Action Recognition Using Graph Embedding and the Co-occurrence Matrices Descriptor

Feng Zheng, Ling Shao, Zhan Song, Xi Chen

▶ **To cite this version:**

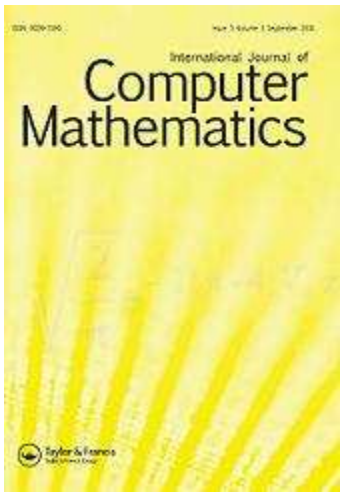
Feng Zheng, Ling Shao, Zhan Song, Xi Chen. Action Recognition Using Graph Embedding and the Co-occurrence Matrices Descriptor. International Journal of Computer Mathematics, 2011, 10.1080/00207160.2011.578741 . hal-00711298

HAL Id: hal-00711298
https://hal.science/hal-00711298

Submitted on 23 Jun 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Action Recognition Using Graph Embedding and the Co-occurrence Matrices Descriptor

Journal:	*International Journal of Computer Mathematics*
Manuscript ID:	GCOM-2011-0086-A.R1
Manuscript Type:	Original Article
Date Submitted by the Author:	15-Mar-2011
Complete List of Authors:	Zheng, Feng; Chinese Academy of Sciences, SIAT Shao, Ling; The University of Sheffield, Electronic and Electrical Engineering Song, Zhan; Chinese Academy of Sciences, SIAT Chen, Xi; Chinese Academy of Sciences, SIAT
Keywords:	Action recognition, Graph Embedding, Co-occurrence matrices, Human silhouette, Bag of words

SCHOLARONE™
Manuscripts

International Journal of Computer Mathematics
Vol. 00, No. 00, January 2010, 1–19

RESEARCH ARTICLE

Action Recognition Using Graph Embedding and the Co-occurrence Matrices Descriptor

Feng Zheng[a,b] Ling Shao[c*] Zhan Song[a,b*] and Xi Chen[a,b]

[a] *Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China;*

[b] *The Chinese University of Hong Kong, Hong Kong, China;*

[c] *Department of Electronic and Electrical Engineering, University of Sheffield, UK.*

(*Received 00 Month 200x; in final form 00 Month 200x*)

Recognizing actions from a monocular video is a very hot topic in computer vision recently. In this paper, we propose a new representation of actions, the co-occurrence matrices descriptor, on the intrinsic shape manifold learned by graph embedding. The co-occurrence matrices descriptor captures more temporal information than the bag of words (histogram) descriptor which only considers the spatial information, thus boosts the classification accuracy. In addition, we compare the performance of the co-occurrence matrices descriptor on different manifolds learned by various graph embedding methods. Graph embedding methods preserve as much of the significant structure of the high-dimensional data as possible in the low-dimensional map. The results show that nonlinear algorithms are more robust than linear ones. Furthermore, we conclude that the label information plays a critical role in learning more discriminating manifolds.

Keywords: Action recognition, graph embedding, co-occurrence matrices, human silhouette, bag of words.

AMS Subject Classification: 68Txx; 68Uxx **Machine vision and scene understanding, Learning and adaptive systems, Knowledge representation; Image processing**

1. Introduction

Action recognition is an important area in computer vision, which has many fundamental applications in video surveillance, behavior understanding and human computer interaction. Although the applications all have their own special demands, they should always cope with spatial and temporal differences in performing actions as well as handle variations in the observed data due to difficult environment conditions [6]. Because of these difficulties, action recognition on real-world data is always a challenging task.

Many action recognition algorithms rely on partial representations, i.e. spatio-temporal interest points based. Those methods are usually more robust to occlusions and viewpoint changes but lack global information of the actions performed. In this paper, we focus on the global representation of human actions. Generally, there are two major groups of approaches to deal with spatial and temporal differences in global action representations. The first common method is to build

*Corresponding author. Email: ling.shao@sheffield.ac.uk; zhan.song@siat.ac.cn

ISSN: 0020-7160 print/ISSN 1029-0265 online
© 2010 Taylor & Francis
DOI: 0020716YYxxxxxxxx
http://www.informaworld.com

For Peer Review Only

templates of activities, including two dimensional images: Motion Energy Images (MEI) and Motion History Images (MHI) [4], and three dimensional space time volumes: Motion History Volumes (MHV) [18, 29] and Spatiotemporal Volume (STV) [3, 31]. These methods use the energy of images or interest points and the geometry of volume to describe different actions.

The other branch of significant and useful methods consider a video sequence as a set of raw data vectors and adopt a graph embedding method to learn the dynamic shape manifolds. The silhouette (binary) images of an action performed by some persons undergoing a smooth motion can be represented as a manifold in the image space. Manifold learning transforms dimensionality of objective data from high to low by mapping similar input data to nearby points on a manifold, preserving as much of the significant structure of the high-dimensional data as possible in the low-dimensional map. Among the linear algorithms, i.e. the subspace learning algorithms, principal component analysis (PCA) [16] and linear discriminant analysis (LDA) [8] are the two most popular ones and both were proposed with Gaussian assumptions on data distributions. Another popular linear algorithm is locality preserving projections (LPP) [14]. In [25], the geometric mean for subspace selection was studied to overcome a drawback of LDA: those classes being close in the original feature space tend to merge together. A method [2]: max-min distance analysis (MMDA), which separateed all class pairs and could be achieved by maximizing the minimum pairwise distance in the selected subspace, had been proposed for discriminative dimension reduction algorithm and also been extended into its kernel version.

Unfortunately, they have a common inherent limitation: they are all linear methods while the distributions of most real data are nonlinear. As for nonlinear dimensionality reduction algorithms, the representative methods include isometric feature mapping (Isomap) [27], locally linear embedding (LLE) [21], Laplacian Eigenmaps (LE) [1], diffusion maps [7] and semi-supervised diffusion maps (SSDM) [35]. These algorithms are generally named as manifold learning which is an emerging and promising approach in nonlinear dimensionality reduction. A manifold is a topological space that is locally Euclidean. LLE [21] and LE [1] focus on the preservation of local neighbor structure. Isomap [27] seeks the subspace that best preserves the geodesic distances between any two data sets. Diffusion maps [7] relates the spectral properties of Markov processes to their geometric counterparts and preserve the diffusion distance in intrinsic space. Furthermore, some methods preserving the manifold structure were proposed for some special tasks. SSDM [35] is a semi-supervised method which preserves the local manifold structure in addition to separating samples in different classes, thus facilitates the classification. In [11], a supervised Gaussian process latent variable model (GP-LVM) was developed for supervised learning tasks, and the maximum a posteriori algorithm was introduced to estimate positions of all samples in the latent variable space. Zhou [36] proposed the manifold elastic net to incorporate the merits of both the manifold learning based dimensionality reduction and the sparse learning based dimensionality reduction. In [12], the authors introduce the manifold regularization and the margin maximization to non-negative matrix factorization and obtain the manifold regularized discriminative non-negative matrix factorization. In [32], Zhang et. al proposed a framework, named "patch alignment", which consists of two stages: part optimization and whole alignment to provide a systematic framework for understanding the common properties and intrinsic difference in different algorithms.

Many interesting methods for learning a compact action representation using manifold leaning algorithms have been proposed. Elgammal and Lee [10] adopted

For Peer Review Only

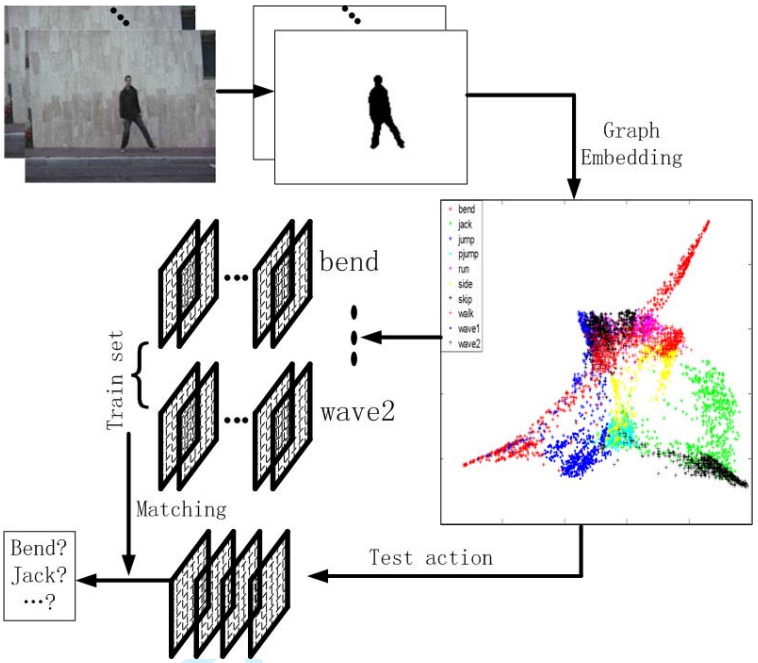


Figure 1. The flow diagram of the proposed method. The square frames denote the co-occurrence matrices. The two dimensional manifold is learned by diffusion maps.

an LLE framework to embed activity manifolds nonlinearly into a low dimensional space for inferring 3D body pose from silhouettes. Dimension reduction of the image silhouette and pose information had also been investigated using kernel principle component analysis (KPCA) in [28]. Jia and Yeung [15] presented their Local Spatio-Temporal Discriminant Embedding (LSTDE) algorithm specially designed for human action recognition. Chin et al. [6] investigated and compared methods for the extrapolation of learned manifolds within the context of activity recognition. In [19], a neighborhood preserving embedding (NPE) algorithm was exploited to embed silhouette images into an intrinsic manifold. Souvenir et al. [23] presented a framework for learning a compact representation of primitive actions that can be used for simultaneous action recognition and viewpoint estimation. Shao and Chen [22] explored Spectral Regression Discriminant Analysis (SRDA) on silhouette based human action recognition. The recognition algorithm adopted the Bag of Words (BoW) model combined with the action representation based on histogram of body poses sampled from silhouettes in the video sequence. Zheng et al. [33] used the semi-supervised diffusion maps (SSDM) to learn the intrinsic shape manifold. And the action was represented by bag of words (histogram) on the learned shape manifold. Tao et al. [26] develop a general tensor discriminant analysis (GTDA) as a preprocessing step for LDA and used human gait recognition to validate the proposed GTDA.

After learning the intrinsic manifold of human silhouettes, how to represent the actions on the intrinsic manifold becomes an important issue. The representation of actions or the framework of recognition must capture the spatial and temporal relations between poses simultaneously. Take actions “sitting down” and “getting up” for example, they are both composed of similar poses. One is performed in a downward direction and another upward, another just the reverse. Only the order of poses is different. If we only consider the spatial information in the learned manifold, such as key frames or bag of words, the two will be classified as one action. In this paper, we present a framework that combines manifold learning and the co-

Peer Review Only

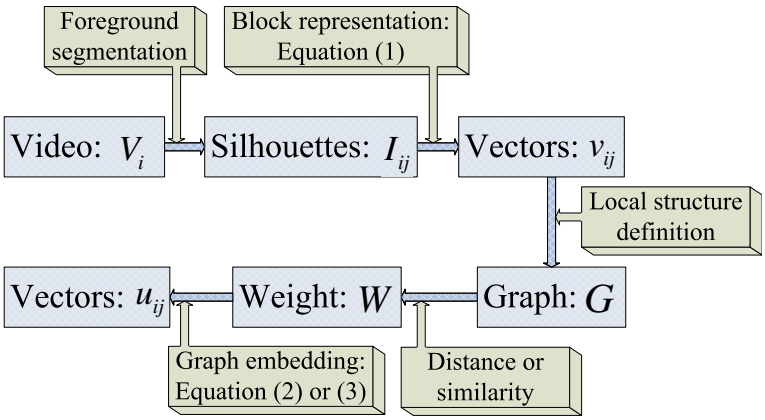


Figure 2. The flow diagram of the low dimensional graph embedding.

occurrence matrices descriptor for action recognition. Firstly, we learn a compact representation using manifold leaning for human silhouettes. Such a representation reflects the intrinsic relationship of points, such as local structure, geodesic distance and diffusion distance. We consider an action manifold as a trajectory with smooth variation on the learned manifold. Then, a set of co-occurrence matrices is used to describe an action sequence. Assuming that the sampling rate of the camera is fixed, the temporal difference of same actions is mainly caused by the speed variation in the actions performed by different subjects. The dimensionality of co-occurrence matrices of each action sequence can be reduced by PCA. Finally, we use the K Nearest Neighbors (KNN) classifier to label the test actions. The flow diagram of the proposed method is shown in Figure 1. We can see that poses of the same action performed by different subjects lie close to each other in the learned low dimensional manifold. The initial results of the co-occurrence matrices descriptor were presented in [34].

In general, our paper has two main contributions as follows:

(i) We propose a new action representation using co-occurrence matrices of human pose silhouettes, which can characterize the spatial-temporal properties of actions. The experimental results demonstrate that co-occurrence matrices outperform the bag of words descriptor. And we give a detail explanation why the co-occurrence matrices descriptor is supervisor to the bag of words descriptor.

(ii) We compare different graph embedding algorithms for learning the intrinsic manifold of silhouettes. The results show that the nonlinear graph embedding is more suitable for human action recognition. In addition, we conclude that the label information is useful for classification.

The rest of this paper is organized as follows. We introduce the silhouette representation and graph embedding methods for learning the intrinsic manifold of human silhouettes in Section 2. In Section 3, we present a set of co-occurrence matrices for action representation. The experimental results and comparison are given in Section 4. Finally, Section 5 draws the conclusion.

2. Learning the Intrinsic Manifold of Human Silhouettes

In this section, we introduce a silhouette representation and graph embedding methods to learn the shape manifolds. The flow diagram of the low dimensional graph embedding is shown in Figure 2. To begin with, in order to avoid the loss of

For Peer Review Only

information, each silhouette I_{ij} contained in video V_i is described as a vector v_{ij} based on combining the block-based features and the Radon transform descriptor. Then a graph G is constructed using the local geometric and label information, and a weight W is set to each edge on the graph. In addition, to avoid the "curse of dimensionality", we use a framework of graph based methods to embed high dimensional descriptors into the low dimensional space represented as u_{ij}. Finally, each action sequence is represented as a set of points on the intrinsic shape manifold $A_i = \{u_{i1}, \cdots, u_{iN_i}\}$, where N_i denotes the number of silhouettes contained in video V_i.

2.1 *Silhouette Representation*

The block-based features are always used in object recognition, such as character or pose [28]. The Radon transform descriptor [24] is a region-based descriptor calculating on a shape as a whole taking into account all the pixels within the shape. Assume that there are action videos in the data set. For each action video with N_i images, i.e. V_i, we assume that the associated sequence of moving silhouettes $V_i = \{I_{i1}, I_{i1}, \cdots, I_{iN_i}\}$, which are discrete binary image data, can be obtained from the original video. With a static background, silhouettes can be easily extracted from the video using existing algorithms. We firstly center and normalize all silhouette frames into the same dimension and keep the aspect ratio property of the silhouette, so that the resulting images contain as much information as possible and have no distortions. We assume that the block-based features of each silhouette image I_{ij} are given by: $v_{ij}^b = (b_{ij}(1), b_{ij}(2), \cdots, b_{ij}(n_b))$, where n_b is the number of sub-blocks (for more details refer to [28]).

The Radon transform descriptor is invariant to common geometrical transformations and converts a silhouette image to a compact signal through the use of the two-dimensional Radon transform. By definition, the Radon transform of an image is determined by a set of projections of the image along lines taken at different angles. The transform extends the Radon transform by calculating the sum of the squared Radon transform values for all of the lines of the same angle in an image. Assume that the Radon transform descriptor of each silhouette image is given by: $v_{ij}^r = (r_{ij}(1), r_{ij}(2), \cdots, r_{ij}(n_r))$, where n_r is the number of sub-blocks. By combining the block-based features and the Radon transform descriptor together, we can denote the silhouette image as:

$$v_{ij} = (v_{ij}^b, v_{ij}^r) = (b_{ij}(1), b_{ij}(2), \cdots, b_{ij}(n_b), r_{ij}(1), r_{ij}(2), \cdots, r_{ij}(n_r)) \quad (1)$$

2.2 *Graph Embedding Methods to Learn the Dynamic Shape Manifolds*

For learning the intrinsic shape manifold, we put all silhouette images sampled from different action videos together. For simplicity, by using the silhouette representation introduced in foresection 2.1, assume that $X = \{x_1, x_2, \cdots, x_N : x_i \in R^D\}$ is the training-set of poses drawn from video sequences containing different actions, where $N = \sum_i N_i$ and $D = n_b + n_r$. Then our goal is to find a mapping function $F : X \to Y, Y = \{y_i : y_i \in R^d, y_i = F(x_i)\}$ from an original space to the desired low-dimensional representation in R^d, where $d \ll D$. The function may be explicit or implicit, linear, or nonlinear in different cases. In [30], a general framework of graph embedding is first introduced which aims at embedding the high dimensional data points on a graph to a low dimensional intrinsic manifold. In this section, we use the graph embedding to exploit the shape manifold.

For Peer Review Only

2.2.1 Constructing a Graph

Given the data set of silhouettes, we can construct a non-directional graph in $G = (X, W)$ with the point on this graph corresponding to a feature vector of silhouette in the data set. The edges between nodes reflect the neighborhood relations along the shape manifold. For each edge (x_i, x_j), we set a weight between the two points x_i and x_j at the end of the edge. Each element of the real symmetric matrix W measures, for a pair of vertices, its similarity. The matrix can be formed using various similarity criteria, such as Gaussian similarity from Euclidean distance, local neighborhood relationship, and prior class information in supervised learning algorithms [30].

2.2.2 Objective Function and Eigenmaps

Although the silhouette image can be represented as a vector, because of the quite high dimension of such a representation, the following tasks will face many difficulties. Furthermore, the perceptually meaningful structure of these images has significantly fewer independent degrees of freedom. We desire a new representation for each image with a low dimension preserving as much information as possible. To the best of our knowledge, there are two methods to define how to preserve the information. In fact, two different objective functions are corresponding to the two different ways to define the weight of graph $G = (X, W)$.

One is to find an embedding of the data points that best preserves the inter-point distances. The quantity distance can de defined by various ways, such as Euclidean distance [5], geodesic distance [27] and diffusion distance [7]. If the distance between data points x_i and x_j in the high original space is denoted by $W^D(x_i, x_j)$, and the distance in the learned low dimensional space is denoted by $W^d(y_i, y_j)$ where $y_i = F(x_i)$, then the objective function can be defined by:

$$Y^* = \arg\min \parallel \tau(W^D) - \tau(W^d) \parallel_{L^2}, \tag{2}$$

where $\| \cdot \|_{L^2}$ is the L^2 matrix norm, and the τ operator converts distances to inner products or nothing operation. Then the weight matrix of graph G is defined by $W = W^D$. To preserve different distances, there are different tricks to solve this optimization problem. For multi-dimensional scaling (MDS) [5] and ISOMAP, let λ_k be the kth eigenvalue (in decreasing order) of the matrix $\tau(W^D)$, and $\phi_k(i)$ be the ith component of the kth eigenvector. Then set the kth component of the $d-$dimensional coordinate vector $y_i(k)$ equal to $\sqrt{\lambda_k}\upsilon_k(i)$. For DM and SSDM, if $P_t(x_i, x_j)$ represents the probability of going from x_i to x_j in the time t steps on the graph G, the distance $W^D(x_i, x_j)$ in the high original space is defined by quantity P_t. Let column vectors $\phi_0, \phi_2, \cdots, \phi_{N-1}$ be the eigenvectors of the tth steps probability matrix P_t, ordered according to their eigenvalues $\lambda_0 \geq \lambda_1 \geq \cdots \geq \lambda_{N-1}$. Thus, the embedding is as follows: $y_i = (\lambda_1^t \phi_1(i), \lambda_2^t \phi_2(i), \cdots, \lambda_d^t \phi_d(i))^T$.

Another method hopes that the points which are close in the high dimensional space are also close in the learned manifold. The important thing is to define the similarity to describe how "close" each pair of data points. The quantity "similarity" is related to "distance" introduced in the first method. They can convert between each other. If W^S is a similarity matrix, then the objective function is given by:

$$Y^* = \arg \min_{Y^T B Y = \varrho} \sum_{i \neq j} \parallel y_i - y_j \parallel^2 W^S(x_i, x_j) = \arg \min_{Y^T B Y = \varrho} tr(Y^T L Y), \tag{3}$$

where L is a Laplacian matrix of graph G, B is the constrain matrix and ϱ is a

For Peer Review Only

constant. If E is a diagonal matrix, $E_{ii} = \sum_j W^S_{ij}$, then $L = E - W^S$. Then the weight matrix of graph G is defined by $W = W^S$. This problem can be converted into a generalized eigenvalue problem. Let column vectors $\phi_0, \phi_2, \cdots, \phi_{N-1}$ be the eigenvectors of the generalized eigenvalue problem $E^{-1}L$, ordered according to their eigenvalues $\lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_{N-1}$. Thus, the embedding is as follows: $y_i = (\lambda_1\phi_1(i), \lambda_2\phi_2(i), \cdots, \lambda_d\phi_d(i))^T$.

Therefore, we can see that various methods are attributed to the distance and similarity of the different definitions. How to define the distance and similarity will be discussed in the following subsection.

2.2.3 Various Distance Matrices

MDS is a group of methods that have a wide range of applications. The key idea is to find a mapping from a high-dimensional space to a low-dimensional space, such that the pairwise distances between the observed points are preserved the best. In the framework of MDS [5], the distance matrixes $W^d(y_i, y_j)$ and $W^D(y_i, y_j)$ are all defined by the Euclidean distance. Isomap [27] was proposed to find the low-dimensional representations for a data set by approximately preserving the geodesic distances of the data pairs. Thus, the distance matrixes W^d in the learned eigenspace is defined by the Euclidean distance. The geodesic distance is approximated by the shortest distance between each pair of points on the graph G.

Diffusion maps were firstly introduced in [7]. It is an unsupervised nonlinear graph embedding algorithm for dimensionality reduction. It is defined on a graph of the objective data via Markov random walk. The low-dimensional representation of the data is achieved by embedding the high dimensional data space to a low dimensional Euclidean space using a set of eigenvectors corresponding to the largest eigenvalues of Markov matrices. Semi-supervised diffusion maps first introduced in [35] is a semi-supervised extension of diffusion maps. The Markov matrices are adjusted by a distributional similarity learned through Expectation Maximization (EM). The adjusted matrix not only reflects the geometric structure of the human action manifold, but also the label information of classification.

For DM and SSDM, the distance $W^D(x_i, x_j)$ is defined by the t step probability matrix of transition P_t, where tth is a parameter.

$$W^D(x_i, x_j) = ||P_t(x_i, \cdot) - P_t(x_j, \cdot)||_{1/\pi}, \tag{4}$$

where π is a stationary distribution for measuring the density of each points and $P_t(x_i, x_j)$ represents the probability of going from x_i to x_j in the time t steps on the graph G. The major difference between DM and SSDM is the definition of transition probability matrix. For SSDM, this quantity reflects the label information of each point.

2.2.4 Various Similarity Matrices

In this subsection, we briefly introduce the preserved similarities of various dimensionality reduction algorithms reformulated within the second framework. The details of this work can be found in [1]. The difference between various algorithms lies in the computation of the similarity matrix of the graph and the selection of the constraint matrix.

Linear algorithms assume that the data points draw from the unknown Gaussian distribution. PCA [16] seeks projection directions with maximal variances. In other words, it finds and removes the projection directions with minimal variances. So the similarities between all pair of points are equal. LDA [8] is a supervised algorithm which searches for the directions that are most effective for discrimination by minimizing the ratio between the intraclass and interclass scatters. Then the

For Peer Review Only

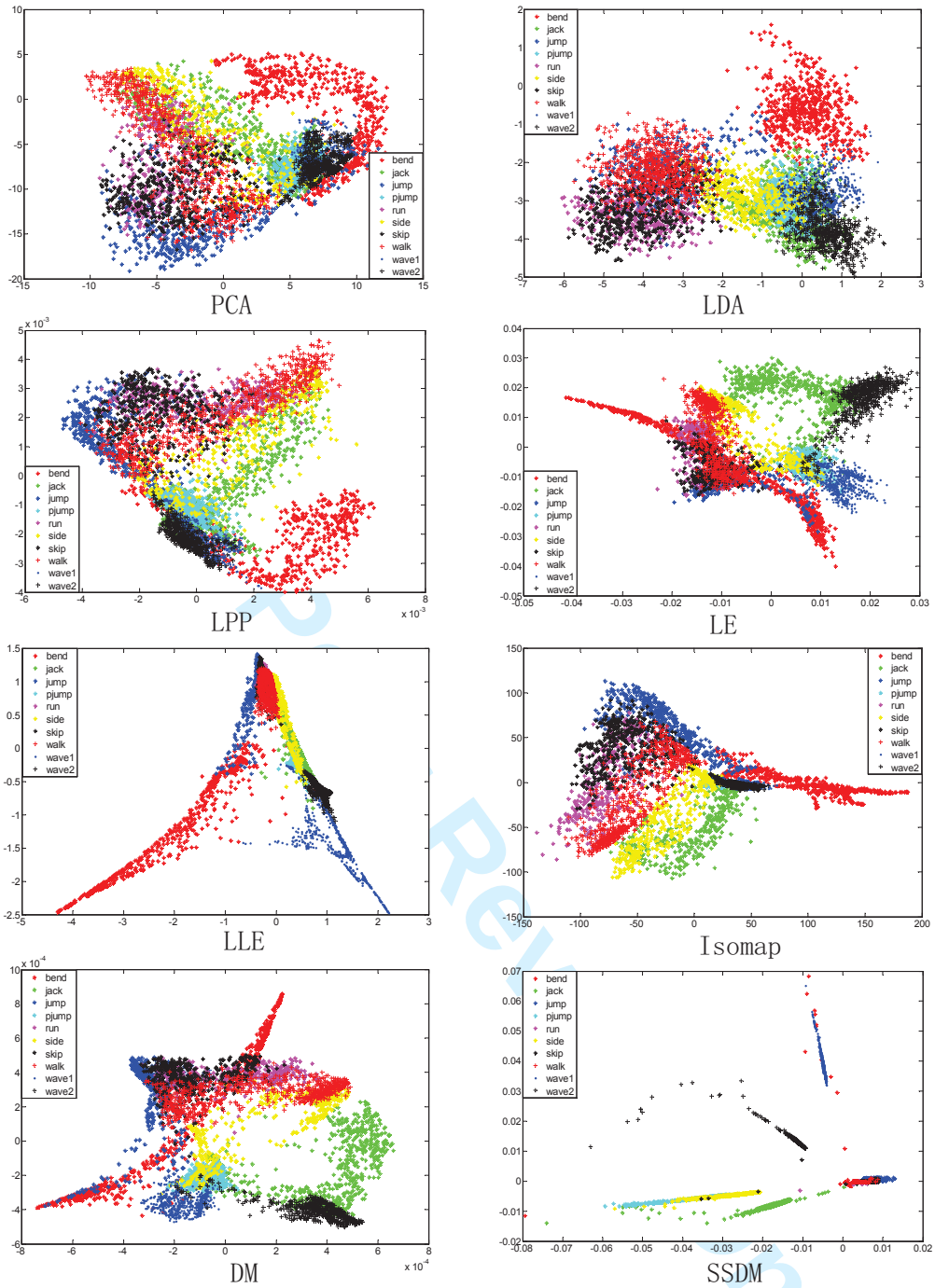


Figure 3. The two dimensional visualization of human silhouettes. The data set for LDA contains 8 subjects' actions which are all labeled. The data set for SSDM contains 9 subjects' actions with 8 of them labeled. The parameter of local structure K for LPP, LE, LLE, Isomap, DM, and SSDM is set to 35. The transition step t in algorithms DM and SSDM is set to 8.

similarity between the two points with the same label is far larger than the points with different labels. These linear algorithms set the same similarity for all pairs of points in a subset or universal set of all samples. However, the intrinsic relation may be local in some cases. Local relationship is applied in the nonlinear algorithms, which can be defined in terms of two variations: ε-neighborhoods and K nearest neighbors. LLE [21] maps the input data to a lower dimensional space in a manner that preserves the relationship between the neighboring points. In LLE, the local

similarity of the data manifold is constructed by writing the data points as a linear combination of their nearest neighbors. Laplacian Eigenmaps (LE) preserves the similarities of the neighboring points which are defined by a Gaussian kernel.

3. Co-occurrence Matrices for Action Recognition

For each silhouette, we get a low dimensional representation $y_i = \mathcal{M}(x_i)$,using the graph embedding algorithms (refer to Fig. 3). Therefore, each action sequence can be represented as:

$$A_i = \{u_{i1}, u_{i2}, \cdots, u_{iN_i}\} = \mathcal{M}\{v_{i1}, v_{i2}, \cdots, v_{iN_i}\} \tag{5}$$

For matching action sequences A_i and A_j, we have to compute the similarity between the two sets of vectors. However, in practice, there exist spatial and temporal mis-alignments between action sequences. In addition, the action sequences may be different in length and in frequency, which further complicates the problem. In this paper, we propose a model-based matching framework for action recognition. We explore a set of co-occurrence matrices to represent each action. The co-occurrence matrices can capture the spatial-temporal difference of actions performed by different subjects.

3.1 *Related Work for Action Recognition on Learned Manifold*

The data point of each test silhouette frame is first projected into the embedding space using graph embedding algorithms, which results in a low-dimensional representation of the data point. For recognizing actions, there are different frameworks to make use of the temporal shape variation information between different actions. In [28], Wang and Suter assumed that a set of key poses can represent an action, then explored factorial conditional random field (FCRF) to label human activity sequences in the embedded space. FCRF models temporal sequences in multiple interacting ways. In [19], a hidden conditional random field (HCRF) was exploited to model and classify actions in a discriminative formulation. Jia and Yeung [15] designed a two-stage recognition scheme. In the first stage, the test action sequence was recognized on a frame-by-frame basis. 7-frame local temporal video segments were used in the second recognition stage. In [6], manifold comparison can be performed by computing distances between trajectories in the embedding space while respecting differences in sequence length and temporal shifts. A histogram representation calculated on probabilities of different poses was proposed in [33].

3.2 *Co-occurrence Matrices Descriptor*

A co-occurrence matrix is a matrix that is defined over an image to be the distribution of co-occurring values at a given offset. The co-occurrence matrix can measure the texture of the image. Haralick et al. proposed the algorithm of gray-level co-occurrence matrix (GLCM) in 1970s [13]. GLCM counts a co-appearance probability of $p(i, j, \delta)$, when the distance between gray-level for the i pixel (p, q) and gray-level for the j pixel $(p + \Delta p, q + \Delta q)$ is δ. Moreover, it can counts the co-appearance probability in some special situation with a fixed direction.

Inspired by GLCM used in the image texture classification, we use a set of co-occurrence matrices to represent the shape manifold in the low dimensional space. If dividing the low dimensional space into subsets, we can consider each subset as

For Peer Review Only

a grid in an image. Then we can use the GLCM method to describe each action reflecting the spacial and temporal information. Suppose a codebook is learned by running k-means clustering on the eigen-space for all silhouettes in the training set. $\mathcal{C} = |c_1, c_2, \cdots, c_{N_0}|$ represents a codebook, where N_0 is the number of codevectors. Each codevector is a d dimensional vector. c_i can be regarded as the centre of each subset in the learned low space. Let $S_1, \cdots, S_{N_0}$ denote this partition of the space, where S_l is the encoding region associated with codevector c_l.

We count the co-appearance probability of $p(c_l, c_m, \nabla t|A_i)$, when the first pose u_{ij} belongs to partition S_l, and the second pose $u_{i(j+\nabla t)}$ after ∇t frames belongs to S_m in the same action video A_i. The probability of points u_{ij} belonging to S_l can be defined as:

$$w_{i,j}(l) = \begin{cases} exp(-\|u_{ij} - c_l\|^2/2\sigma^2), & \text{if } u_{ij} \in S_l; \\ 0, & \text{otherwise,} \end{cases} \tag{6}$$

Histograms are built by allowing low dimensional vectors to vote softly into the few centers nearest to them, with Gaussian weights.

$$\alpha_i(l) = \frac{1}{N_i} \sum_{j=1}^{N_i} w_{i,j}(l) \tag{7}$$

Therefore, using the histogram, an action can be denoted as a vector:

$$A_i^{\alpha} = \alpha_i = \{\alpha_i(1), \alpha_i(2), \cdots, \alpha_i(N_0)\} \tag{8}$$

Action comparison is thus finally reduced to the comparison of N_0 dimensional histograms. The bag of words (histogram) descriptors only build the distribution of poses in the eigen-space. However, there are spatial and temporal differences in actions. For example, some poses of human walking is very similar to the poses of human running, but these poses are performed with different speed. So using the co-occurrence matrices, we can design a descriptor for actions capturing the temporal mutual information. The co-occurrence matrix can be defined as:

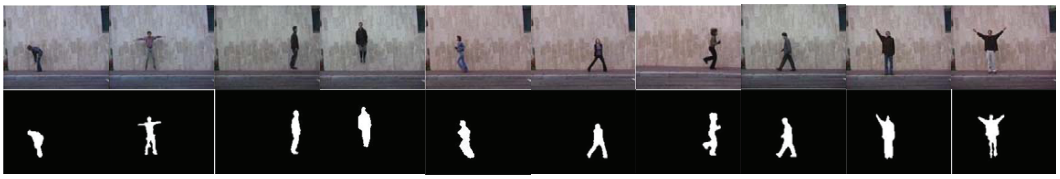
$$\beta_i(l, m, \nabla t) = \sum_{j, j+\nabla t} w_{i,j}(l) w_{i,j+\nabla t}(m) \tag{9}$$

Then, each action can be denoted by a set of co-occurrence matrices modeled from the training set by varying the parameter ∇t.

$$A_i^{\beta} = \beta_i = \{\beta_i(1,1), \beta_i(2,1), \cdots, \beta_i(N_0,1), \cdots, \beta_i(N_0,N_0)\} \tag{10}$$

As the matrix set for each action is in high dimension and is redundant, we use PCA to reduce the dimensionality retaining 95% information. During testing, each action sequence can be represented in the same way as training. In the procedure of recognition, we can directly match each test action video to the train templates. Also, we can first learn a classifier for training templates, then test action videos online using such a classifier.

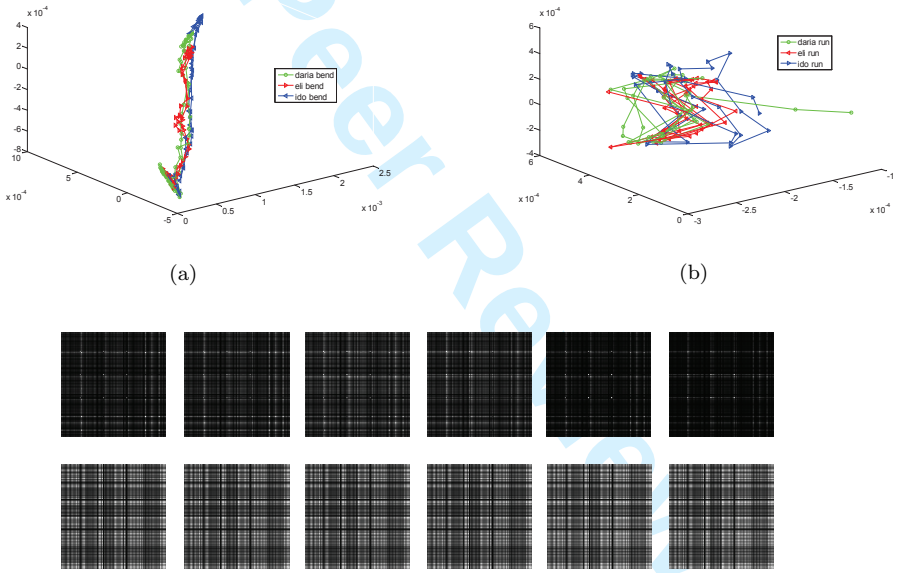
For Peer Review Only



(a)

(b)

Figure 4. (a) Normalized frames and silhouettes of different actions. (b) Example frames and silhouettes of walking sequences for robustness test.



(a) (b)

(c)

Figure 5. (a) The three dimensional visualization of action "bend" performed by Daria, Eli and Ido. (b) The three dimensional visualization of action "run" performed by Daria, Eli and Ido. (c) Up: co-occurrence matrices of "bend" corresponding to the trajectories in Panel (a) with $\nabla t = 2, 10$. Bottom: co-occurrence matrices of "run" corresponding to the trajectories in Panel (b) with $\nabla t = 2, 10$.

4. Experiments

The proposed algorithm is experimented on a number of public data sets to test its performance, such as the Weizmann dataset [3] and IXMAS [29]. In the experiments, we compare our algorithm with classical algorithms, including PCA, LDA, LPP, LE, LLE, Isomap, DM and SSDM.

For Peer Review Only

Table 1. Accuracy Comparison Between the Bag of Words and the Co-occurrence Matrices on the Learned Manifold by Various Algorithms

Methods	PCA	LDA	LPP	LE	LLE	Isomap	DM	SSDM
Bag of Words	92.11%	92.22%	93.11%	94.67%	94.56%	96.67%	97.00%	98.60%
Co-occurrence matrices	94.33%	95.33%	96.00%	96.56%	98.22%	96.56%	98.89%	99.44%

4.1 *Weizmann dataset*

The experiments presented in this section are carried out on a widely used dataset [3]. Specifically, the dataset contains 93 video sequences with ten different actions performed by nine different subjects. The action types include bending (bend), jumping jack (jack), jumping-forward-on-two-legs (jump), jumping-in-place (pjump), running (run), galloping sideway (side), skipping (skip), walking (walk), waving-one-hand (wave1), waving-two-hand (wave2). The example frames and silhouettes are shown in Fig. 4.

4.1.1 Action Recognition Accuracy on Weizmann dataset

In this work, we do not intend to exploit the foreground segmentation issue, so the silhouette masks in [3] are directly used in our experiments. We firstly denote each silhouette as a vector using Radon transform and block-based features, and then the eigen-space of the silhouettes is learned by the graph embedding algorithms. Figure 3 shows that the $2D$ illustration of manifold learning by various different algorithms. For supervised algorithms (LDA), we use eight subjects to train the projection function. For semi-supervised algorithms (SSDM), we set all subjects for training with eight subjects labeled and one subject unlabeled. We can see that nonlinear algorithms learn a more discriminating manifold than linear algorithms. In Figure 5, we show two sets of co-occurrence matrices of actions "bend" and "run". It's obvious that the two sets of matrices have large difference. However, the co-occurrence matrices of same actions performed by different subjects are similar. So, we can use these co-occurrence matrices as an effective means for representing actions. Recognition accuracy comparison between histogram and co-occurrence matrices on the learned manifold by various algorithms is shown in Table 1. Since the co-occurrence matrices descriptor captures the temporal relationship information, its classification accuracy shows to be significantly higher than that of the histogram descriptor. At the same time, from Table 1, we can see that linear methods (PCA, LDA and LPP) are not as good as nonlinear methods (LE, LLE, ISOMAP, DM and SSDM), no matter which descriptor is used. This result also demonstrates that the relationship of silhouettes of different actions is not linear in the original space. Furthermore, since the method SSDM is more advantageous over other methods, it illustrates that the label information plays an important role in learning a more discriminative representation.

4.1.2 Robustness Test for the Co-occurrence Matrices Descriptor

The recognition of human actions can be further challenged when action sequences are captured in front of non-uniform backgrounds, with partial occlusions and non-rigid deformations, at changing viewpoints, etc. In order to evaluate the robustness of our method to these high irregularities of real-world actions, we perform further experiments using 20 video sequences of people walking in various difficult scenarios and changing viewpoints [3]. In particular, these videos include diagonal walking, walking with a dog, walking when swinging a bag, walking in a skirt, walking with partially occluded legs, sleepwalking, limping, walking with knees up, walking when carrying a briefcase, normal walking and changing viewpoints from 0° to 45°.

For Peer Review Only

Table 2. Recognition Results of the Robustness Test

Methods	PCA			LDA			LPP			LE		
Cases	1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd
View 0°	8	8	8	8	8	8	8	8	8	8	8	8
View 5°	8	8	8	8	8	8	8	8	8	8	8	8
View 10°	8	8	8	8	8	8	8	8	8	8	8	8
View 15°	8	8	8	8	8	8	8	8	8	8	8	8
View 20°	8	8	8	8	8	8	8	8	8	8	8	8
View 25°	8	8	8	**6**	**6**	**6**	8	8	8	8	8	8
View 30°	**9**	**9**	**9**	**6**	**6**	**6**	**7**	**7**	8	8	8	8
View 35°	**9**	**9**	**9**	**6**	**6**	**6**	**9**	**9**	**9**	**4**	**4**	**4**
View 40°	**4**	**9**	**4**	**6**	**6**	**6**	**4**	**4**	**4**	**4**	**4**	**4**
View 45°	+	+	+	**6**	**6**	**6**	**4**	**4**	**4**	**4**	**4**	**4**
Swinging bag	**1**	**1**	**1**	8	8	8	8	8	**5**	8	8	8
Carrying a briefcase	8	8	8	8	8	8	8	8	8	8	8	8
Walking with knee up	**6**	**6**	**6**	8	8	8	**6**	**6**	**6**	**6**	**6**	**6**
Limping man	8	8	8	8	8	8	**7**	8	**7**	8	8	8
Sleepwalking	**5**	**5**	**5**	**6**	**6**	**6**	**2**	**2**	**2**	8	**6**	8
Occluded legs	8	8	8	8	8	8	8	8	8	8	8	8
Normal walk	8	8	8	8	8	8	8	8	8	8	8	8
Diagonal walk	8	8	8	8	8	8	8	8	8	8	8	8
Walking in a skirt	8	8	8	8	8	8	8	8	8	8	8	8
Walking with a dog	**5**	**5**	**5**	**5**	**5**	**5**	**7**	**5**	**5**	**5**	**5**	**5**

Methods	LLE			Isomap			DM			SSDM		
Cases	1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd
View 0°	8	8	8	8	8	8	8	8	8	8	8	8
View 5°	8	8	8	8	8	8	8	8	8	8	8	8
View 10°	8	8	8	8	8	8	8	8	8	8	8	8
View 15°	8	8	8	8	8	8	8	8	8	8	8	8
View 20°	8	8	8	8	8	8	8	8	8	8	8	8
View 25°	8	8	8	8	8	8	8	8	8	8	8	8
View 30°	**6**	8	8	8	8	8	8	8	8	8	8	8
View 35°	**4**	**4**	**4**	**2**	**2**	**4**	**4**	**4**	**4**	**2**	**2**	**2**
View 40°	**4**	**4**	**4**	**4**	**2**	**4**	**4**	**4**	**4**	**5**	**5**	**5**
View 45°	**4**	**4**	**4**	**4**	**4**	**2**	**4**	**4**	**4**	**2**	**2**	**2**
Swinging bag	**6**	**6**	8	**1**	**1**	**5**	8	**6**	**6**	8	8	**2**
Carrying a briefcase	8	8	8	8	8	8	8	8	8	8	8	8
Walking with knee up	**6**	**6**	**6**	8	8	8	**6**	8	**6**	8	**2**	8
Limping man	8	8	8	8	8	8	8	8	8	8	8	8
Sleepwalking	8	8	8	**2**	8	**2**	**2**	**2**	**2**	**2**	**2**	8
Occluded legs	8	8	8	8	8	8	8	8	8	8	8	8
Normal walk	8	8	8	8	8	8	8	8	8	8	8	8
Diagonal walk	8	8	8	8	8	8	8	8	8	8	8	8
Walking in a skirt	8	8	8	8	8	8	8	8	8	8	8	8
Walking with a dog	8	8	8	**5**	**5**	8	**5**	**5**	**5**	**5**	8	8

We use ten-fold cross validation for robustness evaluation. The action sequences of eight subjects used in the previous section are labeled. Action videos of eight subjects, including Daria, Denis, Eli, Ido, Ira, Lena, Lyova and Moshe, are used as the training set. For algorithm LDA, all action sequences of eight subjects are labeled. For SSDM, we randomly select two subjects' action sequences as labeled samples. The results are shown in Table 2. We show the three nearest training actions from the test action. The similarities are computed using the norm of co-occurrence matrices. The number in the grid denotes which class the robustness test action is classified. "1": bent; "2": jack; "3": jump; "4": pjump; "5": run; "6": side; "7": skip; "8": walk; "9": wave1; "+": wave2. These robustness test actions belong to action "walking", so "8" demonstrates that the test action has a right classification. Boldface denotes mis-classification. Table 2 shows that most algorithms cannot handle the changing viewpoints from 35° to 45° in the video data. This is because the silhouette in each frame is completely deformed with a large viewpoint change. In addition, we can conclude that nonlinear algorithms are more robust than linear algorithms. The actions "sleepwalking", "walking with a dog" and "change view" are mis-classified by all linear algorithms. Furthermore, SSDM has an advantage over other methods. Only "sleepwalking" is misclassified

For Peer Review Only

Figure 6. Some samples of image and silhouette in the IXMAS data set. There are all eleven actions performed by two subjects (each row is corresponding to a subject). These images are captured by five cameras with different viewpoints.

Table 3. Comparison with the State-of-the-art Methods

Methods	cam0	cam1	cam2	cam3	cam4
Our proposed method	70.79%	65.71%	73.33%	78.41%	53.97%
Weinland et al. [9] 3D	65.4%	70.0%	54.3%	66.0%	33.6%
Weinland et al. [9] 2D	55.2%	63.5%	*	60.0%	*
Junejo et al. [17] Self-similarities	76.4%	77.6%	73.6%	68.8%	66.1%
Liu et al. [20] ST+Spin-Image	73.46%	72.74%	69.62%	70.94%	*

because of the very different poses with hands up. The major reason is that label information has a critical role in learning the more discriminating manifold.

4.2 *IXMAS*

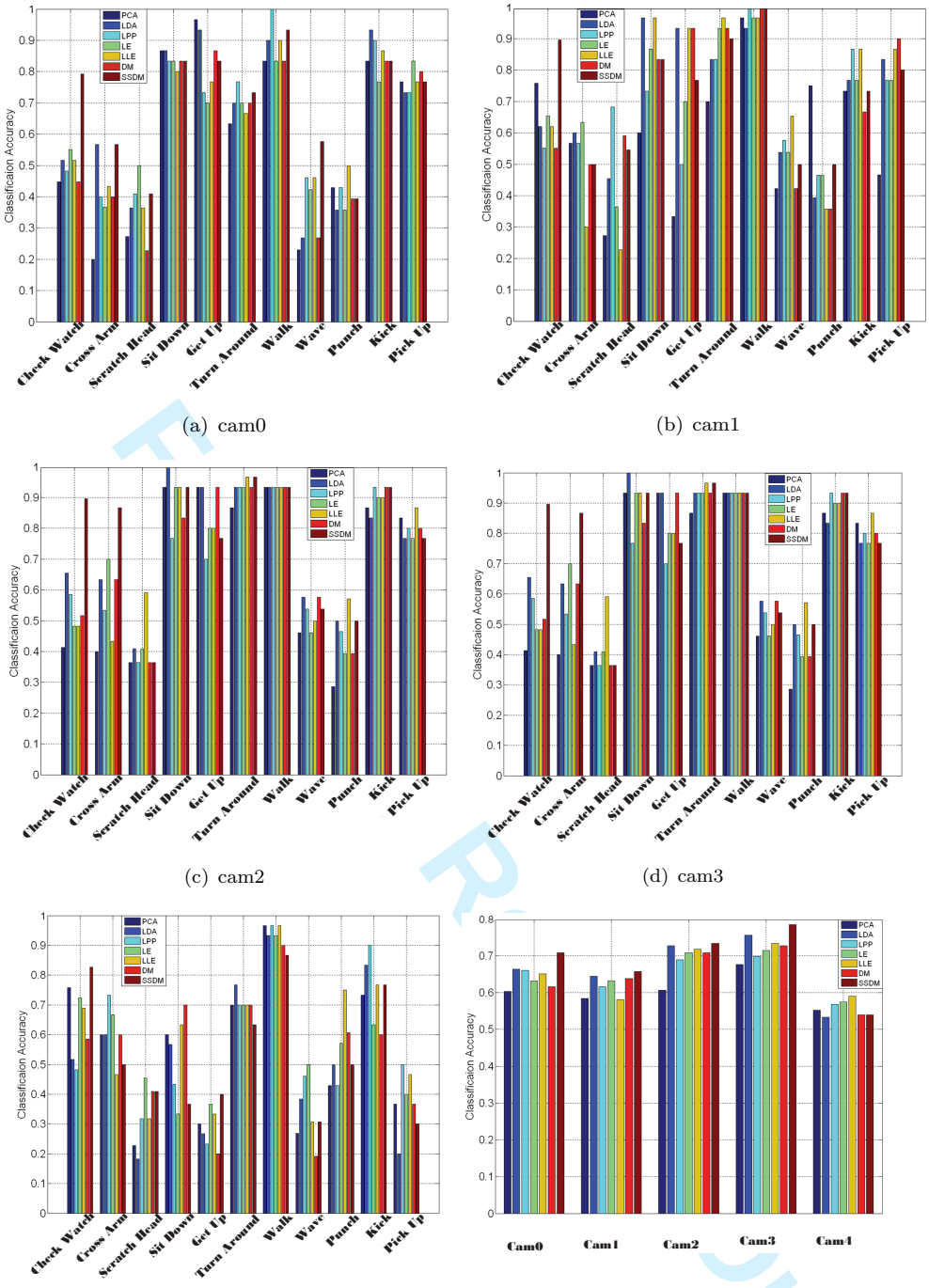
IXMAS [29] is a multiview dataset for view-invariant human action recognition. In this experiment, our recognition model is learned from single views. In essence, we treat the cameras as independent with each other. There are thirteen daily-life motions each performed 3 times by 11 actors. These actors in Ithe XMAS dataset choose freely positions and orientations by themselves. The actions include checking watch, crossing arms, scratching head, sitting down, getting up, turning around, walking, waving, punching, kicking, pointing, picking up and throwing. We choose 11 actions, performed by 10 actors, each 3 times, and viewed by 5 calibrated cameras, the same as in [9]. Some samples are illustrated in Fig. 6. We use the labels offered in [29], but some of them are not very accurate. As shown in Fig. 8, some poses of the scratching head action had been labeled as sitting down. This may decrease the classification accuracy to some degree. We use the leave-one-out scheme to test our proposed method. Each time, we randomly select one subject's actions as test samples, use the others to train our learning model.

4.2.1 The Accuracy of Recognition on IXMAS

In this subsection, we present results on the IXMAS data set using the co-occurrence matrices descriptor and various embedding methods. Also, we will compare our proposed method to the state-of-the-art action recognition methods used on this data-set.

In this experiment, we set the neighborhood of graph $K = 45$, the number of clusters of learned eigen-space $N_0 = 100$. For LDA, we treat all training videos as labeled samples. However, for SSDM, videos of one subject in the training set are randomly selected as labeled samples. All samples including testing samples will be used to learn eigen-space for unsupervised and semi-supervised manifold learning, but LDA exclude test videos because it is a supervised method. Fig. 7 shows the classification accuracy for each camera using different embedding methods and the co-occurrence matrices descriptor. From this figure, we can draw four conclusions

For Peer Review Only



(a) cam0 (b) cam1

(c) cam2 (d) cam3

(e) cam4 (f) Average Accuracy

Figure 7. Classification accuracy for each camera using different embedding methods and the co-occurrence matrices descriptor.

in the following. Firstly, action videos captured by the fourth camera (cam3) are best classified, and the fifth (cam4) produces the worst result. This is because the silhouettes from the top view camera are less distinguishable. Secondly, the action walking is always well classified regardless of view point and graph embedding methods. The actions turning around, kicking, sitting down and picking up are in the second place. The actions checking watch, crossing arms, scratching head, waving and punching are always confused with each other. Similar conclusion were

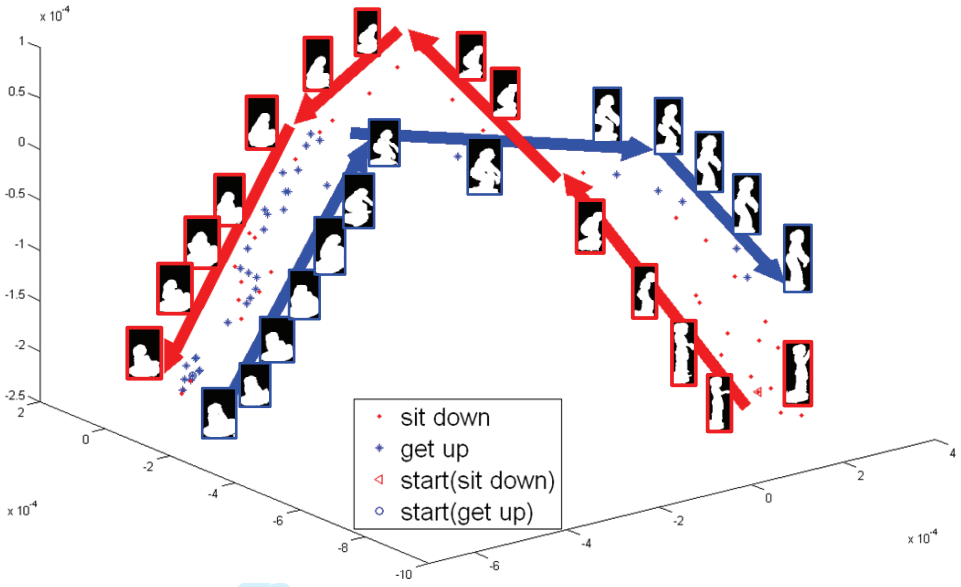


Figure 8. Three dimensionality visualization of actions "sit down" and "get up" both performed by amel. The eigenspace is learned by DM.

found in [9]. Thirdly, SSDM has higher stability of action recognition over view changes than other embedding methods, especially for the action checking watch. Finally, the label samples play an important role on boosting the classification accuracy for this complex data-set. We can see that LDA can also achieve a reasonable accuracy. Moreover, semi-supervised diffusion maps (SSDM) achieves the best classification accuracy.

As a result, using SSDM and the co-occurrence matrices descriptor, the average classification accuracy per camera are illustrate in Table 3. In [9], when learning in 3D and recognizing in 2D, the highest average accuracy only reached 70.0% for the second camera (cam1). Junejo et al. [17] explored self-similarities to recognize actions. However, their method relied on geometric properties and needed to combine them with machine learning for cross-view action recognition. In a single view setting, our method can achieve 78.41% accuracy for the fourth camera (cam3). It is difficult to directly compare our results with [29] (93.33%) because of different experimental settings and their methods used a full 3D model reconstructed from multiple camera views.

4.2.2 Comparison Between the Bag-of-words and the Co-occurrence Matrices Descriptor

The actions, "sit down" and "get up" in the IXMAS data set, are performed successively. Most silhouettes of the two actions are very similar, only with a different order. If we use the bag of words descriptor to describe these two actions, they will be confused because of its non-sensitivity to temporal changes. However, the co-occurrence matrices descriptor captures the temporal information so that it can avoid such confusion. Fig. 8 shows a three dimension visualization of actions "sit down" and "get up" both performed by Amel. We can see that the two sequences of three dimension pointS in learned eigen-space are very close, but with a reverse order.

In Fig. 9, the two images correspond to similarity comparisons between actions "sit down" and "get up". We calculate the similarities between all 30 "sit down" videos performed by 10 subjects with 3 times and all 30 "get up" videos also performed by such 10 subjects with 3 times. From Fig. 9, we can draw two conclusions.

For Peer Review Only

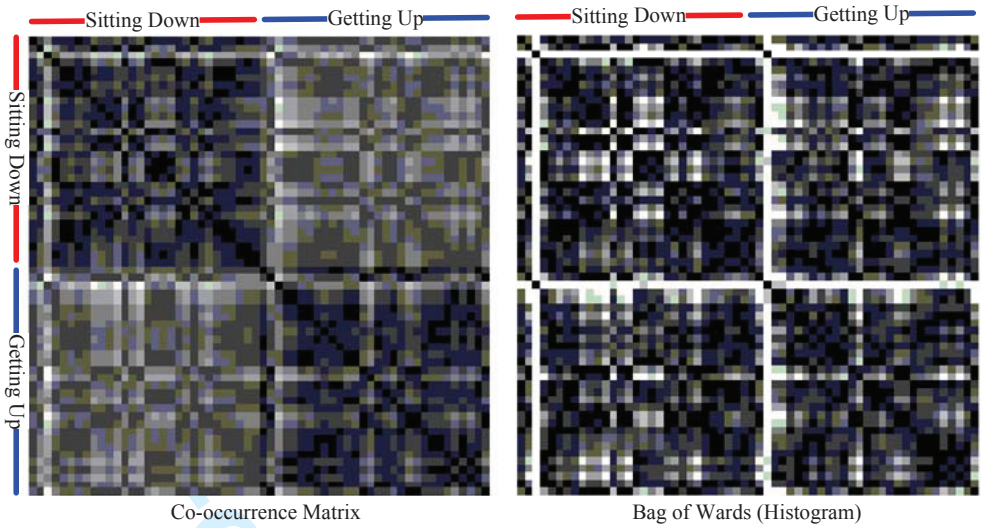
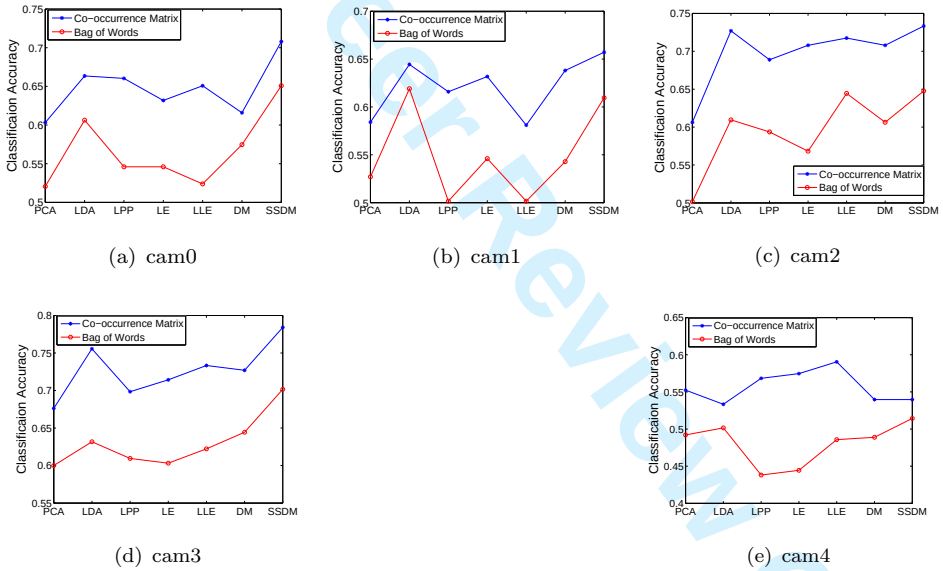


Figure 9. Similarity comparison between the bag of words and the co-occurrence matrices descriptor. Each image contain 60×60 grids which one is corresponding to similarity of two action sequences. The darker grid denotes the two actions are more similar. The eigenspace is learned by DM.



(a) cam0 (b) cam1 (c) cam2

(d) cam3 (e) cam4

Figure 10. Classification accuracy comparison between the co-occurrence matrices descriptor and the bag of words (histogram) for each camera.

Firstly, the two images have the similar texture. Secondly, it's obvious that action "sit down" sequences are more similar with "sit down" sequences than "get up" sequences using the co-occurrence matrices descriptor. However, when using the bag of words descriptor, actions "sit down" and "get up" have very similar behaviors. That is to say, the co-occurrence matrices descriptor is more discriminative than the bag of words descriptor.

Fig. 10 depicts the classification accuracy on five cameras. Compared with the bag of words descriptor, the classification accuracy increases a lot using the co-occurrence matrices descriptor, no matter which camera and which embedding method is used.

For Peer Review Only

5. Conclusion

In this paper, we propose a new representation of actions on the intrinsic shape manifold learned by various graph embedding algorithms. The co-occurrence matrices descriptor captures more temporal information than the bag of words (histogram based) descriptor which only considers the spatial information. In addition, we compare the performance of the co-occurrence matrices descriptor on different manifolds learned by various graph embedding methods. The results show that nonlinear algorithms are more robust than linear algorithms. Furthermore, we conclude that the label information is useful in learning more discriminating manifolds. The future work will be investigated in the following directions:

- We can use more challenging datasets for testing the performance of the co-occurrence matrices descriptor and graph embedding methods.
- For multi-view cameras, we can design a combination framework of all views in the recognition procedure, such as voting.
- In this paper, KNN is used for classification. More advantaged techniques, such as SVM and Adaboost, can be explored for action recognition.
- Since the nonlinear manifold learning methods including LE, LLE, ISOMAP, DM and SSDM all don't offer an explicit mapping function, it will take more time to compute the low dimensional representation of on-line samples. So the future work can address the problem of the out-of-sample issue for embedding unseen samples.

Acknowledgment

The work described in this article was supported partially by the grants from the National Natural Science Foundation of China (NSFC, grant no. 61002040, 60903115) and partially by NSFC-GuangDong (grant no. 10171782619-2000007).

References

[1] M. Belkin and P. Niyogi, *Laplacian eigenmaps for dimensionality reduction and data representation*, Neural Computation 15 (2003), pp. 1373–1396.
[2] W. Bian and D. Tao, *Max-min distance analysis by using sequential sdp relaxation for dimension reduction*, IEEE Transactions on Pattern Analysis and Machine Vision and Applications VOL. 33.
[3] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, *Actions as space-time shapes*, in *Tenth IEEE International Conference on Computer Vision*, vol. 2, oct., 2005, pp. 1395 –1402.
[4] A. Bobick and J. Davis, *The recognition of human movement using temporal templates*, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (2001), pp. 257 –267.
[5] I. Borg and P.J. Groenen, *Modern Multidimensional Scaling: Theory and Applications*, vol. 14, 2nd ed., Springer, New York, NY. (2005).
[6] T.J. Chin, L. Wang, K. Schindler, and D. Suter, *Extrapolating Learned Manifolds for Human Activity Recognition*, in *IEEE International Conference on Image Processing*, vol. 1, Oct., 2007, pp. I–381 – I–384.
[7] R. Coifman, S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, and S. Zucker, *Geomatic diffusions as a tool for harmonics amalysis and structure definition of data*, Proc. Nat'l academy of sciences 102 (2005), pp. 7426–7431.
[8] R.D. Cook, *Fisher lecture: Dimension reduction in regression*, Statistical Science 22 (2007), pp. 1 – 26.
[9] W. Daniel, R. Remi, and B. Edmond, *Action recognition from arbitrary views using 3d exemplars*, ICCV (2007).
[10] A. Elgammal and C. su Lee, *Inferring 3d body pose from silhouettes using activity manifold learning*, In CVPR (2004), pp. 681–688.
[11] X. Gao, X. Wang, D. Tao, and X. Li, *Supervised gaussian process latent variable model for dimensionality reduction*, IEEE Transactions on Systems, Man and Cybernetics, Part: B: CYBERNETICS (2010).
[12] N. Guan, D. Tao, Z. Luo, and B. Yuan, *Manifold regularized discriminative non-negative matrix factorization with fast gradient descent*, IEEE Transactions on Image Processing (2011).

For Peer Review Only

[13] R.M. Haralick, Dinstein, and K. Shanmugam, *Textural features for image classification*, IEEE Transactions on Systems, Man, and Cybernetics SMC-3 (1973), pp. 610–621.
[14] X. He and P. Niyogi, *Locality preserving projections*, Advances in Neural Information Processing Systems 16 (2004).
[15] K. Jia and D.Y. Yeung, *Human action recognition using local spatio-temporal discriminant embedding*, IEEE Conference on Computer Vision and Pattern Recognition (2008), pp. 1–8.
[16] I.T. Jolliffe, *Principal Component Analysis*, 2nd ed., New York: Springer-Verlag (2003).
[17] I.N. Junejo, E. Dexter, I. Laptev, and P. Perez, *Cross-view action recognition from temporal self-similarities*, ECCV 2 (2008), pp. 293–306.
[18] V. Kellokumpu, G. Zhao, and M. Pietikinen, *Recognition of human actions using texture descriptors*, Machine Vision and Applications (2009), pp. 1–14.
[19] F. Liu and Y. Jia, *Human action recognition using manifold learning and hidden conditional random fields*, The 9th International Conference for Young Computer Scientists (2008), pp. 693 –698.
[20] J. Liu, S. Ali, and M. Shah, *Recognizing human actions using multiple features*, CVPR Anchorage, AK (2008), pp. 1–8.
[21] S.T. Roweis and L.K. Saul, *Nonlinear dimensionality reduction by locally linear embedding*, Science 2 (2000), pp. 2323–2326.
[22] L. Shao and X. Chen, *Histogram of body poses and spectral regression discriminant analysis for human action categorization*, In Proceedings of the British Machine Vision Conference (BMVC) Aberystwyth, UK (2010).
[23] R. Souvenir and K. Parrigan, *Viewpoint manifolds for action recognition*, Image Video Process 1 (2009).
[24] S. Tabbone, L. Wendling, and J.P. Salmon, *A new shape descriptor defined on the radon transform*, Computer Vision and Image Understanding archive 102 (2006), pp. 42–51.
[25] D. Tao, X. Li, X. Wu, and S.J. Maybank, *Geometric mean for subspace selection*, IEEE Transactions on Pattern Analysis and Machine Vision and Applications VOL. 31, pp. 260–270.
[26] D.C. Tao, X. Li, X. Wu, and S.J. Maybank, *General tensor discriminant analysis and gabor features for gait recognition*, IEEE Transactions on Pattern Analysis and Machine Vision and Applications VOL. 29 (OCTOBER 2007), pp. 1700–1715.
[27] J.B. Tenenbaum, V. de Silva, and J.C. Langford, *A global geometric framework for nonlinear dimensionality reduction*, Science 2 (2000), pp. 2319–2323.
[28] L. Wang and D. Suter, *Recognizing human activities from silhouettes: Motion subspace and factorial discriminative graphical model*, IEEE CVPR 2 (2007.), pp. 1 –8.
[29] D. Weinland, R. Ronfard, and E. Boyer, *Free viewpoint action recognition using motion history volumes*, Computer Vision and Image Understanding 104 (2006), pp. 249–257.
[30] 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 Transactions on Pattern Analysis and Machine Intelligence 29 (2007), pp. 40–51.
[31] A. Yilmaz and M. Shah, *Actions sketch: a novel action representation*, in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 984 – 989.
[32] T. Zhang, D. Tao, X. Li, and J. Yang, *Patch alignment for dimensionality reduction*, IEEE Transactions on Knowledge and Data Engineering VOL. 21 (SEPTEMBER, 2009).
[33] F. Zheng, L. Shao, and Z. Song, *Eigen-space learning using semi-supervised diffusion maps for human action recognition*, ACM International Conference on Image and Video Retrieval (2010), pp. 151–157.
[34] F. Zheng, L. Shao, and ZhanSong, *A set of co-occurrence matrices on the intrinsic manifold of human silhouettes for action recognition*, CIVR (2010), pp. 454–461.
[35] F. Zheng and Z. Song, *Diffusion maps for dimensionality reduction with partially labeled samples*, The 2nd International Conference on Computer and Automation Engineering 2 (2010), pp. 68–73.
[36] T. Zhou, D. Tao, and X. Wu, *Manifold elastic net: a unified framework for sparse dimension reduction*, Data Mining and Knowledge Discovery (2010).

For Peer Review Only