strict enough. As a result, the subspace obtained by LPP might be incompact; i.e., a data point in the subspace may be neighbored with other data points that are unrelated or similar to it.

Several supervised manifold learning methods based on class label information have been proposed in recent years, including Marginal Fisher Analysis (MFA) [9], supervised-LPP [5], and Locality Sensitive Discriminant Analysis (LSDA) [34]. The class label information makes possible the discovery of the local spatial discriminant structure and therefore enables the separation of images with different action classes. However, in addition to spatial information (e.g., silhouette shape), temporal information, such as the dynamic variation of the silhouette shape over a sequence of video frames, is also helpful in accomplishing reliable human action recognition systems. Accordingly, various researchers have incorporated temporal information into the action recognition process. For example, Jia and Yeung [4] proposed a local spatio-temporal subspace learning method (LSTDE) in which temporal subspaces associated with the data points in consecutive frames were constructed in such a way as to maximize both the discriminant structure in accordance with the class labels and the principal angles among the temporal subspaces of the different classes. Meanwhile, Wang and Suter [5] modeled the temporal evolution of an action motion as a sequence of projection points with associated temporal orders and used a hidden Markov model (HMM) to capture the structural and dynamic nature of the corresponding motion.

Consequently, the methods for human action recognition are mainly divided into two approaches: silhouette base and feature base. The first one utilizes human silhouettes as features which are commonly extracted by background subtraction [4,5,25]. The second one adopts motion or interesting information from human movement such as optical flow or interesting points [6,12,21]. Generally, the feature-based method does not use background subtraction during preprocessing. The objective in this study is to design a silhouette-based human action recognition system that can be integrated into an ordinary visual surveillance system with real-time moving object detection, classification and activity analysis capabilities. This system applies a three-step procedure to learn a discriminant spatio-temporal subspace for classification purposes. In the first step, a dimensionality reduction method designated as the Adaptive Locality Preserving Projection (ALPP) method is used to construct a compact spatial subspace. ALPP applies a modified graph construction process and a linearity measurement mechanism and will preserve the linearity in the local structure information while simultaneously reducing the dimensionality of the original database. Although ALPP can preserve the linearity and local structure information well, the ambiguity of the human body shape among different action classes will still result in an overlap of the silhouette information within the spatial subspace. Accordingly, in the second step of the proposed system, a Non-base Central-Difference Action Vector (NCDAV) method is used to extract the temporal data from the reduced spatial subspace to characterize the motion information in a temporal vector. It should be noted that NCDAV encodes the difference information between each consecutive frame with the base data. However, the base data is discarded in the temporal vector; otherwise, the temporal vector containing base data will result in overlapped distributions between different action types in the subspace. Finally, the Large Margin Nearest Neighbor (LMNN) metric learning method [22] is applied in the third step to construct a discriminant spatio-temporal subspace where the temporal vectors belonging to the same action class are clustered together while those associated with different action classes are separated by a margin. Having established the spatio-temporal subspace, human action recognition is achieved by utilizing a k-NN classifier to determine the action class of each input frame, and a majority voting mechanism is then applied to identify the action class of the entire input sequence.

In summary, there are three contributions in this study. The first one is that the proposed feature extraction method, named as Adaptive Locality Preserving Projection (ALPP), modifies the building of adjacency graph to solve the problems of LPP. The second one is the temporal information extraction, named as Non-base Central-Difference Action Vector (NCDAV). The designed temporal vector can reduce the corrupted effect by the base data and thus reduce the system degradation from the ambiguous and noise. The third one is the proposed framework itself, which is to extract discriminant features for action recognition. According to the properties of the three approaches, the proposed system is able to not only recognize human actions in real-time, but also considerably tolerate noise condition.

2. Learning process in a spatio-temporal subspace

Fig. 1 shows the overall framework of the proposed human action recognition system. As shown, the system comprises a learning process (see Fig. 1(a)) and a recognition process (see Fig. 1(b)). Assume that there are M training sequences $X=[X_1,X_2,...,X_M]$ and that each sequence comprises n_i frames. The learning process commences by extracting the human silhouettes from the training sequence frames using a background subtraction method. To reduce intra-class variations in the subject size, the silhouettes are centralized and normalized to a consistent size of $w \times h$ pixels. Therefore, each silhouette x_i can be represented by a D-dimensional vector ($D=w \times h$), and the training set has the form $X=[x_1,x_2,...,x_N] \in R^{D \times N}$, where $N=\sum_{i=1}^{M} n_i$.

Having constructed the training set, a three-step procedure is applied to analyze the spatial and temporal information of the silhouettes in the training sequences and to construct a spatio-temporal subspace for classification purposes. Section 2.1 describes the ALPP algorithm, which is used to reduce the dimensionality of the original spatial subspace while preserving the linearity in the local structure information. Section 2.2 describes the use of the NCDAV method for extracting temporal information from the training sequences and constructing the corresponding temporal vector. Finally, Section 2.3 describes the LMNN method, which is used to construct a discriminant spatio-temporal subspace for classification purposes.

2.1. Adaptive Locality Preserving Projection (ALPP) for dimensionality reduction

Human action sequences, as represented by a contiguous series of human silhouettes, can be viewed as a set of data points on non-linear manifolds in high-dimensional space. To reduce the computational cost of the learning process, it is desirable to

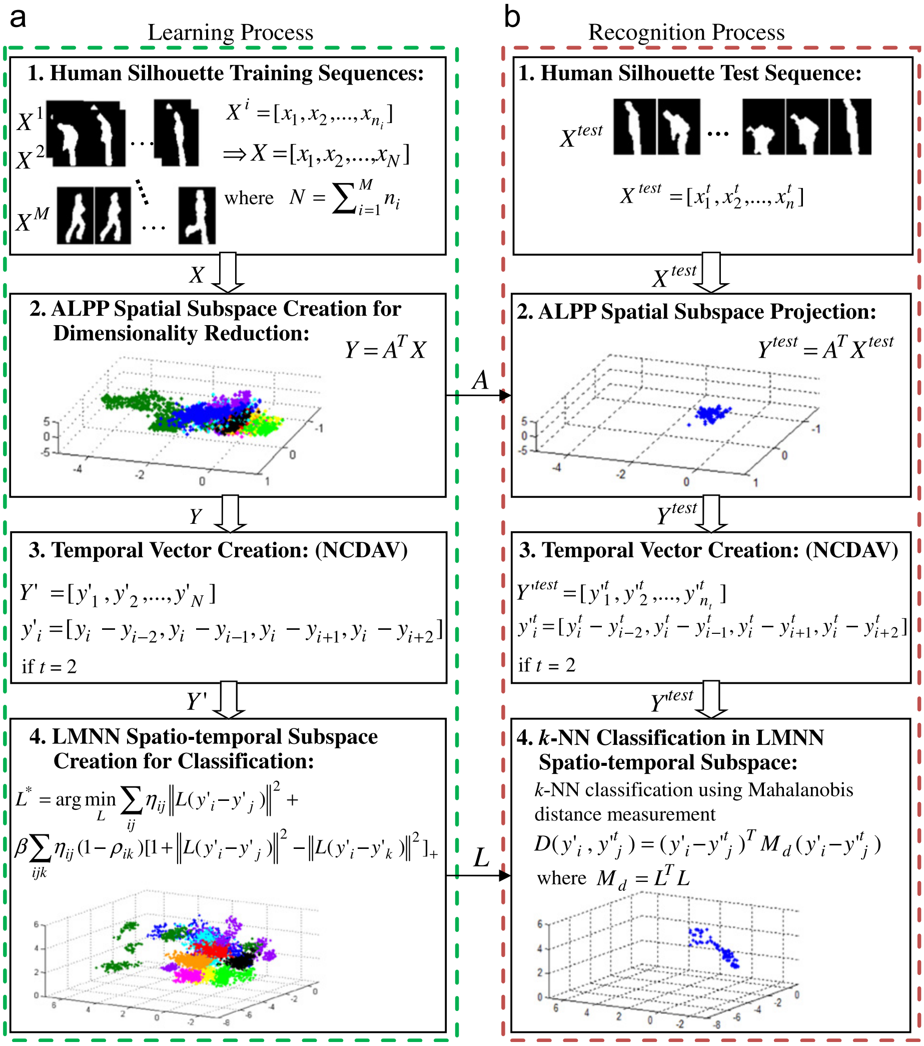

a
Learning Process
1. Human Silhouette Training Sequences:
X^1
X^2
X^M
$X^i=[x_1,x_2,...,x_{n_i}]$
$\Rightarrow X=[x_1,x_2,...,x_N]$
where $N=\sum_{i=1}^{M} n_i$
X
2. ALPP Spatial Subspace Creation for Dimensionality Reduction:
$Y=A^TX$
Y
3. Temporal Vector Creation: (NCDAV)
$Y'=[y'_1,y'_2,...,y'_N]$
$y'_i=[y_i-y_{i-2},y_i-y_{i-1},y_i-y_{i+1},y_i-y_{i+2}]$
if $t=2$
Y'
4. LMNN Spatio-temporal Subspace Creation for Classification:
$L^*=\arg\min_L \sum_{ij}\eta_{ij}\|L(y'_i-y'_j)\|^2+$
$\beta\sum_{ijk}\eta_{ij}(1-\rho_{ik})[1+\|L(y'_i-y'_j)\|^2-\|L(y'_i-y'_k)\|^2]_+$
A
L
b
Recognition Process
1. Human Silhouette Test Sequence:
X^{test}
$X^{test}=[x_1^t,x_2^t,...,x_n^t]$
X^{test}
2. ALPP Spatial Subspace Projection:
$Y^{test}=A^TX^{test}$
Y^{test}
3. Temporal Vector Creation: (NCDAV)
$Y'^{test}=[y'^t_1,y'^t_2,...,y'^t_{n_t}]$
$y'^t_i=[y^t_i-y^t_{i-2},y^t_i-y^t_{i-1},y^t_i-y^t_{i+1},y^t_i-y^t_{i+2}]$
if $t=2$
Y'^{test}
4. k-NN Classification in LMNN Spatio-temporal Subspace:
k-NN classification using Mahalanobis distance measurement
$D(y'_i,y'^t_j)=(y'_i-y'^t_j)^TM_d(y'_i-y'^t_j)$
where $M_d=L^TL$

Fig. 1. Framework of the proposed human action recognition system: (a) the learning process which can generate the spatio-temporal subspace for classification. (b) The recognition process which use k-NN method in LMNN spatio-temporal subspace to recognize the human action.

eliminate the redundant information within the original sequences in order to obtain a low-dimensional spatial subspace. However, in constructing this subspace, the local spatial structure and relations among the data points must be preserved. That is, the data points (images) that are close (similar) in the original high-dimensional space would be also close in the low-dimensional subspace. In the present study, this computation is achieved by using a new Adaptive Locality Preserving Projection (ALPP) algorithm. Importantly, ALPP retains the well-known advantages of the Locality Preserving Projection (LPP) method [33]. In other words, less computational complexity is needed than in methods such as the LE or LLE methods [30,31], which utilize a non-linear spectral embedding technique. In addition, a linear transformation enables LPP to provide a low embedding for new data points without computing the entire matrix from scratch. Moreover, ALPP solves the problem of LPP, which will be discussed in the following paragraph because of its use of a modified graph construction process and linearity measurement.

In LPP, the k-NN method is used to define the neighborhood information in high-dimensional space before constructing the corresponding graph. When applying the k-NN method, four possible relationships exist between any pair of points (see Fig. 2(a), in which the blue edge indicates that the red point is a neighbor of the blue point, while the red edge indicates that the blue point is a neighbor of the red point). For example, consider the case shown in Fig. 3(a), in which a 5-NN clustering scheme is used to construct the graph. Because k is specified as 5, the clustering scheme attempts to group each data point with five neighbors. Consider the red data point shown on the left of Fig. 3(a). This data point has only 4 neighbors, and thus, the clustering scheme is forced to add a remote data point (i.e., the red data point shown on the right of Fig. 3(a) is added to the group of neighbors, as indicated by the red edge between the two points). As shown in Fig. 3(a), the graph comprises two distinct groups of images; i.e., they are unrelated or dissimilar to one

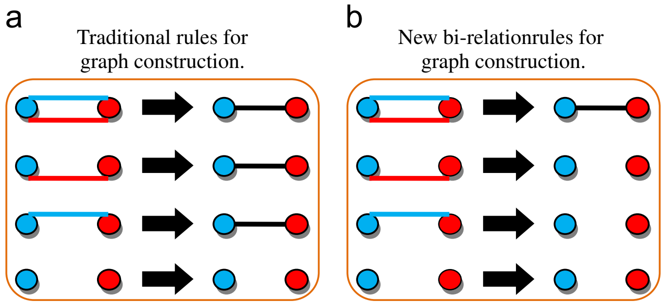

a
Traditional rules for graph construction.
b
New bi-relationrules for graph construction.

Fig. 2. Four situations of graph construction in (a) LPP and (b) ALPP. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

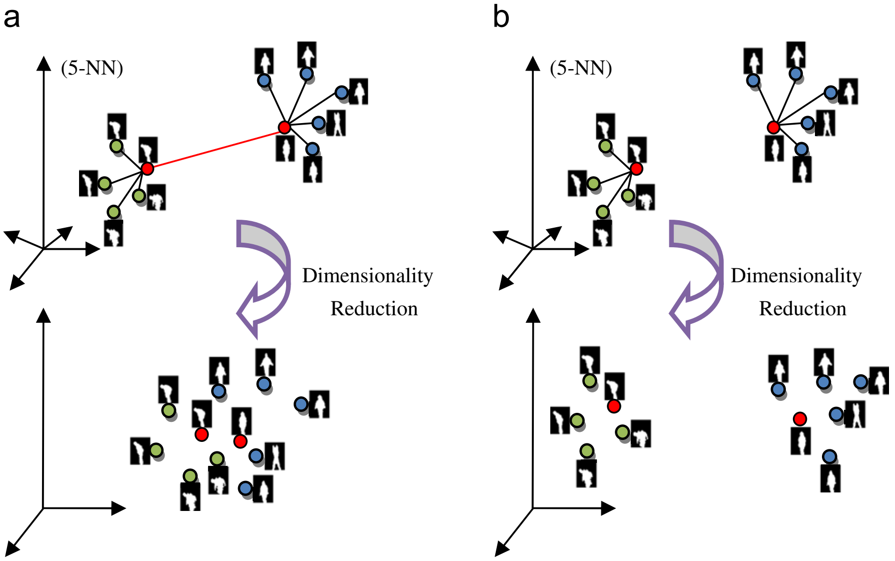


Fig. 3. (a) The problem of LPP where an improper edge to connect the unrelated points may cause the congregation of two different groups and the classification error after the dimensionality reduction. (b) The example of ALPP which use more strict connection rule can avoid the mistake by keeping unimportant data nearby and make the distribution of the same group more discriminant after the dimensionality reduction. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

another. As a result, the edge constructed between them may cause an inappropriate congregation of the two different groups following the dimensionality reduction process (see the lower graph in Fig. 3(a)).

The modified process of building the adjacency undirected graph in ALPP comprises two major components, namely, graph construction and linearity measurement. The goal of the graph construction process is to enforce stricter connection rules in order to discard the neighbors with low connection relations. Meanwhile, the goal of the linearity measurement mechanism is to evaluate the linearity between every pair of points in order to connect data points that are not neighbors but have linearity strong enough to have a corresponding edge in the adjacency undirected graph. In this way, similar data points are grouped together in the low-dimensional subspace, thereby improving its discriminatory power and reducing its dimensionality.

In the process of graph construction, the discriminatory power of the graph is improved by imposing new bi-relation connection rules between each pair of nodes, as shown in Fig. 2(b). As in the LPP method, ALPP also recognizes four possible relationships between each pair of nodes when applying the *k*-NN clustering method. However, in contrast to LPP, ALPP constructs an edge in the adjacency undirected graph only when both nodes in the pair are neighbors of one another. As a result, ALPP avoids the problem inherent in LPP of retaining redundant data points simply to satisfy the requirement for a given number of neighbors and to make the distribution of the same group more discriminant after dimensionality reduction (see the lower graph in Fig. 3(b)).

Following the graph construction process, the linearity measurement mechanism [35] is applied to calculate the linearity between each pair of points with a connected path. Thus, the data with high linearity are grouped together and thereby the low-dimensional spatial subspace can be more compact. The linearity measurement mechanism contains two terms: the Euclidean distance and the geodesic distance. Consider the graph shown in Fig. 4(a), which includes four points and has a path between points A and B. Fig. 4(b) illustrates the Euclidean distance and the geodesic distance between points A and B. As shown, the Euclidean distance is the absolute distance between the two points, while the geodesic distance is the length of the shortest path between the two points. In accordance with [35], the linearity l_{ij} between two data points, x_i and x_j, can be evaluated as

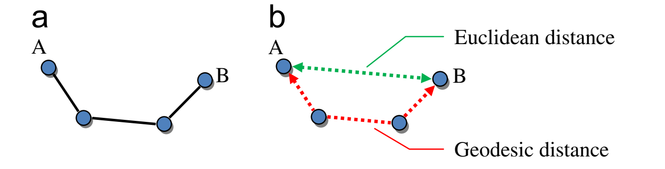


Fig. 4. (a) An example graph constructed by ALPP. (b) The “Geodesic distance” and “Euclidean distance” of points A and B.

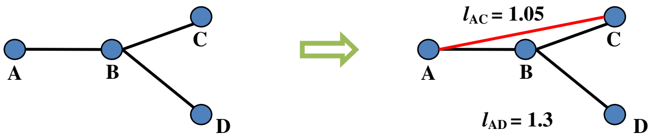


Fig. 5. An example of linearity measurement indicates that the data point C which is not the neighbor of A but its linearity is strong enough will add an edge between them. However, the data point D which is not the neighbor of A and the linearity is not strong enough will not add an edge between them.

$$l_{ij} = \frac{\text{Geodesic distance } (x_i,x_j)}{\text{Eulidean distance } (x_i,x_j)}. \qquad (1)$$

Because the geodesic distance is always equal to or greater than the Euclidean distance, l_{ij} is always equal to or greater than 1. Clearly, the closer the geodesic distance and Euclidean distance are to one another, the greater the linearity is between the two data points. Thus, in the limiting case of $l_{ij}=1$, the two data points are totally linear. In ALPP, l_{ij} is computed for every possible pair of data points, and any edges with a linearity of less than a linearity measurement threshold l_t ($l_t=1.1$ which is defined by the experiments) are added to the adjacency graph that was constructed in the previous step. For example, assume that the original ALPP graph contains four data points, A–D, and that each edge in the graph indicates that the corresponding data points are both neighbors of one another (see Fig. 5). Furthermore, assume that the linearity l_{AC} between points A and C is found to have a value of less than l_t. Consequently, an edge is constructed between them (see the right graph in Fig. 5). In contrast, data points A and D have only a weak linearity (i.e., $l_{AD}>l_t$), and thus, no edge is added

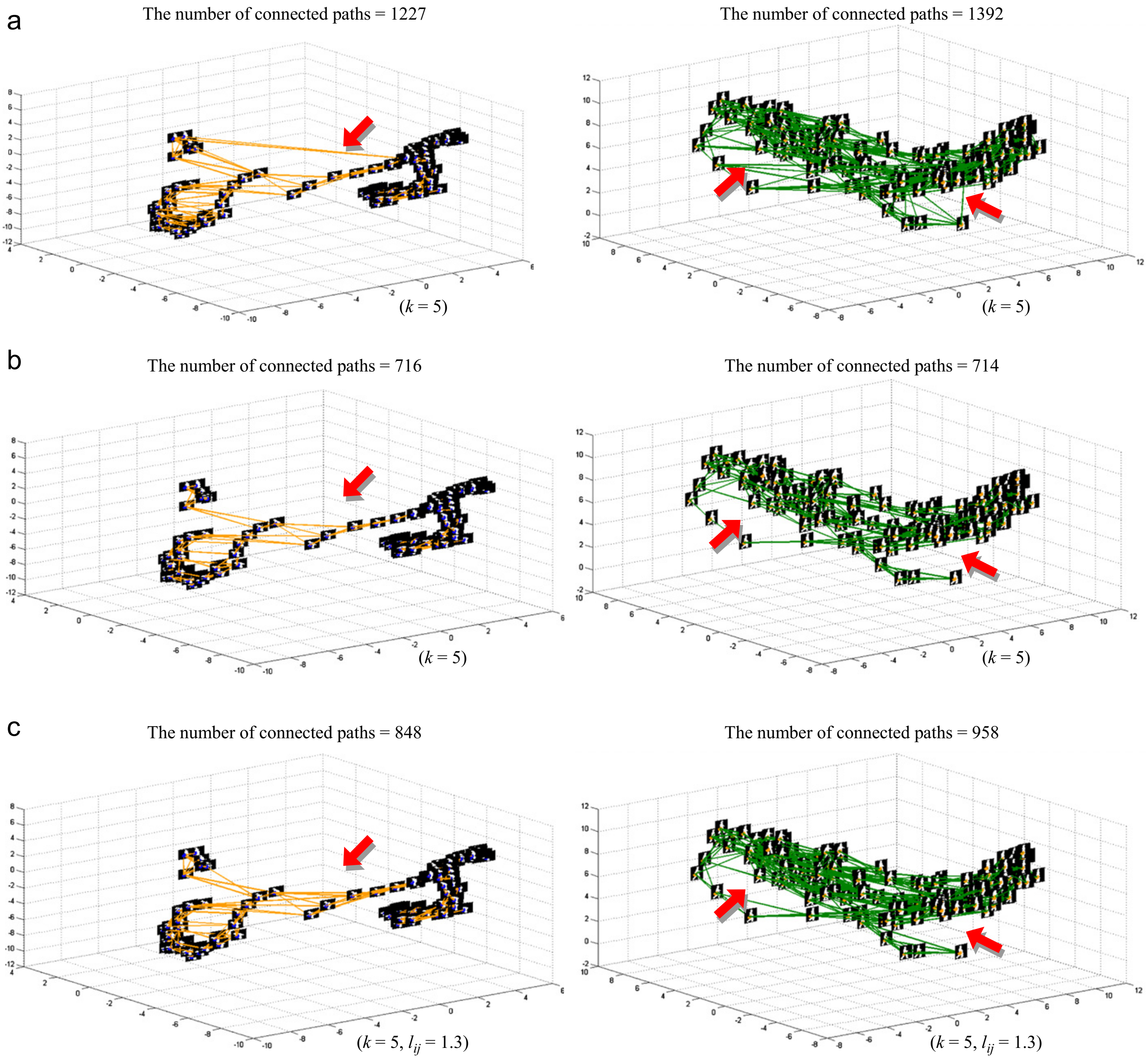


Fig. 6. The adjacency undirected graphs of the two action sequences of *bend* (orange edge) and *walk* (green edge): (a) using the traditional rules, (b) using only new bi-relation rules, and (c) using both new bi-relation rules and linear measurement. The red arrows indicate the difference between adjacency undirected graphs using traditional and proposed method. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

between them. Fig. 6 shows the adjacency undirected graphs of two action sequences, *bend* (orange edge) and *walk* (green edge), using the traditional rules applied in the LPP method, new bi-relation rules, and new bi-relation rules and linear measurement, respectively. The red arrows in Fig. 6(a) indicate that a data point in the graph may have wrong connections with other data points that are far from it. In contrast, the proposed method can reduce the wrong connections to preserve the local structure of the data set (see Fig. 6(c)).

After the graph construction and linearity measurement processes, a weight matrix W is obtained whose elements W_{ij} are either 1 or 0, depending on the connection between x_i and x_j. It is to be noted that W describes both the linearity and the relationship between every pair of points. For example, $W_{ij}=1$ implies a strong linearity and a relationship between points x_i and x_j, whereas $W_{ij}=0$ indicates that the two data points are unrelated to one another. In other words, the value of W is assigned as follows:

$$W_{ij} = \begin{cases} 1 & \text{if } x_i \text{ and } x_j \text{ are connected,} \\ 0 & \text{else.} \end{cases} \tag{2}$$

According to the locality-preserving criterion, which ensures that those points that have sufficient linearity (i.e., $l_{ij} < l_t$) in the original high-dimensional space are grouped together in the low-dimensional subspace [33,35], the weight matrix W that records the local structure or linearity between each pair of data points is added to the objective function. There exists a transformation matrix A to minimize the following objective function:

$$\arg\min \sum_{ij} (y_i - y_j)^2 W_{ij} = \arg\min_{A} \sum_{ij} (A^T x_i - A^T x_j)^2 W_{ij}, \tag{3}$$

where x_i is the original data point and y_i is the corresponding data point in the low-dimensional subspace. Because each transformation vector in A can work independently, Eq. (3) can be rewritten as follows:

$$\arg\min_{a} \sum_{ij} (a^T x_i - a^T x_j)^2 W_{ij}, \tag{4}$$

where a is the transformation vector. After a process of algebraic manipulation [15], Eq. (4) can be reformulated as follows:

$$\arg\min_{a} a^T XLX^T a, \tag{5}$$

where L is the Laplacian matrix; D is the diagonal matrix, in which $D_{ii} = \sum_j W_{ij}$; and $L = D - W$. The value of D_{ii} in D indicates the number of neighbors of data point x_i. In other words, the more neighbors x_i has, the larger the value of D_{ii} and the greater the importance of x_i. As described in [33], the following constraint is imposed on the objective function given in Eq. (5):

$$a^T XDX^T a = 1. \tag{6}$$

This constraint not only causes the data point with the largest value of D_{ii} to be located close to the origin of the low-dimensional subspace but also restrains the distribution of all of the remaining data points. Combining Eqs. (5) and (6), the optimization problem becomes

$$\arg\min_{a} a^T XLX^T a$$
$$\text{s.t. } a^T XDX^T a = 1. \tag{7}$$

Then, Eq. (7) can be solved via the Lagrangian formulation as follows:

$$\begin{aligned} Lagrangian &= a^T XLX^T a - \lambda a^T XDX^T a \\ &\Rightarrow \frac{\partial}{\partial a}(a^T XLX^T a - \lambda a^T XDX^T a) = 0 \\ &\Rightarrow XLX^T a = \lambda XDX^T a, \end{aligned} \tag{8}$$

where the two matrices XLX^T and XDX^T are both symmetric and positive semi-definite. Eq. (8) is a generalized eigen-decomposition problem [4,5,30–33,35], and the transformation matrix $A = [a_1, a_2, \ldots, a_d] \in R^{D \times d}$ is given by the eigenvectors corresponding to the d smallest eigenvalues. Thus, the data in the low-dimensional subspace can be obtained as $Y = A^T X$, where $Y = [y_1, y_2, \ldots, y_N] \in R^{d \times N}$. Fig. 7(a) and (b) show the distribution of Y in the LPP and ALPP subspace. As the diagrams indicate, the distribution of Y in the ALPP subspace is more compact than its distribution in the LPP subspace. In addition, as shown in Fig. 7, the continuity of action in the ALPP subspace is smoother than that in the LPP subspace (i.e., the images that are close (similar) in the original high-dimensional space are also close in the low-dimensional subspace).

2.2. Temporal vector creation

After obtaining the spatial subspace by ALPP, all of the training silhouettes are projected in this subspace, and thus, the

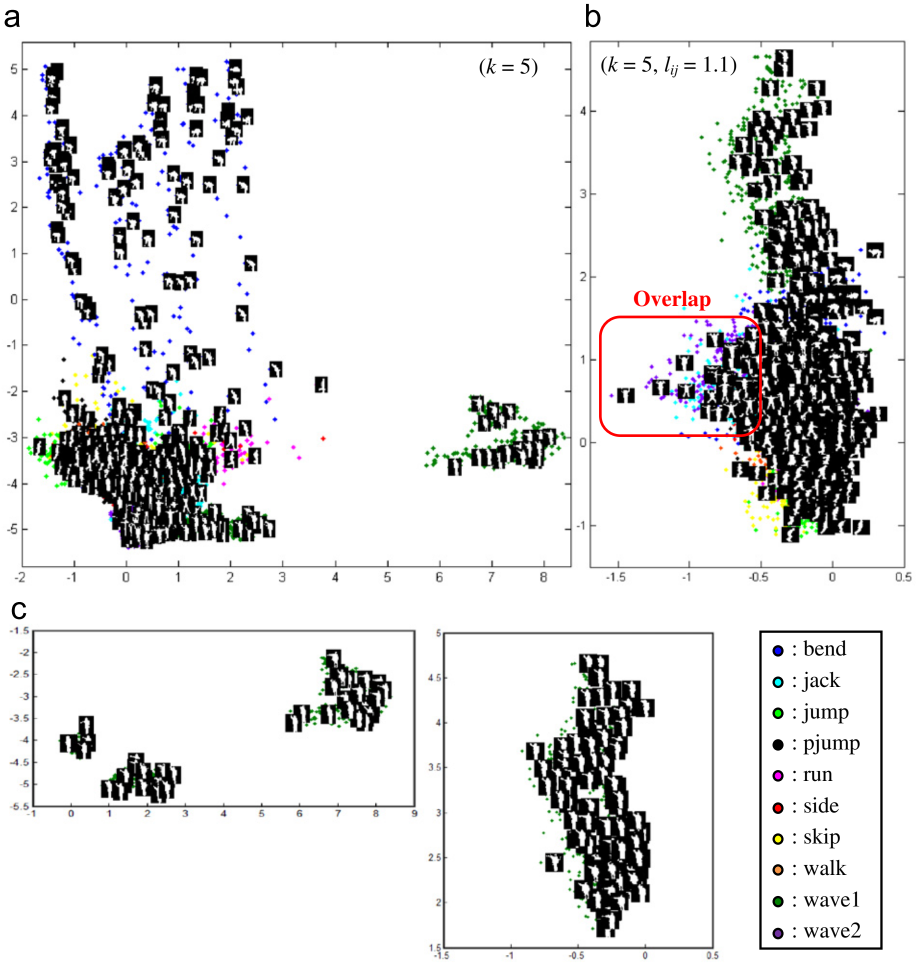


Fig. 7. 2D distribution of 10 action sequences in (a) LPP subspace, and in (b) ALPP subspace, (c) is the 2D distribution of *wave1* action sequence in LPP and ALPP, respectively. In addition, the smooth change of silhouette images and the compactness of distribution indicate that ALPP subspace can preserve better local structure and linearity of data points. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

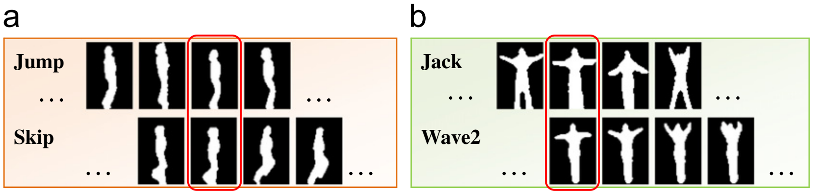


Fig. 8. (a) Example silhouette sequences of *jump* and *skip*. (b) Example silhouette sequences of *jack* and *wave2*. The red round rectangles indicate the similar and ambiguous parts of two different actions. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

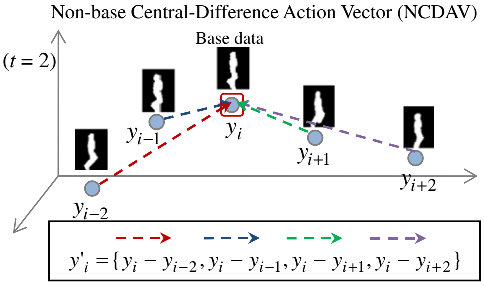

Non-base Central-Difference Action Vector (NCDAV)
(t = 2)
Base data
y_{i-1}
y_i
y_{i+1}
y_{i+2}
y_{i-2}
$y'_i=\{y_i-y_{i-2},y_i-y_{i-1},y_i-y_{i+1},y_i-y_{i+2}\}$

Fig. 9. Conceptual graph of NCDAV in the ALPP subspace. NCDAV is formed by the central difference which is concatenated between temporally closet data points and base data.

computational complexity of the learning process is reduced. However, due to the ambiguity of the human body shape in some specific images among different action types (see the red round rectangles as shown in Fig. 8), an overlap occurs in the spatial subspace. In other words, according to the graph construction rules of ALPP, those similar images will be located together (see the red round rectangle as shown in Fig. 7(b)) in the ALPP subspace even if they belong to different action types. Accordingly, inspired from the extraction of feature points variation by optical flow for the facial expression recognition, a Non-base Central-Difference Action Vector (NCDAV) method is proposed which only considers the variation between consecutive data in the spatial subspace to reduce the ambiguous or corrupted effect, as follows:

$$y'_i=\begin{cases}[y_i-y_1,\ldots,y_i-y_{i-1},y_i-y_{i+1},\ldots,y_i-y_{2t+1}] & \text{if } i\le t,\\ [y_i-y_{m-2t},\ldots,y_i-y_{i-1},y_i-y_{i+1},\ldots,y_i-y_m] & \text{if } i\ge m-t,\\ [y_i-y_{i-t},\ldots,y_i-y_{i-1},y_i-y_{i+1},\ldots,y_i-y_{i+t}] & \text{otherwise,}\end{cases} \tag{9}$$

where t is a parameter governing the period over which the temporal information is to be taken into account, m is the total number of frames in each sequence, and y_i denotes the "base data". Fig. 9 illustrates the basic concept of the proposed approach. Assuming that t is assigned a value of 2, four data points that not only belong to the same sequence but are also temporally closest to the base data y_i are added to form a new vector, which includes temporal information. Note that if $i\le t$ or $i\ge m-t$, only the first and last $2t+1$ data points are added to the new vector, to preserve the consistency of the temporal data. In other words, the temporal vector (see Eq. (9)) encodes the temporal information by computing the difference between the consecutive data y_j with the base data y_i $(i\ne j)$. Moreover, the temporal vector including the base data which is similar to other different action types or corrupted by noise would still result in the ambiguity problem. Hence, to reduce the effects of ambiguity between different actions, the base data is discarded from the temporal vector. Fig. 10 shows the data distribution of actions (*jack* and *wave2*), and it can be observed from this figure that the distribution with the temporal vector including the base data is highly overlapped and results in an ambiguous problem (see Fig. 10(a)). The distribution of temporal vectors without the base data can reduce the effect of ambiguity different actions (see Fig. 10(b)). Finally, all of the $y'_i\in R^{2dt}$ are aggregated to form Y', in which $Y'=[y'_1,y'_2,\ldots,y'_N]\in R^{2dt\times N}$, where N is the total number of data in the training set.

2.3. Spatio-temporal subspace creation using Large Margin Nearest Neighbor (LMNN)

In this study, a simple and non-linear approach, namely k-nearest neighbor (k-NN) classifier, is applied to decide the class label of the test data in the test process. The mechanism of k-NN classifier is to classify the test data according to the labels of nearby neighbors. However, there are two issues which should be taken into consideration: First, if the extracted features for the training data are not discriminant enough to maximize the separability between different action classes, the classification results of k-NN will not be promising. Second, the distance measurement has effects on the determination of the nearby neighbors. As known, there are several distance measurements (e.g. Euclidean distance and Mahalanobis distance), which approach is suitable for k-NN classifier? For the past few years, many researches devoted themselves on this issue. Among the studies about the derivation of good distance metric, the Large Margin Nearest Neighbor (LMNN) method is the state-of-the-art. Also it is the first approach to design the distance metric based on the mechanism of k-NN classifier [22]. Therefore, the LMNN method, which ensures the compact grouping of data points with the same class while simultaneously maximizing the separation distance between data points with different classes is used to construct the spatio-temporal subspace required for k-NN classifier.

The LMNN method commences by assigning to the temporal vectors $Y'=[y'_1,y'_2,\ldots,y'_N]$ a corresponding class label $l_i=\{1,2,\ldots,c\}$, where c is the total number of class label types. The distance between vectors y'_i and y'_j is then computed using the Mahalanobis distance metric. Let the transformation matrix L with dimensions $2dt\times 2dt$ be defined as follows:

$$D(y'_i-y'_j)=\|L(y'_i-y'_j)\|^2. \tag{10}$$

To improve the performance for the k-NN classifier, the transformation matrix L can be obtained from the following objective function:

$$L^*=\arg\min_L \sum_{ij}\eta_{ij}\|L(y'_i-y'_j)\|^2 +\beta\sum_{ijk}\eta_{ij}(1-\rho_{ik})[1+\|L(y'_i-y'_j)\|^2-\|L(y'_i-y'_k)\|^2]_+ \tag{11}$$

As shown, Eq. (11) comprises two competing terms. In the first term, η_{ij} has a value of either 1 or 0, depending on the relationship between y'_i and y'_j. If y'_j is a neighbor of y'_i and both vectors share the same label type, η_{ij} is assigned a value of 1; otherwise η_{ij} is set to 0. This term penalizes a large separation distance between vectors y'_i and y'_j if η_{ij} has a value of 1. In other words, it prompts the LMNN algorithm to minimize the distance between neighbors

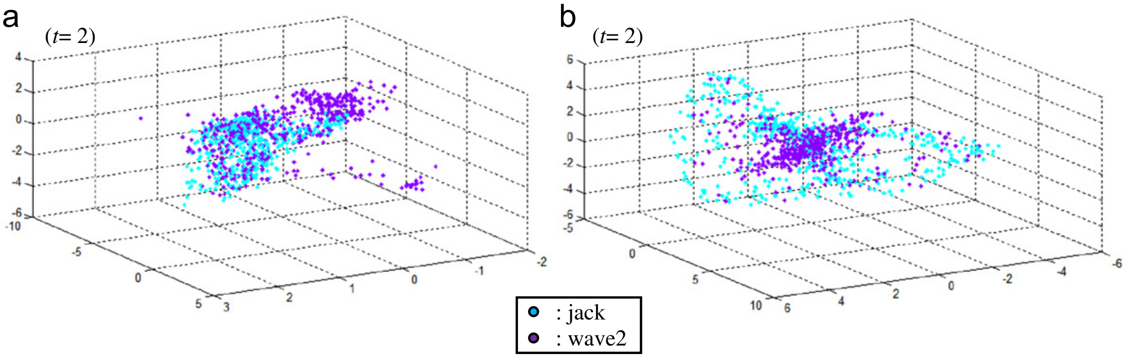


Fig. 10. 3D visualization of ambiguous temporal vectors (*jack* and *wave2*): (a) base data is retained and (b) base data is discarded.

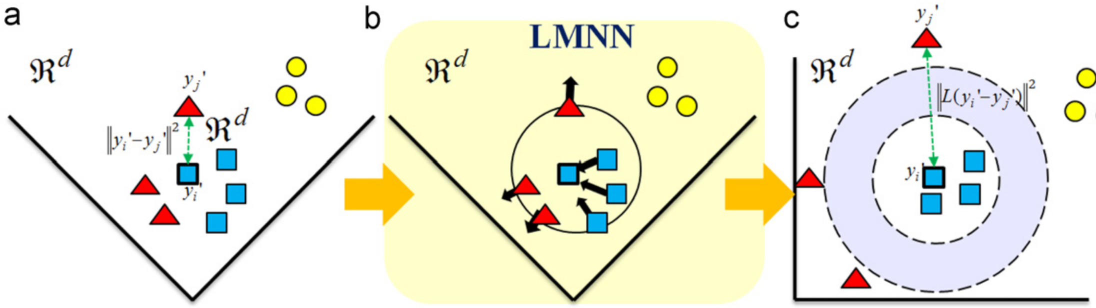


Fig. 11. Complete schematic illustration of LMNN: (a) the original data distribution which dimension is *d*, (b) during the LMNN process, the neighbor data with the same label will be pulled in, and the neighbor data with different label will be pushed out and (c) the result after the LMNN which dimension is *d*.

with the same class label when constructing the spatio-temporal subspace.

In the second term in Eq. (11), β is a positive constant that is used to weight the relative importance of the two competing terms, and ρ_{ik} has a value of either 1 or 0. Specifically, ρ_{ik} is set to 1 if vectors y'_i and y'_k have the same class label and is set to 0 otherwise. This term penalizes a small separation distance between vectors with a different class label. In other words, the term serves to separate vectors that have a different class label within the spatio-temporal subspace.

We note that the dimension of the spatial–temporal subspace is the same as that of the temporal vector ($\in R^{2dt}$). In other words, the aim of LMNN is not to reduce the dimension of the temporal vector but to re-organize the data points in such a way that they are more amenable to classification. Fig. 11 presents a schematic illustration of the LMNN process. Fig. 11(a) shows the original graph constructed by the ALPP and NCDAV algorithms, while Fig. 11(b) shows the effect of the optimization function in bringing together vectors with the same class label while driving away vectors with a different class label. Finally, Fig. 11(c) shows the final spatio-temporal graph. As described in [22], the optimization function given in Eq. (11) can be reformulated as a semi-definite program (SDP) problem and more discussion can be referred to [22]. Fig. 12 shows the 3D visualization of the spatio-temporal subspace after the LMNN method. The data points of the same action cluster together, whereas the data points of ambiguous actions, such as *jack* and *wave2*, are separated (see Fig. 12(b)).

3. Recognition process using *k*-NN classification in a spatio-temporal subspace

After the learning process, two transformation matrices are obtained that allow us to analyze the spatial and temporal information of new data points; the matrices are the spatial subspace transformation matrix *A* obtained from ALPP and the spatio-temporal subspace transformation matrix *L* obtained from LMNN. The recognition process commences by obtaining human silhouettes as input test sequences. The input sequences are transformed to the spatio-temporal subspace via the two transformation matrices and the NCDAV method, and they are then recognized using a *k*-NN classifier.

As shown in Fig. 1(b), the binary silhouettes in the input test sequence are centralized and normalized to a consistent size of 64 × 48 pixels. The normalized sequence is noted as $X^{test} = [x_1^t, x_2^t, \ldots, x_n^t]$, where *n* is the total number of frames in the sequence. Each normalized frame $x_i^t(i = 1-n)$ is projected to the spatial subspace ($Y^{test} = A^T X^{test}$) by transformation matrix *A*. The spatial data are then extended to the temporal vector by the NCDAV method, as defined in the training process. Finally, the temporal data are transformed to the spatio-temporal subspace via the transformation matrix *L*.

Having transformed the temporal data to the spatio-temporal subspace, the human action portrayed in the input sequence is recognized using the *k*-NN classifier. When implementing the classifier, the value of *k* is set to 6. In other words, 6 neighbors are found for each test frame in the spatio-temporal subspace. Because the transformation matrix *L* has already been obtained from the learning process, the class label of the six neighbors is examined, and a majority voting mechanism is used to determine the overall class of the test frame. Once all of the test frames $x_i^t(i = 1-n)$ have been assigned a class, a majority voting mechanism is once again applied to determine the overall class of the test sequence.

4. Experimental results

In this study, three databases are considered: Weizmann [24], ORL [36] and MNIST [37], to ensure the evaluation is extensive for proposed method. ORL database and MNIST database are only

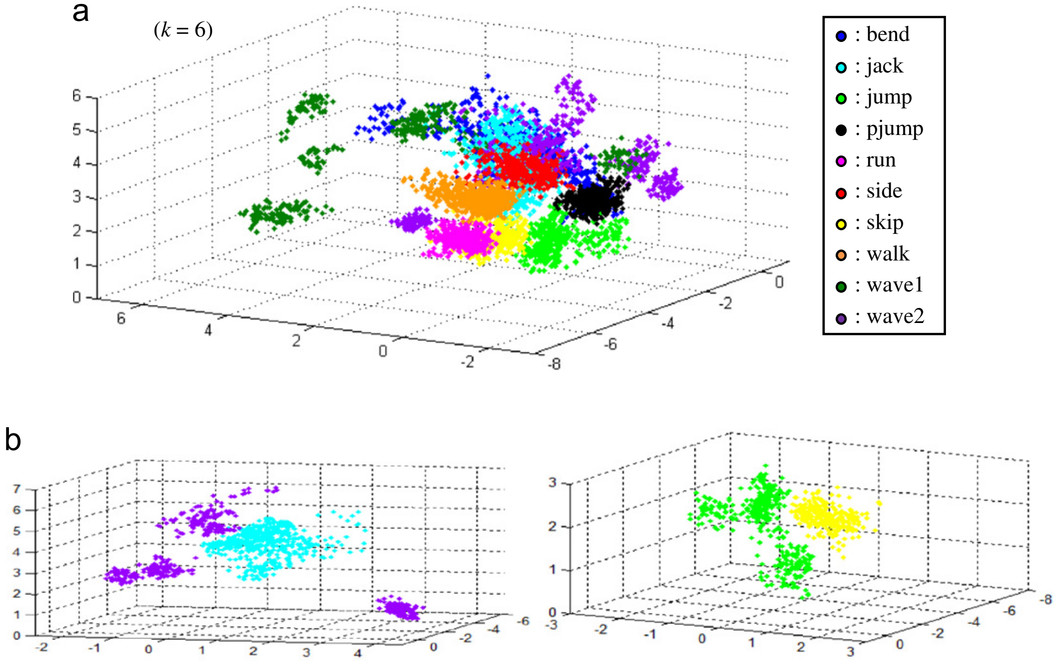


Fig. 12. 3D visualization of the distribution in spatio-temporal subspace after LMNN method: (a) the distribution of total 10 action sequences and (b) the distribution of different ambiguous action sets by different angle view, such as *jack* and *wave2*, *jump* and *skip*.

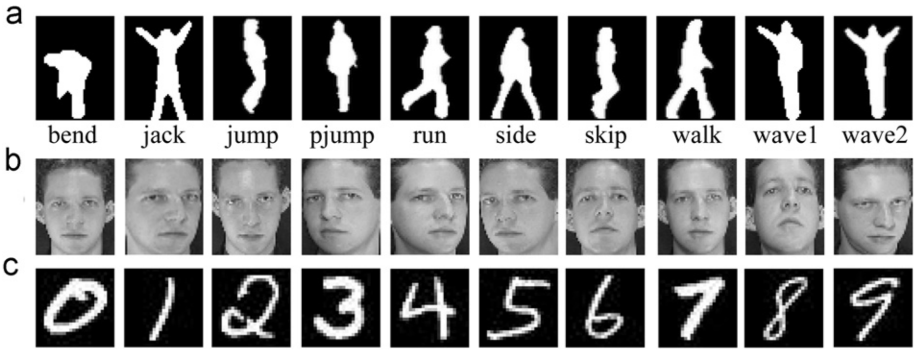


Fig. 13. The sample images cropped from (a) Weizmann database [24], (b) ORL database [36] and (c) MNIST database [37], respectively.

used for the evaluation of ALPP, but Weizmann database is used for all the evaluations, such as the robustness test and evaluation of whole system. In this section, the optimized parameters (k, l_t and d) of ALPP were defined first with quantitative experiments on Weizmann, ORL and MNIST databases. Then, to evaluate the proposed ALPP, those three databases are used again to compare ALPP with other methods, such as Principal Component Analysis (PCA) [38], Linear Discriminant Analysis (LDA) [39], LSDA [34] and LPP [33]. Subsequently, the recognition performance with the NCDAV method was compared with other alternative forms of temporal vectors. In addition, the action sequences with various degrees of synthetic noise were used to test the robustness of the proposed method. Finally, the performance of the proposed system was compared with that of existing silhouette-based and feature-based human action recognition systems. In every case, the experiments were performed on an ASUS PC with an AMD 3.20 GHz CPU and 2 G of RAM.

4.1. Database collection

The Weizmann database consists of 10 different actions performed by 9 different individuals. The 10 actions are displayed in Fig. 13(a): bending (*bend*), jumping jack (*jack*), jumping forward on two legs (*jump*), jumping in place on two legs (*pjump*), running (*run*), galloping sideways (*side*), skipping (*skip*), walking (*walk*), waving one hand (*wave1*), and waving two hands (*wave2*). The database contains a total of 93 sequences because some of the actions are performed more than once by the same individual. However, in the present experiments, 90 sequences were used (i.e., the repeated actions were omitted). As in the learning process, the silhouettes were centralized, cropped and resized to 64 × 48 pixels. To obtain unbiased estimation results, the recognition tests were performed using a nine-fold cross-validation technique. More specifically, in each run of a test, all of the sequences corresponding to a specific individual (i.e., 10 sequences with different actions) were used as test sequences in the recognition process, while the remaining 80 sequences in the database were used in the learning process. The recognition results were then averaged over nine runs, where each run corresponded to a different individual.

The ORL database contains 400 images of 40 individuals. The images are gray-level bitmaps captured at different times and have different variations including expressions (i.e., open or closed eyes, smiling or non-smiling) and facial details (i.e., glasses or no glasses). Also, the images were taken with a tolerance for some tilting and rotation of the face up to 20 degrees, and resized to 64 × 64 pixels. The sequence of experiment is designed as same as previous studies [9], [34]; a random subset with p (=4, 5, 6) images per individual was taken for training and the rest of the database was considered to be the testing set, namely p-Train. For each given p, the results are recorded over 20 random splits. Then, the best result and its corresponding dimensionality for each method are reported. Some sample images are displayed in Fig. 13(b).

The MNIST database is a widely known benchmark which contains a training set of 60,000 images of 10 individuals and a test set of 10,000 images. The images are gray-level bitmaps resized and centered in a 28 × 28 pixels frame. For computational reasons, 800 images of each class in training set are selected randomly for training and 8000 images of test set are selected randomly for testing. Also, we tested for 20 times, and merely the best result and corresponding dimensionality for each method are reported. Some selected images are displayed in Fig. 13(c).

4.2. Comprehensive evaluation of ALPP

Quantitative experiments on Weizmann database [24], ORL database [36] and MNIST database [37] are conduced to investigate the optimal parametric settings of the proposed ALPP method, including the dimensionality (d) of the reduced subspace, the linearity measurement threshold (l_t), and the number of neighborhood (k). Note that the values of the parameters are l_t=1.0–1.5, d=1–50 and k=1–8 for the tests. For each object parameter, Fig. 14 reports the lowest error rate among all the tests with variant values of other parameters.

For the dimensionality of the ALPP subspace (see Fig. 14(a)), it can be observed that the error rate is stable (the maximum difference of error rates is smaller than 0.3%) when the dimensionality of the reduced subspace is larger than 34, 38, and 39 for Weizmann, ORL (5-Train) and MNIST databases, respectively. According to the results, considering the trade-off between the computation time and the error rate, d=34, 38 and 39 are recommended for the ALPP on Weizmann, ORL (5-Train) and MNIST databases, respectively. For the optimal setting of the linearity measurement threshold (see Fig. 14(b)), the error rate is the lowest when the value of l_t=1.1 for Weizmann database and MNIST database, and l_t=1.2 for ORL database (5-Train). In addition, it can be observed that the recognition performance deteriorates as the value of the linearity threshold parameter increases on all the databases. This result is consistent with expectation because the aim of ALPP in constructing the spatial subspace is to preserve linearity in the local data structure. Consequently, it follows intuitively that the classification performance decreases as data points having weaker linearity are involved in the constructed adjacency matrix. For determining the number of neighbors in the ALPP (see Fig. 14(c)), it can provide the lowest error rate on Weizmann and MINST databases when k is set to 5 and 6. However, the optimal value of k for ORL database (5-Train) is 3 and the difference of the error rate by using k=3 and 5 is 2.5%. It's because the training number of each class in ORL database is smaller than others. Hence, smaller k can help ALPP preserve local structure of the data distribution when the training number of each class is small.

Table 1 compares the recognition accuracy of different approaches, such as PCA [38], LDA [39], LSDA [34], LPP [33] and ALPP on ORL [36], MNIST [37] and Weizmann [24] databases. Noted, all the methods use k-NN as the classifier (k is set to 5) for its simplicity. It can be seen that ALPP can outperform other methods with sufficient training data, such as for the MNIST and Weizmann databases. However, when the distribution of data set is complex and the training data cannot represent the data distribution well, such as the case of ORL database (5-Train), ALPP appears to be less effective than supervised methods (LDA and LSDA), though still better than the unsupervised methods (LPP and PCA). In addition, when the training samples are larger, such as the case of ORL database (6-Train), ALPP performs better than LDA.

4.3. Performance comparison of different temporal vector types

To investigate the contribution of the spatio-temporal information toward the performance of the proposed human action

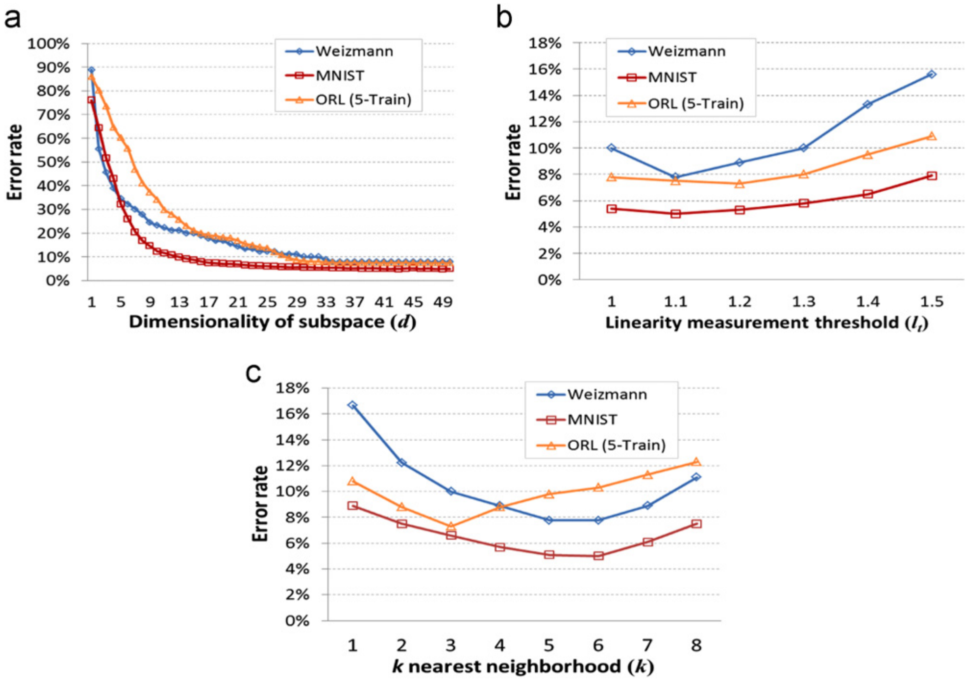


Fig. 14. Error rate of ALPP on Weizmann database [24], ORL database [36] and MNIST database [37] using different parameters: (a) reduced dimensionality (d), (b) linearity measurement threshold (l_t) and (c) k nearest neighborhood (k).

Table 1
The accuracy rates of different approaches on ORL, MNIST and Weizmann databases. The number in parentheses is the reduced dimensionality (d) for the corresponding methods.

Methods	Recognition accuracy (%)				
	ORL (4-Train)	ORL (5-Train)	ORL (6-Train)	MNIST	Weizmann
PCA [38]	81.9 (161)	87.2 (198)	89.3 (233)	93.6 (59)	80.2 (58)
LDA [39]	89.9 (38)	93.3 (38)	94.2 (38)	85 (9)	83.1 (9)
LSDA [34]	90.5 (39)	93.6 (39)	94.9 (39)	91.5 (35)	90.0 (35)
LPP [33]	90.3 (39)	92.2 (39)	93.5 (39)	93.7 (31)	86.7 (40)
ALPP	90.3 (38)	92.7 (38)	94.4 (38)	95.0 (39)	92.2 (34)

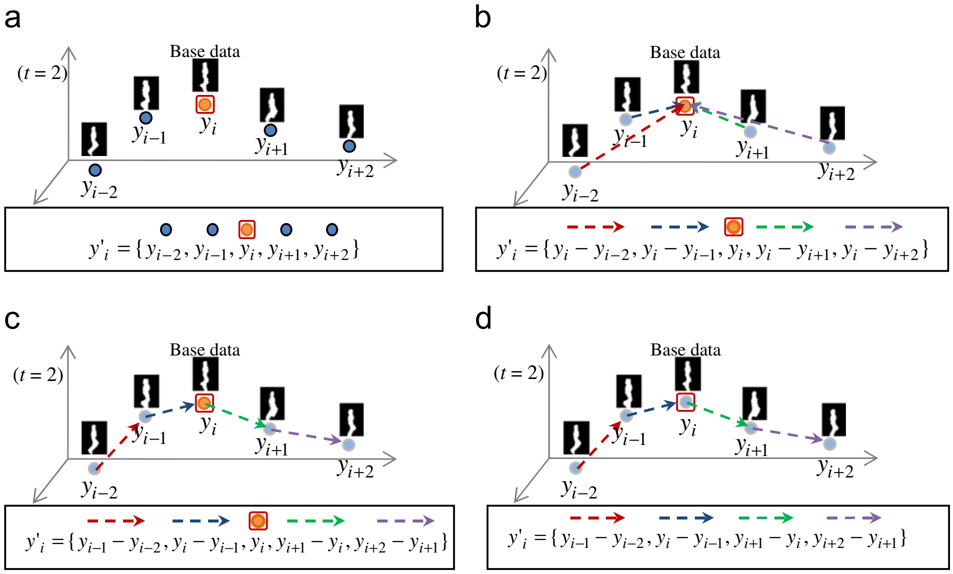

a
Base data
(t = 2)
y_{i-2}
y_{i-1}
y_i
y_{i+1}
y_{i+2}
$y'_i = \{y_{i-2}, y_{i-1}, y_i, y_{i+1}, y_{i+2}\}$
b
Base data
(t = 2)
y_{i-2}
y_{i-1}
y_i
y_{i+1}
y_{i+2}
$y'_i = \{y_i - y_{i-2}, y_i - y_{i-1}, y_i, y_i - y_{i+1}, y_i - y_{i+2}\}$
c
Base data
(t = 2)
y_{i-2}
y_{i-1}
y_i
y_{i+1}
y_{i+2}
$y'_i = \{y_{i-1} - y_{i-2}, y_i - y_{i-1}, y_i, y_{i+1} - y_i, y_{i+2} - y_{i+1}\}$
d
Base data
(t = 2)
y_{i-2}
y_{i-1}
y_i
y_{i+1}
y_{i+2}
$y'_i = \{y_{i-1} - y_{i-2}, y_i - y_{i-1}, y_{i+1} - y_i, y_{i+2} - y_{i+1}\}$

Fig. 15. Various forms of temporal vectors in the ALPP space: (a) Action Trajectory Vector (ATV), (b) Central-Difference Action Vector (CDAV), (c) Adjacent-Difference Action Vector (ADAV) and (d) Non-base Adjacent-Difference Action Vector (NADAV).

recognition system, a series of experiments was performed in which the recognition accuracy of the proposed ALPP+NC-DAV+LMNN scheme was compared with that of ALPP+LMNN schemes in which NCDAV was replaced by four forms of temporal vectors: Action Trajectory Vector (ATV), Adjacent-Difference Action Vector (ADAV), Central-Difference Action Vector (CDAV), and Non-base Adjacent-Difference Action Vector (NADAV). ATV is formed by incorporating the feature vectors in the spatial subspace of those data that are temporally close to the base data as additional information. ADAV calculated the difference between two adjacent data; this type of difference is called an adjacent difference. Then, ADAV is formed by incorporating the adjacent difference as additional information. CDAV is formed by adding the difference between the base data and those data that are close to it as additional information. Finally, NADAV also uses the adjacent difference of the data that are close to the base data as additional information, but it eliminates the base data. The various spatio-temporal vectors are illustrated schematically in Fig. 15.

Fig. 16 shows the recognition results obtained when using the LPP-LMNN and ALPP-LMNN methods with various types of temporal information vectors. (Note that the value of t used by NCDAV in extracting the temporal data is set to 2.) Comparing the recognition results shown in Fig. 16, it is clear that the inclusion of temporal information in the classification process yields a notable improvement in the recognition accuracy, irrespective of the method used to construct the spatial subspace (i.e., LPP or ALPP). Furthermore, irrespective of the temporal vector, the recognition accuracy obtained using the ALPP-based framework is better than that obtained using the LPP-based framework because the compact spatial subspace created by ALPP preserves linearity in the local data structure. In addition, the results of ALPP incorporated only with NCDAV (orange bars) indicate that the use of temporal vectors can reduce the effects of ambiguity between some action sequences. Moreover, the frameworks in which the temporal vectors eliminate the base data (i.e., NADAV and NCDAV) achieve a higher classification performance than those in which the base data is retained. Generally speaking, misclassification errors of the input test sequences are due to either an ambiguity in the body shape among different action classes (see Fig. 8) or the effects of noise or a non-standard execution of the prescribed action (see Fig. 17). Hence, NADAV and NCDAV, which remove the base information, can avoid the above disadvantages. As shown in the two confusion matrices presented in Fig. 18, the ALPP+NCDAV+LMNN framework has an improved robustness toward ambiguity, noise-corruption and non-standard actions compared to ALPP+CDAV+LMNN. Moreover, as shown in Fig. 16, the ALPP+NCDAV+LMNN framework obtains the highest recognition accuracy (98.9%) of all of the various frameworks.

4.4. Performance evaluation under noise-corrupted silhouettes

To test the robustness of the proposed ALPP+NCDAV+LMNN framework, an additional series of experiments was performed in which salt and pepper noise with a density of between 0.1 and

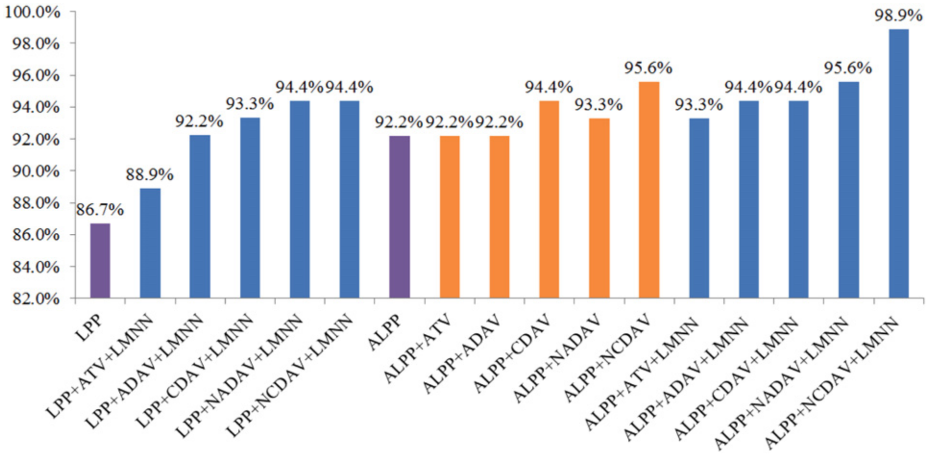


Fig. 16. Comparison bar chart based on various combinations of methods. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

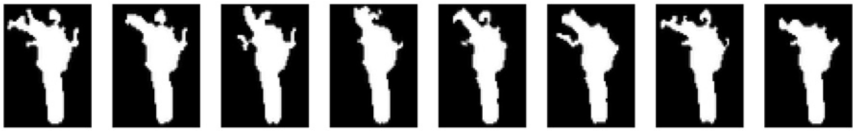
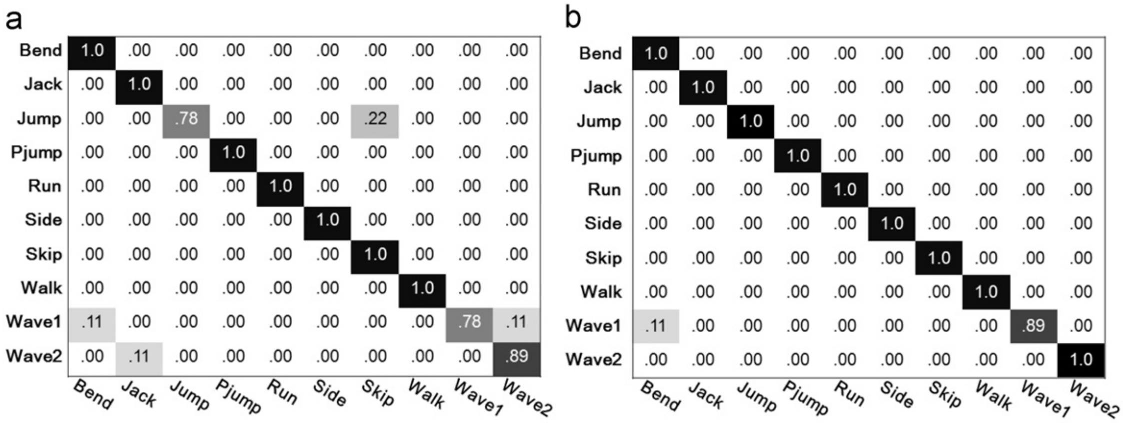


Fig. 17. A noisy sequence of action *wave1* that will lead to classification error.



a

	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	.78	.00	.00	.00	.22	.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	.11	.00	.00	.00	.00	.00	.00	.00	.78	.11
Wave2	.00	.11	.00	.00	.00	.00	.00	.00	.00	.89

b

	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	.11	.00	.00	.00	.00	.00	.00	.00	.89	.00
Wave2	.00	.00	.00	.00	.00	.00	.00	.00	.00	1.0

Fig. 18. Recognition performance of our approach measured using confusion matrices: (a) using ALPP+CDAV+LMNN and (b) using ALPP+NCDAV+LMNN. Vertical rows show ground truth, and horizontal columns indicate recognition results.

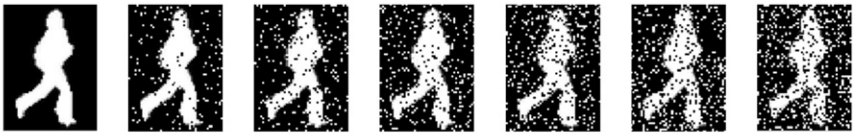


Fig. 19. From left to right are the noise-corrupted silhouettes with different degrees of synthetic noise, which are 0, 0.1, 0.15, 0.2, 0.25, 0.3 and 0.35, respectively.

Table 2
Robustness evaluation with respect to different degrees of synthetic noise density *V*.

Methods	Recognition accuracy (%)						
	$V=0$	$V=0.1$	$V=0.15$	$V=0.2$	$V=0.25$	$V=0.3$	$V=0.35$
ALPP+NCDAV+LMNN	**98.9**	96.7	93.3	92.2	89.9	81.1	77.8
ALPP+CDAV+LMNN	94.4	92.2	91.1	88.9	85.6	75.6	68.9

0.35 was added to the input silhouette images. Note that the noise density V indicates the number of corrupted pixels as a fraction of the total number of pixels in the image. Fig. 19 illustrates the effects of the various density values V on one frame within the *run* silhouette sequence in the Weizmann database. For each action in the Weizmann database, the uncorrupted silhouettes were used as training data. Then, the noise-corrupted silhouettes were classified individually using the ALPP+NCDAV+LMNN and ALPP+CDAV+LMNN methods, respectively. The classification results obtained using the two frameworks are summarized in Table 2. As shown, the ALPP+CDAV+LMNN method is more sensitive to noise than the ALPP+NCDAV+LMNN method. This result is to be expected because the corrupted base data is retained within the temporal vector constructed using CDAV but excluded by NCDAV. For both frameworks, the recognition performance deteriorates as the noise density increases. However, the proposed ALPP+NCDAV+LMNN framework achieves a consistently high recognition performance (i.e., $>92.2\%$) provided that the noise density does not exceed 0.2.

4.5. Performance comparisons with existing action recognition systems

The performance of the proposed method was further evaluated by comparing its recognition accuracy with that of other silhouette-based human action recognition systems using the Weizmann database. The corresponding results are presented in Table 3. Among the various methods listed in Table 3, the Local Spatio-Temporal Discriminant Embedding (LSTDE) method proposed by Jia and Yeung [4] is similar to the method proposed in this study in that it is also based on a dimensionality reduction approach. Specifically, LSTDE uses LSDA to generate a spatial subspace and then applies a canonical correlation technique to mesh the temporal information with the spatial information. However, the LSTDE method achieves lower recognition accuracy (90.9%) than the ALPP+NCDAV+LMNN method proposed in this study. Furthermore, the ALPP+NCDAV+LMNN method allows the human action recognition process to be accomplished in real time. For example, given a value of $t=2$ in the NCDAV algorithm, each frame in the input sequence can be classified within 33 ms. The recognition accuracy achieved using the method proposed by Wu et al. [40] is identical to that achieved using the proposed method. However, the computational cost of the method in [40] is significantly higher than that of the ALPP+NCDAV+LMNN method.

Table 3
Recognition accuracy of some silhouette-based approaches which use the silhouette of Weizmann database as the input data. All of these approaches use evaluation method of leaving one out cross validation.

Methods	Recognition accuracy (%)
Our approach	**98.9**
Wu et al. [40]	98.9
Zhang and Gong [41]	89.4
Jia and Yeung [4]	90.9
Poppe and Poel [25]	95.6
Wang and Suter [26]	97.8

Table 4 compares the recognition accuracy of the proposed method with that of several existing methods in which certain local features of the Weizmann database (e.g., optical flow features [6] or sparse spatio-temporal interest points [16]) are taken as the input to the recognition process rather than the entire silhouette image. The results show that the ALPP+NCDAV+LMNN method can provide the promising results.

5. Conclusions

This paper has proposed a silhouette-based human action recognition system in which a discriminant spatio-temporal subspace is constructed in a learning process, and unknown human actions are then recognized using a k-NN classifier. In the learning process, the silhouette data are transferred to a low-dimensional discriminant spatial subspace by a modified version of the Locality Preserving Projection (LPP) method [33] designated as the Adaptive Locality Preserving Projection (ALPP) method. The temporal information within the spatial subspace is then extracted using a new Non-base Central-Difference Action Vector (NCDAV) technique. Finally, the Large Margin Nearest Neighbor (LMNN) metric learning method is used to build a discriminant and efficient spatio-temporal subspace for classification purposes. The experimental results have shown that the proposed system outperforms existing silhouette-based or feature-based human action recognition systems. Moreover, the proposed method has a low-computational complexity due to the simple operation of low-dimensional matrices used in the recognition process. Finally, the proposed system is robust toward the effects of noise in the input images. Accordingly, the ALPP+NCDAV+LMNN framework proposed in this study represents an ideal solution for real-time, real-world human action recognition applications.

In the future, the system could be improved from the following perspectives. In order to apply the system for the daily applications, more action types are needed. However, only silhouette information as input neglecting the 3D information will cause the ambiguity and limit the system capability for real-world. For example, waving hands forward the body or clapping hands will be ambiguous. Hence, applying multiple data source, such as depth information or multi-view images (capturing one action sequence with different cameras from different views) could be a way to improve our system. Besides, depth information can provide more delicate human motion for recognition.

Table 4
Recognition accuracy of some other feature-based approaches on Weizmann database. Noted, the extracted features and the recognition strategy are organized for each approach, and all of these approaches use evaluation method of leaving one out cross validation.

Methods	Features	Recognition approach	Recognition accuracy (%)
Our approach	**Spatial–temporal vectors**	**LMNN+KNN**	**98.9**
Bregonzio et al. [16]	Clouds of interest points	Nearest neighbor classifier and support vector machine	96.6
Chaudhry et al. [6]	Histogram of oriented optical flow features	Non-linear dynamical systems	94.4
Lee and Chen [21]	Histogram-based interest points	Bhattacharyya coefficient measurement	84.4
Filipovych and Ribeiro [42]	2D and 3D Interest sub-regions	Pose models and motion dynamics model	88.9
Ali et al. [43]	Trajectories of body joints	Phase space embedding	92.6
Neibles and Li [12]	Spatial–temporal interest points	Hierarchical model	72.8

References

[1] R. Xiao, W. Li, Y. Tian, X. Tang, Joint boosting feature selection for robust face recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2006, pp. 1415–1422.

[2] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, Yi Ma, Robust face recognition via sparse representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2) (2009) 210–227.

[3] H. Zhou, P. Miller, J. Zhang, Age classification using radon transform and entropy based scaling SVM, in: Proceeding of the British Machine Vision Conference, 2011, pp. 28.1–28.12.

[4] L.K. Jia, D.Y. Yeung, Human action recognition using local spatio-temporal discriminant embedding, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[5] L. Wang, D. Suter, Visual learning and recognition of sequential data manifolds with applications to human movement analysis, Computer Vision and Image Understanding 110 (2) (2008) 152–172.

[6] R. Chaudhry, Avinash Ravichandran G. Hager, R. Vidal, Histograms of oriented optical flow and Binet–Cauchy kernels on nonlinear dynamical systems for the recognition of human actions, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1932–1939.

[7] A. Efros, A. Berg, G. Mori, J. Malik, Recognizing action at a distance, in: IEEE International Conference on Computer Vision, vol. 2, 2003, pp. 726–733.

[8] C. Schuldt, I. Laptev, B. Caputo, Recognizing human actions: a local SVM approach, IEEE International Conference on Pattern Recognition, vol. 3, 2004, pp. 32–36.

[9] S. Yan, D. Xu, B. Zhang, H.J. Zhang, Graph embedding and extensions: a general framework for dimensionality reduction, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (1) (2007) 40–51.

[10] A. Bissacco, A. Chiuso, Y. Ma, S. Soatto, Recognition of human gaits, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, 2001, pp. 52–57.

[11] C. Bregler, Learning and recognizing human dynamics in video sequences, in: IEEE Conference on Computer Vision and Pattern Recognition, 1997, pp. 568–574.

[12] J.C. Niebles, F.F. Li, A hierarchical model of shape and appearance for human action classification, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. 99, 2007, pp. 1–8.

[13] L. Wang, H.Z. Ning, T.N. Tan, W.M. Hu, Fusion of static and dynamic body biometrics for gait recognition, in: IEEE International Conference on Computer Vision, 2003, pp. 1449–1454.

[14] Y. Wang, G. Mori, Max-margin hidden conditional random fields for human action recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 872–879.

[15] Y. Yacoob, M.J. Black, Parameterized modeling and recognition of activities, Computer Vision and Image Understanding 73 (2) (1999) 232–247.

[16] M. Bregonzio, S. Gong, Tao Xiang, Recognising action as clouds of space–time interest points, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1948–1955.

[17] P. Dollar, V. Rabaud, G. Cottrell, S. Belongie, Behavior recognition via sparse spatio-temporal features, in: IEEE International Conference on Computer Vision, 2005, pp. 65–72.

[18] Y. Ke, R. Sukthankar, M. Hebert, Efficient visual event detection using volumetric features, in: IEEE International Conference on Computer Vision, 2005, pp. 166–173.

[19] I. Laptev, On space–time interest points, International Journal of Computer Vision 64 (2–3) (2005) 107–123.

[20] I. Laptev, M. Marszalek, C. Schmid, B. Rozenfeld, Learning realistic human actions from movies, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[21] W.T. Lee, H.T. Chen, Histogram-based interest point detectors, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1590–1596.

[22] K.Q. Weinberger, L.K. Saul, Distance metric learning for large margin nearest neighbor classification, Journal of Machine Learning Research 10 (2009) 209–244.

[23] A. Bobick, J. Davis, The recognition of human movement using temporal templates, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (3) (2001) 257–267.

[24] L. Gorelick, M. Blank, E. Shechtman, M. Irani, R. Basri, Action as space–time shapes, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (12) (2007) 2247–2253.

[25] R. Poppe, M. Poel, Discriminative human action recognition using pairwise CSP classifiers, in: IEEE International Conference on Automatic Face and Gesture Recognition, 2008, pp. 1–6.

[26] L. Wang, D. Suter, Recognizing human activities from silhouettes: motion subspace and factorial discriminative graphical model, in: IEEE International Conference on Pattern Recognition, 2007, pp. 1–8.

[27] L. Wang, D. Suter, Learning and matching of dynamic shape manifolds for human action recognition, IEEE Transactions on Image Processing 16 (6) (2007) 1646–1661.

[28] D. Weinland, Edmond Boyer, Action recognition using exemplar-based embedding, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–7.

[29] J.B. Tenenbaum, V.D. Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.

[30] S. Roweis, L. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2322–2326.

[31] M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering, Advances in Neural Information Processing Systems 14 (2002) 585–591.

[32] A. Elgammal, C.S. Lee, Inferring 3D body pose from silhouettes using activity manifold learning, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, 2004, pp. 681–688.

[33] X. He, P. Niyogi, Locality preserving projections, Advances in Neural Information Processing Systems 16 (2003) 152–160.

[34] D. Cai, X. He, K. Zhou, J. Han, H. Bao, Locality sensitive discriminant analysis, in: International Joint Conferences on Artificail Intelligence, 2007, pp. 708–713.

[35] R. Wang, X. Chen, Manifold discriminant analysis, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 429–436.

[36] 〈http://www.cam-orl.co.uk/facedatabase.html〉.

[37] 〈http://yann.lecun.com/exdb/mnist/index.html〉.

[38] M. Turk, A. Pentland, Face recognition using eigenfaces, in: IEEE International Conference on Pattern Recognition, 1991, pp. 586–591.

[39] P.N. Belhumeur, J.P. Hepanha, D.J. Kriegman, Eigenfaces vs. Fisherfaces: recognition using class specific linear projection, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7) (1997) 711–720.

[40] X. Wu, Y. Jia, W. Liang, Incremental discriminant-analysis of canonical correlations for action recognition, Pattern Recognition 43 (12) (2010) 4190–4197.

[41] J. Zhang, S. Gong, Action categorization with modified hidden conditional random field, Pattern Recognition 43 (1) (2010) 197–203.

[42] R. Filipovych, E. Ribeiro, Learning human motion models from unsegmented videos, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–7.

[43] S. Ali, A. Basharat, M. Shah, Chaotic invariants for human action recognition, in: IEEE International Conference on Computer Vision, 2007, pp. 1–8.

Chien-Chung Tseng received his B.S. degree in computer science and information engineering from the National Cheng Kung University, Tainan, Taiwan, in 2004. He now is a Ph.D. candidate in computer science and information engineering at the National Cheng Kung University, Tainan, Taiwan. In addition to his current research into license plate recognition and human action recognition, his interests lie in automatic caricature generation, event classification, computer vision, and pattern recognition.

Ju-Chin Chen received her B.S., M.S. and Ph.D. degrees in computer science and information engineering from the National Cheng Kung University, Tainan, Taiwan, in 2002, 2004 and 2010, respectively. She is now an assistant professor in the Department of Computer Science and Information Engineering at the National Kaohsiung University of Applied Science, Taiwan. Her research interests lie in the fields of machine learning, computer vision and pattern recognition.

Ching-Hsien Fang received his B.S. degree in computer science and information engineering form the National Tsing Hua University, Hsinchu, Taiwan, in 2008. Then, he received his M.S. degrees in computer science and information engineering from the National Cheng Kung University, Tainan, Taiwan, in 2010. He is now a engineer in Cyberlink which advances and innovates video and audio technology for the people's enjoyment. In addition to his current research into automatic face detection, tracking and recognition, his interests lie in the fields of digital image processing, computer vision and pattern recognition.

Jenn-Jier James Lien (M'00) received his M.S. and Ph.D. degrees in electrical engineering from Washington University, St. Louis, MO, and the University of Pittsburgh, Pittsburgh, PA, in 1993 and 1998, respectively. From 1995 to 1998, he was a research assistant at the Vision Autonomous Systems Center in the Robotics Institute at Carnegie Mellon University, Pittsburgh, PA. From 1999 to 2002, he was a senior research scientist at L1-Identity (formerly Visionics) and a project lead for the DARPA surveillance project. He is now an associate professor in the Department of Computer Science and Information Engineering at the National Cheng Kung University, Taiwan.