Pattern Recognition 53 (2016) 148–162

Contents lists available at ScienceDirect

Pattern Recognition

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

Human action recognition with graph-based multiple-instance learning

Yang Yi [a,b,*], Maoqing Lin [a]

[a] School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China
[b] Xinhua College of Sun Yat-sen University, Guangzhou 510520, China

ARTICLE INFO

Article history:
Received 26 April 2015
Received in revised form
31 August 2015
Accepted 28 November 2015
Available online 3 December 2015

Keywords:
Action recognition
Dense trajectory
Motion compensation
Spectral embedding
Multiple-instance learning

ABSTRACT

A new approach to human action recognition from realistic videos is presented in this paper. First, an affine motion model is utilized to compensate background motion for the purpose of extracting dense foreground trajectories. Then, a trajectory spectral embedding is introduced to split up foreground action into multiple spatio-temporal action parts for constructing a mid-level representation. To deal with over-segmentation, a novel density discontinuity detector is proposed for the sake of generating semantically salient action parts. Finally, to handle the ambiguity in the training set, action classification is formulated within the multiple-instance learning framework, which a spatio-temporal graph model is incorporated into. Extensive experiments show that the proposed approach achieves competitive results to state of the art on UCF Sports, Kisses/Slaps, YouTube, and Hollywood datasets.

1. Introduction

Human action recognition from videos is considered as an important topic in the field of computer vision due to the large number of potential applications in the areas of automatic visual surveillance, advanced human–computer interaction and content-based video retrieval, etc. Recent interest in human action recognition research has shifted from the simple action recognition in a well-controlled scene, to the more realistic action identification under an unconstrained environment which is also referred to as "in the wild" (e.g., videos recorded by an amateur using a hand-held camera, feature films [1,2], sports broadcast [3,4] and home videos on YouTube [5]). The action recognition from a complex environment is challenging owing to camera motion, multi-object occlusion, background clutter, motion blur, low resolution and dramatic changes in illumination, object scale, and viewpoint, etc. The main difficulty lies in how to extract reliable, informative and discriminative features and how to create a video representation model to narrow the semantic gap between low-level features and high-level actions.

1.1. Background and motivation

The earlier research on the feature extraction of action recognition is primarily based upon the methods of human body geometry or silhouette [6], which however, is only applicable to the simple action recognition under monitoring scenarios, such as the Weizmann dataset [6]. In more complex scenes, it is often difficult to extract reliable information of the object shape and contour because of lacking prior information of the appearance, scale and recording viewpoint of the actor. Particularly, in the case of significant camera motion, segmenting moving objects from videos has proven to be a hard problem in computer vision [7,8].

The representation based on local spatio-temporal descriptor is the most popular in the current research of human action recognition. It is usually combined with the bag-of-features (BoF) model to generate a single histogram descriptor per action clip by counting the visual words in the neighborhood of spatio-temporal interest points or trajectories. However, most of the local descriptors are purely computed from small and fragmentized cuboids that only provide inadequate motion information and result in limited discriminative power, which are insufficient to represent more complex movements. Furthermore, fixing the size of cuboids is not adaptive to describe the dynamic properties of the actions whose spatio-temporal locations are unknown.

To overcome the limitations of the foregoing conventional action representations, some recent attempts [9,10] have been made to apply the state-of-the-art object detection techniques [11]

* Corresponding author at: School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China. Tel.: +86 13902295111.
E-mail address: issyy@mail.sysu.edu.cn (Y. Yi).

http://dx.doi.org/10.1016/j.patcog.2015.11.022
0031-3203/

to the development of mid-level video representation, which aim to detect motion-related semantic entities in videos (e.g., the person and the interacting objects) for modeling the complex actions from a high semantic level. These approaches are designed to learn the internal structure of action video in terms of its constituent parts, which are informative and discriminative. Nevertheless, their effectiveness is highly dependent on the success of the existing detection algorithms for object and action, which limits their application in recognizing actions from videos "in the wild". More importantly, the detector generally operates in every single frame, resulting in a lack of spatio-temporal stability and consistency [12].

Therefore, more recent approaches are based on discriminative spatio-temporal "patches" or "parts" rather than semantic entities. For example, Raptis et al. [13] extracted trajectory groups and developed a latent model over a fixed number of activity parts. However, their approach requires bounding box annotations and uses only a fixed subset of clusters for all the videos, ignoring the fact that each video has its own decomposition structure. Furthermore, it relies on the latent parts and thus needs to solve a complex inference problem for each test video. Liu et al. [14] proposed the attribute-based action representation useful for recognizing novel action classes even without training examples via human-specified attributes, but its performance relies on the quantity of attributes that training data and testing data share. Jain et al. [15] learned discriminative spatio-temporal patches with exemplar-based clustering method, and Wang et al. [16] obtained motionlets, namely mid-level and spatio-temporal parts by extracting and clustering the 3D regions with high motion saliency. Peng et al. [17] presented a method for action recognition with multi-layer nested fisher vector encoding to improve the recognition performance by preserving global structure and rich information, and Sapienza et al. [18] decomposed a video volume into multiple subvolumes to learn discriminative space–time actions cast as bags of features in a multiple-instance learning (MIL) framework. Zhu et al. [19] also developed a new max-margin multi-channel multiple-instance learning to exploit discriminative video features named by actons.

1.2. Overview of the proposed approach

Inspired by the ideas of the above-mentioned mid-level feature representations, especially the work of [18], we propose a novel mid-level video representation, namely foreground motion segmentation based on density discontinuity detection in trajectory spectral embedding. Different from [18], this paper utilizes an affine model for background motion compensation before trajectory extraction, and then employs the density discontinuity detector to handle cluster over-segmentation. With regard to the multiple instance learning, we incorporate a spatio-temporal graph kernel to emphasize the mid-level semanteme since space–time relationship exists between action parts. Specifically, the trajectory spectral clustering has been successfully applied to the construction of mid-level video representation [20,21], object segmentation [22] and tracking [23], which however, probably results in over-segmentation. On this account, an embedding density discontinuity detector is proposed. Different from the traditional idea of spectral embedding [21], the proposed method can perfectly handle the problem of over-segmentation caused by the articulated body motion under complex environments. It is shown that the discontinuity of the embedding density can serve as the strong indicator to locate the boundaries of action parts and when the density discontinuity is incorporated into the embedding discretization process, over-segmentation error can be well controlled. Compared with the previous approaches [13], the resulting trajectory clusters are more semantically meaningful, directly corresponding to the mid-level parts of the foreground action. Note that no explicit object detector is used to discover the action parts. Instead, all the moving regions that have sufficient temporal and spatial coherence are regarded as the candidate action parts in a completely unsupervised manner. Therefore, the tedious process of manually labeling the training data is avoided.

Video action recognition is inherently a weakly supervised learning task. The class label of each video sample is given, but the exact spatio-temporal location of the specified action in the video sequence is still unknown. Furthermore, due to the various factors such as irregular camera motion, cluttered background, self-occlusion, and non-rigid body deformation, the automatic localization of the foreground object and the automatic extraction of the action part are usually far from perfect, yielding the spurious action parts which are irrelevant to the action of interest. In this case, the training set ineluctably involves ambiguity. To tackle the irrelevant action parts, the action classification is formulated within a MIL framework, where each action video is cast as a "bag" of action parts and each part is treated as an instance. Different from the traditional supervised learning systems, the training label is only associated with the instance bag rather than every single instance, which tolerates the existence of irrelevant instances in the training set and false detections in the input video sequences. The MIL paradigm has been favorably applied in the field of motion detection [24], action recognition [25], video object tracking [26] and image classification [27], etc.

Furthermore, it is known that different action parts in a video sequence are often highly correlated, rather than independently distributed, e.g., there exist the interaction between human bodies and action-related objects, and the spatial relationship of each body parts. If the rich contextual information hidden within these relationships is fully utilized, the recognition performance can be potentially improved. For this purpose, a spatio-temporal graph (STG) model is introduced in this paper to capture the spatio-temporal interaction relationship among spatially adjacent action parts and then incorporated into the MIL framework (STG-MIL). Unlike the most existing MIL algorithms [28,29], STG-MIL treats multiple instances in the bags in a non-independent and identically distributed way that exploits the latent contextual information among multiple instances, which therefore is more suitable for dealing with the multi-instance video classification.

Overall, compared with the existing approaches of human action recognition, the contributions of the study are threefold. Firstly, a motion compensation approach based on affine model is utilized to estimate the foreground flow field and suppress the background flow field. The compensated optical flow field is used to extract dense foreground trajectories and compute feature descriptors. Secondly, a new trajectory clustering algorithm based on trajectory spectral embedding and density discontinuity detection is proposed, in which an action is decomposed into a collection of semantically salient spatio-temporal (i.e. two spatial dimensions and one temporal dimension, 2D+t) action parts, in order to construct the mid-level feature representation for action videos. Thirdly, a graph-based MIL approach is presented to handle the ambiguity involved in the training set. Each mid-level action part is cast as an instance in the MIL bag and a spatio-temporal graph model is incorporated in order to leverage the interaction relationship among the instances. Correspondingly, a novel designed spatio-temporal graph kernel is introduced for the action classification. The overview of the proposed approach is shown in Fig. 1.

The rest of this paper is organized as follows. Section 2 describes the middle level representation of action videos. Section 3 gives a detailed description of the spatio-temporal graph-based multiple-instance learning. The experimental results and analysis on the

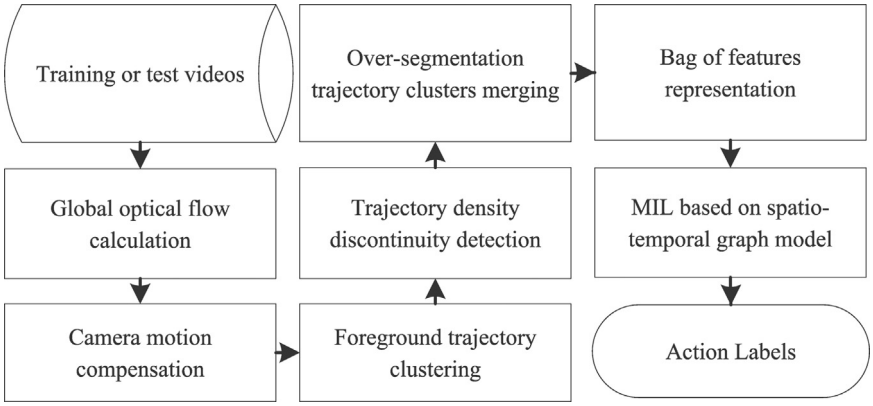

Fig. 1. An overview of the proposed approach.

performance of the proposed approach are shown in Section 4, followed by the conclusions in Section 5.

2. Middle level representation of action videos

The key idea of the mid-level representation proposed in this research is to represent human actions by a set of spatio-temporal (2D+t) action parts and to capture the inner-structure information among these action parts. Here, each action part can be regarded as a 2D+t tube in the 3-dimensional video volume, which may correspond to the primitive human sub-action (e.g., the swing movement in the activity of playing tennis), a motion-related semantic entity (e.g., the tennis racket), or a random but informative spatio-temporal patch in the video.

2.1. Motion compensation for foreground trajectory extraction

The camera motion is inevitably involved when we process the unconstrained videos. When there exists camera motion and the background is dynamic, the global background flow and the local foreground flow are mixed up in the video frames. Therefore, the action recognition in such scenarios requires preprocessing to eliminate camera motion and recover the independent motion merely resulting from the actors. The dominant motion compensation is used for this purpose. Compared with other polynomial motion models, such as the 4-parameter model and the 8-parameter quadratic model, the 6-parameter 2D affine model [30] describing the global (i.e. dominant) motion between consecutive frames is a better trade-off between accuracy and efficiency, which is rather necessary for dealing with large-scale video datasets. Considering that there exists large distance between the camera and the moving body in the experimental datasets, especially in outdoor scenes, we utilize this model to estimate the background dominant motion.

Let $\mathbf{p}_t = (x_t, y_t)$ denote the point coordinate on the t-th frame and define the affine optical flow of $\boldsymbol{p}_t$ as:

$$\boldsymbol{f}_A(\boldsymbol{p}_t) = \begin{bmatrix} u_A(\boldsymbol{p}_t) \\ v_A(\boldsymbol{p}_t) \end{bmatrix} = \begin{pmatrix} c_1(t) \\ c_2(t) \end{pmatrix} + \begin{pmatrix} a_1(t) & a_2(t) \\ a3(t) & a4(t) \end{pmatrix} \begin{pmatrix} x_t \\ y_t \end{pmatrix} \tag{1}$$

where $u_A(\boldsymbol{p}_t) = c_1(t) + a_1(t)x_t + a_2(t)y_t$ and $v_A(\boldsymbol{p}_t) = c_2(t) + a_3(t)x_t + a_4(t)y_t$ represent the horizontal component and the vertical component of $\boldsymbol{f}_A(\boldsymbol{p}_t)$, respectively. $\boldsymbol{\theta} = [c_1, c_2, a_1, a_2, a_3, a_4]^T$ is the parameter vector of the 6-parameter motion model, where c_1 and c_2 are associated with the camera translation whilst a_1, a_2, a_3 and a_4 are related to camera rotation and scaling.

For the overall optical flow computation, the TV $-L^1$ variational optical flow algorithm proposed by Chambolle et al. [31] is introduced to obtain the accurate and high-resolution optical flow field with higher computational efficiency. Denote $\boldsymbol{f}(\boldsymbol{p}_t)$ as the whole optical flow of $\boldsymbol{p}_t$. Then the rest optical flow after removing the affine optical flow, i.e. the compensated optical flow is defined as:

$$\boldsymbol{f}_F(\boldsymbol{p}_t) = \boldsymbol{f}(\boldsymbol{p}_t) - \boldsymbol{f}_A(\boldsymbol{p}_t) \tag{2}$$

Here dominant motion is considered as $\boldsymbol{f}_A(\boldsymbol{p}_t)$, thus Eq. (2) aims at eliminating (i.e. compensating) the camera motion. Notice that not all video sequences are in the same condition, e.g., given the close-up shot of the athletes in sports videos, the dominant motion is the affine estimation of the athletes' movement. Under the situation like this, the result of motion compensation is tough to explain. Nevertheless, $\boldsymbol{f}_F(\boldsymbol{p}_t)$ can still perform different patterns as foreground motion or background motion. Fig. 2 shows the contrast between the original overall flow field and the foreground flow field after compensation. Apparently, the affine model could well describe the dominant motion resulting from dynamic cameras by effectively suppressing the noise from background motion and strengthening the foreground optical flow field.

The dense trajectories are then extracted by tracking densely using this compensated optical flow fields [32]. The resulting trajectories naturally correspond to the foreground object, as shown in Fig. 3. After motion compensation, the dense trajectories mostly focus on the foreground area and thus are much more meaningful and descriptive, compared with the trajectories before motion compensation.

2.2. Trajectory spectral embedding

The foreground area is over-segmented into spatially compact spatio-temporal blocks by trajectory spectral clustering. Let $[f_i, F_i]$ denote the trajectory interval and $\boldsymbol{p}_t^i$ denote the point coordinate on the t-th frame in the i-th trajectory. The affinity between trajectory $\boldsymbol{T}_i$: $\{\boldsymbol{p}_t^i,\ t = f_i \cdots F_i\}$ and trajectory $\boldsymbol{T}_j$: $\{\boldsymbol{p}_t^j,\ t = f_j \cdots F_j\}$ is defined as $\boldsymbol{W}_{ij}$:

$$\boldsymbol{W}_{ij} = \exp\left(-d(\boldsymbol{T}_i, \boldsymbol{T}_j)\right)$$
$$d(\boldsymbol{T}_i, \boldsymbol{T}_j) = \max_{t \in o(\mathbf{T}_i, \mathbf{T}_j)} d_{\mathrm{sp}}(t) * \frac{1}{o(\boldsymbol{T}_i, \boldsymbol{T}_j)} \sum_{t \in o(\boldsymbol{T}_i, \boldsymbol{T}_j)} d_{\mathrm{vel}}(t) \tag{3}$$

where d is the distance measure function between trajectory T_i and $\boldsymbol{T}_j$, $d_{\mathrm{sp}}(t) = \|\boldsymbol{p}_t^i - \boldsymbol{p}_t^j\|_2$ and $d_{\mathrm{vel}}(t) = \|(\boldsymbol{p}_t^i - \boldsymbol{p}_{t-1}^i) - (\boldsymbol{p}_t^j - \boldsymbol{p}_{t-1}^j)\|_2$ denote the spatial Euclidean distance and the velocity distance of $\boldsymbol{T}_i$ and $\boldsymbol{T}_j$, respectively, at a particular instant t during their temporal overlap $o(\boldsymbol{T}_i, \boldsymbol{T}_j) = [f_i, F_i] \cap [f_j, F_j]$. Specially, the affinity of trajectory pairs which are not temporally overlapped is enforced to be zero. Compared with [22], this distance metric penalizes the

Fig. 2. Comparison of the optical flow field before and after motion compensation. From up to bottom: the original video frames, the original whole optical flow field, and the foreground flow field after motion compensation.

Fig. 3. Comparison of the dense trajectories before and after motion compensation. From left to right: the consecutive frames, the dense trajectories, and the foreground trajectories after motion compensation. The yellow line represents trajectory whilst red point represents the current position of the trajectory.

trajectories which are spatially far apart so as to ensure spatial compactness of the subsequent trajectory clustering.

With the above-mentioned weight matrix $\boldsymbol{W}$ at hand, the original trajectory set $\{\boldsymbol{T}_i\}_{i=1}^{N}$ can be mapped into a K-dimensional Eigenspace $\mathbf{R}^K$ by an eigen decomposition of the normalized graph Laplacian, where N is the number of the dense trajectories extracted after motion compensation and K is the number of clusters. Let $\boldsymbol{D}$ denote the diagonal matrix $\boldsymbol{D}ii = \sum_j \boldsymbol{W}_{ij}$, which represents the degree matrix of the weight matrix $\boldsymbol{W}$. Then, the trajectory spectral embedding can be obtained by the eigenvectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_K$ corresponding to the first K smallest eigenvalues $\lambda_1, \ldots, \lambda_K$ of the normalized Laplacian $\boldsymbol{D}^{-\frac{1}{2}}(\boldsymbol{D}-\boldsymbol{W})\boldsymbol{D}^{-\frac{1}{2}}$. The every row vector

Fig. 4. Results of the foreground trajectory clustering based on spectral embedding. The first row represents the original video frames, and the following four rows correspond to different values of K as 5, 6, 10, and 12.

of matrix $[\mathbf{v}_1, \ldots, \mathbf{v}_K] \in \mathbf{R}^{N\times K}$ corresponds to the new coordinate of $\boldsymbol{T}_i$ after spectral embedding in space $\mathbf{R}^K$. Eigenvector rotation [33] is adopted to discretize the resulting spectral embedding, and then an initial trajectory over-segmentation is obtained.

As is well known, spectral clustering has become one of the most popular modern clustering algorithms and very often outperforms traditional algorithms such as k-means [34,35], and then the spectral clustering is employed to cluster trajectories in this study. Fig. 4 shows the examples of the foreground motion segmentation results with the different choices of eigenvector number K.

2.3. Embedding density discontinuity detector

It is found that the foregoing trajectory clustering algorithm often tends to break large coherent regions into chunks as the number of eigenvectors K varies. Motion-based trajectory clustering becomes difficult in situations where articulated human motion or non-rigid body motion exist since the articulated body parts may move differently while separate subjects may move in a similar manner, resulting in over-segmentation or under-segmentation. As shown in Fig. 4, the optimum number of clusters K is by no means obvious. K equaling 5 results in under-segmentation, which makes some trajectory clusters span across the boundary of the object (e.g., the human body and the horse are regarded as a whole in the last column). Instead, K equaling 6 leads to over-segmentation (e.g., the person in the background is subdivided into multiple chunks in the first column). Hence, determining the number of action parts K automatically and computing a corresponding clustering of the trajectory embedding turn out to be a non-trivial problem [33].

As a remedy to this problem, an embedding density discontinuity detector is proposed to locate the boundaries of action parts in trajectory spectral embedding, and then to merge trajectory clusters whose inter-cluster boundaries have weak discontinuity. The over-segmentation error is efficiently controlled by this method in contrast to previous trajectory clustering approaches. Specifically, the density discontinuity is defined as the local extremum of trajectory affinity in spectral embedding.

Firstly, the trajectory affinity matrix $\hat{\boldsymbol{W}}$ in the embedding space $\mathbf{R}^K$ is defined as:

$$\hat{\boldsymbol{W}} = V\boldsymbol{\Lambda}\boldsymbol{V}^T \tag{4}$$

where $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_K)$ and $\boldsymbol{V} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_K]$. Notice $\hat{\boldsymbol{W}}$is a $N \times N$ symmetric matrix whose element $\hat{\mathbf{W}}ij$ measures the affinity of $\mathbf{T}_i$ and $\mathbf{T}_j$ in the trajectory embedding. Embedding affinities are visualized in Fig. 5.

Then, consider the neighborhood of trajectories for computing motion embedding densities. Spatially neighboring trajectory points in video frames are regarded as candidates for motion discontinuity detection in the embedded space. The neighborhood relations among trajectories are captured with a Delaunay triangulation graph. Specifically, the Delaunay triangulation graph Del_t is constructed by all the trajectory points of a video frame t. Let e_{ij}^t be the Delaunay edge connecting trajectory $\boldsymbol{T}_i$ and $\boldsymbol{T}_j$ in Del_t. With regard to $\boldsymbol{T}_i$, its spatio-temporal neighborhood $Nei(\boldsymbol{T}_i)$ is defined as the set of its neighboring trajectories:

$$Nei(\boldsymbol{T}_i) = \left\{ \boldsymbol{T}j \mid \exists t, t \in o(\boldsymbol{T}_i, \boldsymbol{T}_j), e_{ij}^t = 1 \right\} \tag{5}$$

where $o(\boldsymbol{T}_i, \boldsymbol{T}_j)$ denotes the overlap interval between $\boldsymbol{T}_i$ and $\boldsymbol{T}_j$ in the time domain. The neighborhood measure method considers the Delaunay triangulation subdivision graph for all the frames in

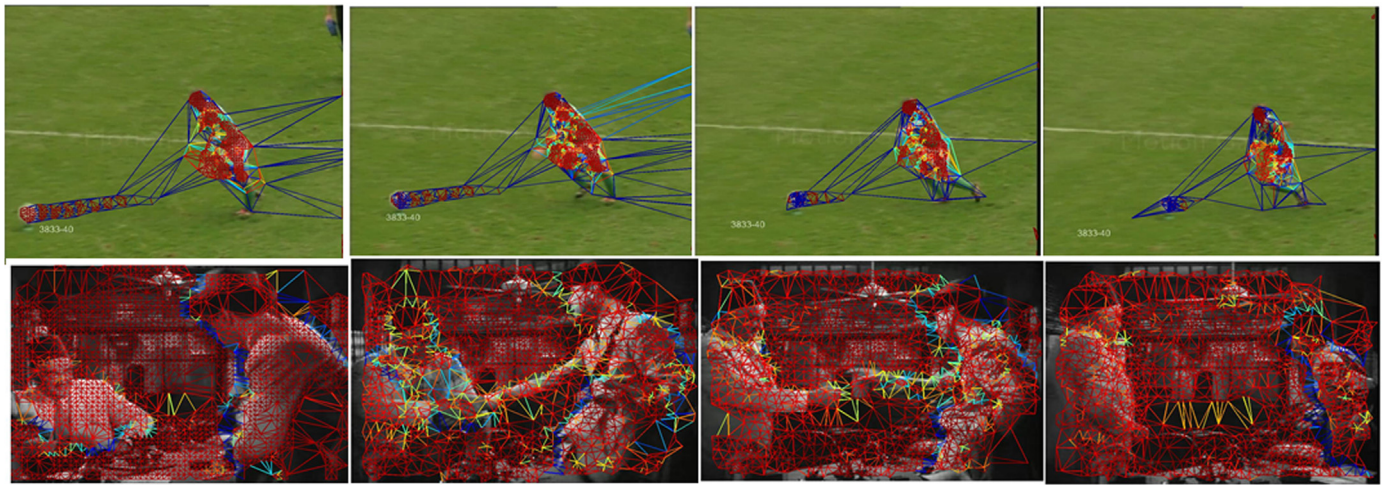

Fig. 5. Visualization of the affinity of embedded trajectories. The color of the edge in Delaunay triangle represents the affinity weight, red representing a high value while blue representing a low value.

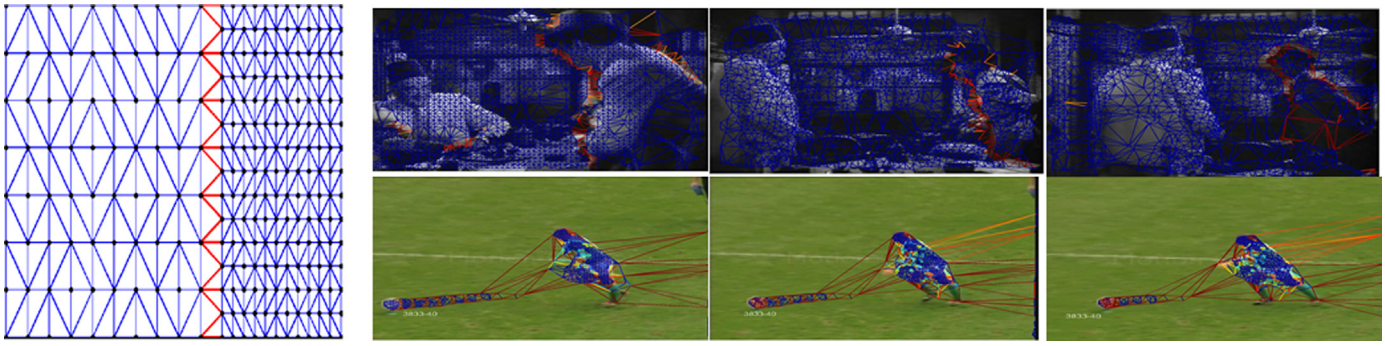

Fig. 6. Examples of embedding density discontinuity. Trajectory spectral embedding has varying density and thus exhibits discontinuity. The proposed discontinuity detector outputs high discontinuity values (shown in red) across object boundaries, and low (shown in blue) at object interiors. Therefore, it can serve as the segmentation evidence of action parts.

video sequences. Thus, the information of the trajectory movement in a larger time scale is utilized sufficiently.

Thirdly, the trajectory density $\rho(\boldsymbol{T}_i)$ is defined as the max embedded affinity between $\boldsymbol{T}_i$and its spatio-temporal neighborhood trajectory set:

$$\rho(\boldsymbol{T}_i) = \max_{\mathbf{T}_j \in Nei(\mathbf{T}_i)} \hat{\boldsymbol{W}}_{ij} \tag{6}$$

Eq. (6) measures the local distribution density of the trajectories in the feature space, based on the K-dimensional affinity between trajectories in the spatio-temporal neighborhood. The more compact the trajectory distribution is, the bigger $\rho(\boldsymbol{T}_i)$ is; otherwise, $\rho(\boldsymbol{T}_i)$ is smaller.

For the adjacent trajectory pair $(\boldsymbol{T}_i, \boldsymbol{T}_j)$ in the Delaunay triangulation graph, its density discontinuity evaluation function is defined as:

$$Dis(\boldsymbol{T}_i, \boldsymbol{T}_j) = \begin{cases} 1 - \hat{\boldsymbol{W}}_{ij} \frac{1}{\max(\rho(\boldsymbol{T}_i), \rho(\boldsymbol{T}_j))}, & \boldsymbol{T}_j \in Nei(\boldsymbol{T}_i) \\ 1, & \text{otherwise} \end{cases} \tag{7}$$

Note that $\forall \boldsymbol{T}_j \in Nei(\boldsymbol{T}_i)$, $\hat{\boldsymbol{W}}_{ij} \leq \max(\rho(\boldsymbol{T}_i), \rho(\boldsymbol{T}_j))$, thus the value range of $Dis(\boldsymbol{T}_i, \boldsymbol{T}_j)$ is $[0, 1]$. When the Delaunay edge which connects $\boldsymbol{T}_i$ and $\boldsymbol{T}_j$ crosses the boundaries of different object parts, $Dis(\boldsymbol{T}_i, \boldsymbol{T}_j)$ gets a higher response value. Otherwise, when the Delaunay edge falls in the interior of the same object part, $Dis(\mathbf{T}_i, \mathbf{T}_j)$ obtains a lower value. Hence, $Dis(\boldsymbol{T}_i, \boldsymbol{T}_j)$ can be used to detect the location which has bigger density variety in the K-dimensional embedded space and indicate the object boundaries as shown in Fig. 6.

Suppose that a set of trajectory clusters $\{\boldsymbol{V}k\}_{k=1}^{K}$ is produced by the foreground trajectory clustering process. If the amount of Delaunay neighboring trajectories of trajectory cluster $\boldsymbol{V}_p$and $\boldsymbol{V}_q$ is larger than the threshold n (set as 50 experimentally), then $\boldsymbol{V}_p$ and $\boldsymbol{V}_q$ are considered as spatially adjacent. The discontinuity between cluster $\boldsymbol{V}_p$ and $\boldsymbol{V}_q$ is defined as:

$$Dis(\boldsymbol{V}_p, \boldsymbol{V}_q) = \frac{\sum_{\boldsymbol{T}_i \in \boldsymbol{V}_p, \boldsymbol{T}_j \in \boldsymbol{V}_q} Dis(\boldsymbol{T}_i, \boldsymbol{T}_j)}{\left|\left\{(\boldsymbol{T}_i, \boldsymbol{T}_j) \,|\, \boldsymbol{T}_i \in \boldsymbol{V}_p, \boldsymbol{T}_j \in \boldsymbol{V}_q, \boldsymbol{T}_j \in Nei(\boldsymbol{T}_i)\right\}\right|} \tag{8}$$

Finally, to correct the over-segmentation mistake made by the trajectory spectrum embedding, the trajectory clusters whose discontinuity is less than the setting threshold η are merged, owing to the fact that lower discontinuity value usually means a lower object boundary response. It can be observed from the experiment that the ideal boundary location can be obtained when η is equal to 0.4. Fig. 7 shows the foreground segmentation images based on this strategy, and simultaneously tests the sensitivity of the inner-cluster discontinuity on the value selection of the feature vector number K. So, the cluster merging strategy based on the density discontinuity detector can effectively combine the split "pipes" in video sequences, thereby overcoming the defects of over-segmentation during the initial trajectory clustering period. Each trajectory cluster produced by the above merging process is finally taken as an action part of the human action to build the video mid-level representation. The clear flow chart of how to merge the over-segmented trajectory clusters is shown in Fig. 8.

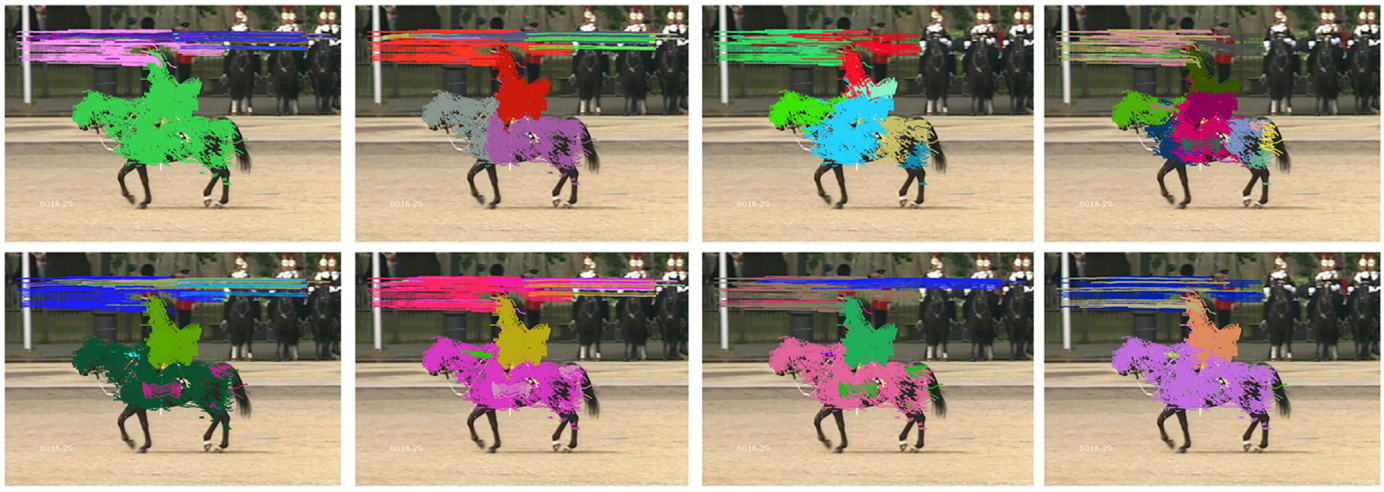

Fig. 7. Video segmentation results of original trajectory embedding (first row) and the proposed discontinuity-based cluster merging (second row) with varying numbers of eigenvectors K. From left to right: $K=5$, $K=6$, $K=10$, and $K=12$.

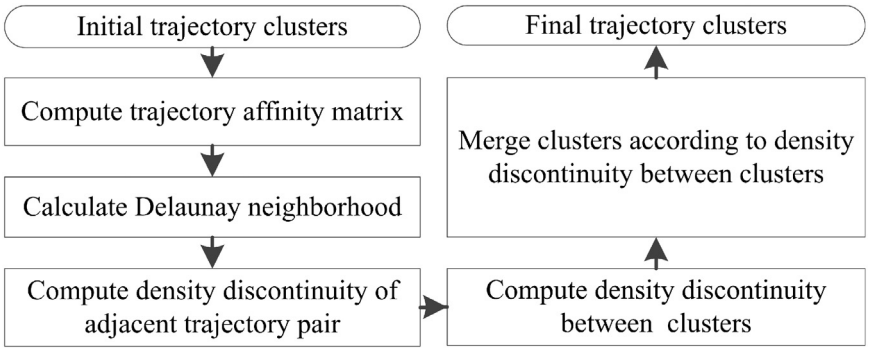

Fig. 8. A flow chart of merging over-segmented trajectory clusters.

2.4. Feature representation based on bag of histograms

It is significant for action recognition to mine the discrimination of the action parts which contain rich motion information and appearance information. Firstly, the descriptors of histogram of oriented gradient (HOG), histogram of optical flow (HOF) and motion boundary histogram (MBH) are extracted for every video frame on the multi-scale regular grids. Then, a visual dictionary for each of the three kinds of local feature descriptors is constructed by k-means, and all the feature descriptors are quantized to the nearest visual words by Euclidean distance. Finally, three histograms of HOG, HOF and MBH (denoted as $\boldsymbol{h}_l^{\mathrm{HOG}}$, $\boldsymbol{h}_l^{\mathrm{HOF}}$ and $\boldsymbol{h}_l^{\mathrm{MBH}}$, respectively) are obtained from the quantizing process. These three histograms are jointed to form the final representation of action part $\boldsymbol{V}_l$, i.e. $\boldsymbol{H}_l=\left[\boldsymbol{h}_l^{\mathrm{HOG}},\boldsymbol{h}_l^{\mathrm{HOF}},\boldsymbol{h}_l^{\mathrm{MBH}}\right]$. Thus, every action part is associated with a BoF histogram and a video sequence is represented by the BoF histogram set $\{\boldsymbol{H}_l\}_{l=1}^{L}$, i.e. the bag of histograms (BoH), where L denotes the number of clusters after the discontinuity-based cluster merging.

3. Spatio-temporal graph-based multiple-instance learning

In this study, an activity is defined as a set of action parts (i.e. 2D+t tubes), for which their class attributes are initially not known since an action class label is only assigned to the whole video clip in action classification datasets. At the same time, the exact spatio-temporal position of the specific action in each video and which action part will contribute to the recognition are both unknown. Still, there may be many action parts extracted from each video sequence, some of which may be irrelevant to the action since the foreground motion localization and boundary detection is imperfect in complex dynamic environments. All of the above-mentioned factors imply the ambiguity in the training set.

This kind of situation can be well handled by the multiple-instance learning framework. In MIL, a video sequence is treated as a "bag" of action parts in which each part is referred to as an instance associated with a BoF histogram. The action label is associated with the bag, yet individual labels of the instances are unknown. Therefore, the subsequent learning procedure operates over the bags. However, the typical MIL [36–38] assumes that instances in the bags are independently and identically distributed and thus neglects the relationship among the instances, which leads to the loss of the inner structure information in the bag. However, this assumption does not hold since there always are rich additional spatio-temporal relationships among different action parts, e.g., the spatial relation between the person and horse in the action clip of "riding horse", the relative movement of the human body and the racket in the activity "playing tennis". Intuitively, the contextual information contained in the spatio-temporal relationship is helpful to improve the accuracy of action prediction. Therefore, the spatio-temporal graph-based multiple-instance learning is proposed by mapping the training data to the set of spatio-temporal graph. And a new kernel function is designed accordingly, which is subsequently plugged into SVM for action classification.

3.1. Combination of spatio-temporal graph and multiple-instance learning

3.1.1. MIL (Multiple-Instance Learning) framework

Given the instance set $\boldsymbol{B}$ and the class label set $\boldsymbol{Y}$, assume that there is a training dataset consisting of m video samples: $\{(\mathbf{B}_1, y_1), \ldots, (\mathbf{B}_i, y_i), \ldots, (\mathbf{B}_m, y_m)\}$, where $\boldsymbol{B}_i = \{\boldsymbol{x}_{ij} : j = 1, \ldots, n_i\} \subset \boldsymbol{B}$ denotes the bag composed of n_i instances corresponding to the i-th video sample, in which $\boldsymbol{x}_{ij}$ denotes the j-th action part in the i-th video sample, corresponding to a BoF histogram $\boldsymbol{H}_{ij}$, and a class label $y_i \in \boldsymbol{Y}$ corresponds to the instance bag $\boldsymbol{B}_i$. In regard to a certain class, assert that $\mathbf{B}_i$ is a positive sample if there exists $j \in \{1, \ldots, n_i\}$ which makes $\boldsymbol{x}_{ij}$ positive example; otherwise, it is a negative one. The learning task is to learn the map function $f : 2^{\boldsymbol{B}} \rightarrow \boldsymbol{Y}$ from the training dataset which is applied to predict the class label of the unknown instance bag.

3.1.2. STG (Spatio-Temporal Graph) model

Construct the undirected spatio-temporal graph $\boldsymbol{G}_i^{\mathrm{ST}} = (\boldsymbol{V}_i, \boldsymbol{E}_i)$ for every instance bag $\boldsymbol{B}_i = \{\boldsymbol{x}_{ij} : j = 1, \ldots, n_i\}$, where the node $\boldsymbol{v} \in \boldsymbol{V}_i$ corresponds to the instance $\boldsymbol{x}_{ij}$, represented by a BoF histogram $\boldsymbol{H}_{ij}$. If the action parts $\boldsymbol{x}_{iu}$ and $\boldsymbol{x}_{iv}$ are adjacent in space (namely, the number of Delaunay neighborhood trajectories is larger than the threshold n), an edge $\mathbf{e}_{uv} \in \boldsymbol{E}_i$ between them is constructed in the graph to represent their spatio-temporal interactive relationship. Note $\boldsymbol{G}_i^{\mathrm{ST}}$ is not always full-connected under this constraint, as not every two action parts are correlative.

3.1.3. Interaction descriptor

The interactive relationship among the action parts is modeled from the relative space positions and the movement features. Specifically, given a pair of action parts (x_u, x_v) that satisfies the Delaunay neighborhood relation, the action parts are substantially the spatio-temporal pipelines (2D+t tube) constituted by trajectory clustering. The relative position relation $\boldsymbol{l}$ can be calculated as:

$$\boldsymbol{l}_{i,j}(t) = \boldsymbol{p}_t^i - \boldsymbol{p}_t^j,\ t \in o(\boldsymbol{T}_i, \boldsymbol{T}_j),\ \boldsymbol{T}_i \in \boldsymbol{x}_u, \boldsymbol{T}_j \in \boldsymbol{x}_v, \boldsymbol{T}_j \in Nei(\boldsymbol{T}_i) \tag{9}$$

where $\boldsymbol{p}_t^i$ and $\boldsymbol{p}_t^j$ are the pixels of the trajectory $\boldsymbol{T}_i$ and $\boldsymbol{T}_j$ at the time t, respectively, and $o(\mathbf{T}_i, \mathbf{T}_j)$ denotes the overlap interval of the trajectory $\boldsymbol{T}_i$ and $\boldsymbol{T}_j$ in the time domain. Similarly, the relative movement feature $\boldsymbol{m}$ of the action part $(\boldsymbol{x}_u, \boldsymbol{x}_v)$ can be calculated as:

$$\boldsymbol{m}_{i,j}(t) = \boldsymbol{l}_{i,j}(t) - \boldsymbol{l}_{i,j}(t-1) \tag{10}$$

The results of Eqs. (9) and (10) are both 2D vectors, whose components represent the values on the direction x and y, respectively. Repeat the aforementioned calculating process for all the trajectories which construct the action parts $(\boldsymbol{x}_u, \boldsymbol{x}_v)$and satisfy the Delaunay neighborhood constraint. Meanwhile, the values of $\boldsymbol{l}_{i,j}(t)$ and $\boldsymbol{m}_{i,j}(t)$ are quantized and accumulated from the orientation and the spatial distance into histogram $\boldsymbol{h}_{uv}^{\mathrm{L}}$ and $\boldsymbol{h}_{uv}^{\mathrm{M}}$. For the orientation, a whole circle is divided into 16 sections equally, and each section occupies 22.5°. For the spatial distance, four uniform quantization levels are adopted. Then, a 64-dimensional histogram combined with these two types of information can be obtained as shown in Fig. 9. Finally, $\boldsymbol{h}_{uv}^{\mathrm{L}}$ and $\boldsymbol{h}_{uv}^{\mathrm{M}}$ are normalized and pieced together to get the interaction descriptor $\boldsymbol{H}_{uv} = \left[\boldsymbol{h}_{uv}^{\mathrm{L}}, \boldsymbol{h}_{uv}^{\mathrm{M}}\right]$, which is taken as a 128-dimensional feature representation vector for the spatio-temporal graph edge $\boldsymbol{e}_{uv}$.

3.2. Spatio-temporal graph kernel function

The above process converts the training instance bags $\{\boldsymbol{B}_i\}_{i=1}^m$ into the spatio-temporal graph set $\left\{\boldsymbol{G}_i^{\mathrm{ST}}\right\}_{i=1}^m$. Correspondingly, a new spatio-temporal graph kernel function is designed to measure the similarity among different instance bags and to learn the action classifier. Given the instance bag $\boldsymbol{B}_i$ and $\boldsymbol{B}_j$ which are converted into spatio-temporal graph $\boldsymbol{G}_i^{\mathrm{ST}} = (\boldsymbol{V}_i, \boldsymbol{E}_i)$ and $\boldsymbol{G}_j^{\mathrm{ST}} = (\boldsymbol{V}_j, \boldsymbol{E}_j)$, the spatio-temporal graph kernel function is defined as:

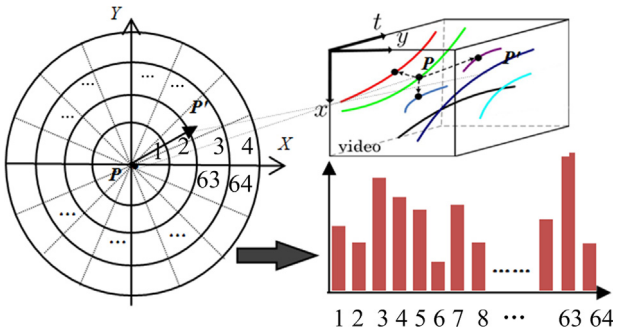

Fig. 9. Spatial relationship and motion features between action parts described by histograms (P and P' denote the pixels of adjacent trajectories in the same frame).

$$K_G(\boldsymbol{B}_i, \boldsymbol{B}_j) = \sum_{a=1}^{|\boldsymbol{V}_i|} \sum_{b=1}^{|\boldsymbol{V}_j|} k_v(\boldsymbol{x}_{ia}, \boldsymbol{x}_{jb}) + \sum_{a=1}^{|\boldsymbol{E}_i|} \sum_{b=1}^{|\boldsymbol{E}_j|} k_e(\boldsymbol{e}_{ia}, \boldsymbol{e}_{jb}) \tag{11}$$

where k_v denotes the node kernel which is used to measure the similarity of the appearance and motion features (HOG, HOF, MBH) in each single action part, and k_e denotes the edge kernel which is used to measure the similarity of the spatio-temporal interactive relationship between different action parts, respectively. By the Mercer theorem, $k_v$and k_e must be positive definite kernel functions, thus the simple χ^2 kernel is adopted. Besides, in case that a video with more trajectory clusters output a larger value, $K_G(\boldsymbol{B}_i, \boldsymbol{B}_j)$ is normalized by Eq. (12):

$$\overline{K}_G(\boldsymbol{B}_i, \boldsymbol{B}_j) = \frac{K_G(\boldsymbol{B}_i, \boldsymbol{B}_j)}{\sqrt{K_G(\boldsymbol{B}_i, \boldsymbol{B}_i)}\sqrt{K_G(\boldsymbol{B}_j, \boldsymbol{B}_j)}} \tag{12}$$

In the action classification stage, plug the spatio-temporal graph kernel K_G into SVM directly, and train the SVM classification model to get the values of the parameter $\boldsymbol{w}$ and b. In the end, classify the test instance bag $\boldsymbol{B}_z$ by Eq. (13):

$$f(\boldsymbol{B}_z) = sign\left(\sum_{i=1}^{m} \alpha_i y_i \overline{K}_G(\boldsymbol{B}_i, \boldsymbol{B}_z) + b\right) \tag{13}$$

4. Experiments

4.1. Datasets

The effectiveness of the proposed approach is evaluated on the four datasets: UCF Sports, YouTube, Kisses/Slaps, and Hollywood. The UCF Sports dataset [3] consists of 10 human action classes with 150 video sequences in total. The YouTube dataset [5] contains 11 human actions with a total of 1168 sequences. The Kisses/Slaps dataset [3] has 92 Kissing class sequences and 112 Slapping class sequences. The Hollywood dataset [1] comprises 8 daily behavior classes indoor and outdoor. All the video sequences are collected from complex dynamic scenes. Fig. 10 shows some sample frames.

4.2. Experimental setup

For the first three datasets, the leave-one-out (LOO) cross validation is adopted. In order to expand the UCF Sports, a horizontally flipped version of each sequence is introduced [39]. For the YouTube, the entire dataset is divided into 25 subsets, in which, 24 subsets are utilized to train and 1 subset is utilized to

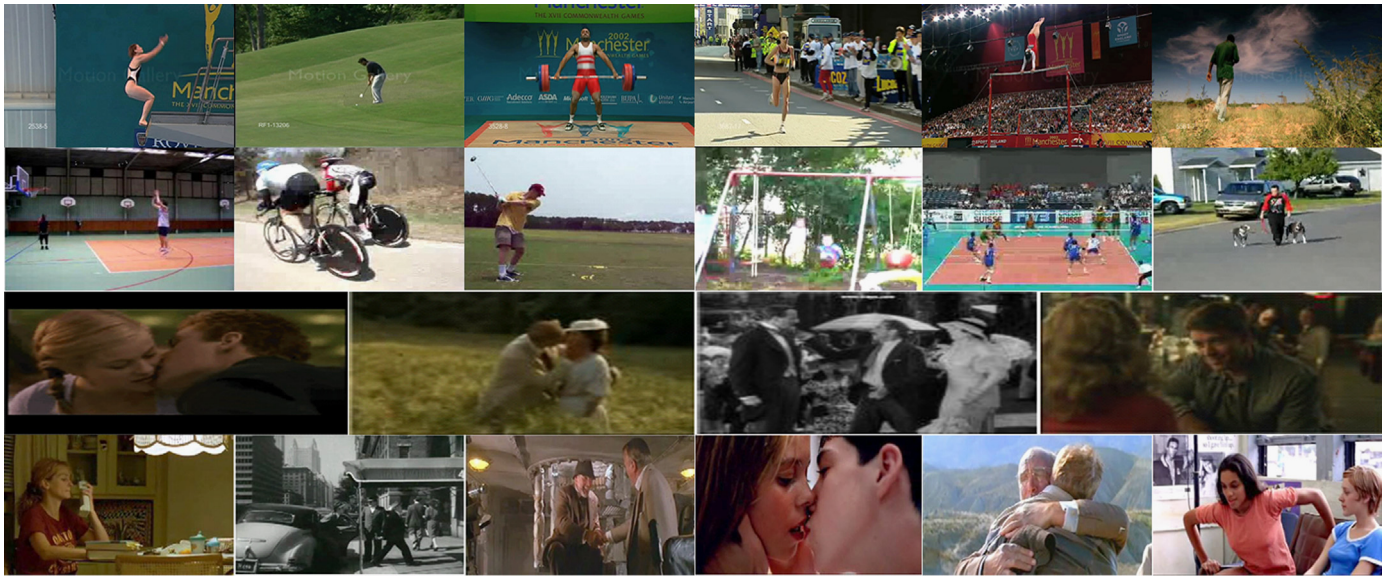

Fig. 10. Representative frames from videos in four datasets. From top to bottom: UCF Sports dataset, YouTube dataset, Kisses/Slaps dataset, and Hollywood dataset.

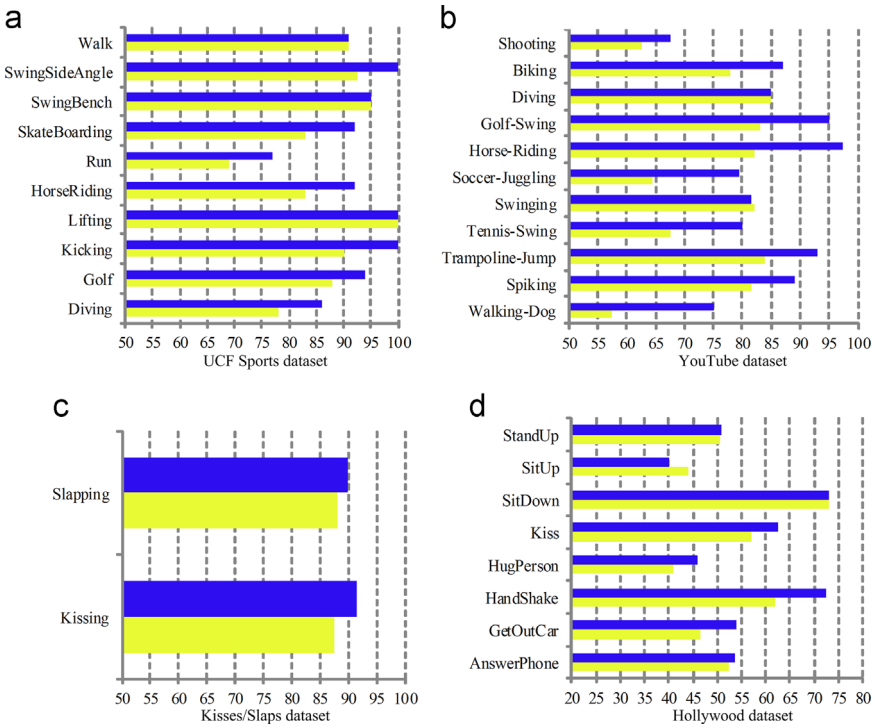

Fig. 11. Recognition performance (%) on four datasets. The performance before motion compensation is depicted in yellow and the performance after motion compensation is depicted in blue.

test. The process is repeated until every subset has been set as the test data [5]. As for the Kisses/Slaps, the experimental setup is similar to that on the UCF Sports. Concerning the Hollywood, the train-test split setting is taken in consistent with paper [1], i.e. 231 sequences are set as training data and 217 sequences are set as testing data.

In this experiment, the value of K is selected adaptively to make sure that the condition $\lambda_i < 0.2,\ i = 1, \ldots, K$ holds. The threshold of Delaunay neighborhood trajectory number n is set as 50, and the threshold of discontinuity between clusters η is set as 0.4. For the bag-of-features approach, train a codebook for each type of local descriptors using 100 000 randomly sampled features with k-means. The size of the codebook is set to be 500. With regard to the multi-class classification, adopt the APIs of LIBSVM and the one-against-rest algorithm. The optimal value of penalty parameter C is calculated by the 10-Fold Cross Validation on the training sets. Ultimately, the mean average precision (mAP) over all classes is reported.

4.3. *Effect of the camera motion compensation*

This section verifies the effectiveness of the camera motion compensation. The recognition performances on the four datasets before and after motion compensation are compared as shown in Fig. 11.

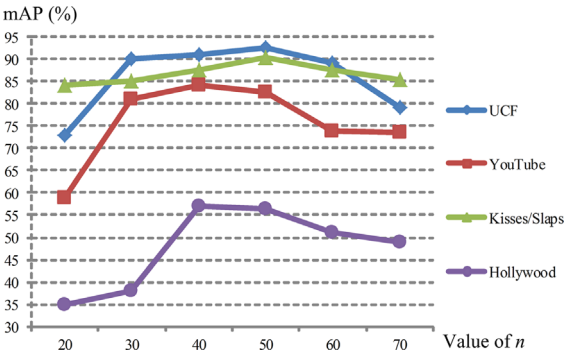

Fig. 12. mAPs (%) for different values of n on four datasets.

Fig. 13. mAPs (%) for different values of η on four datasets.

For the UCF Sports dataset, the mean recognition accuracy before and after the motion compensation is 87.15% and 92.63%, respectively, increased by nearly 5.5%. The improvement over each action class reaches 6–10% except Lifting, SwingBench and Walk, owing to the static-background recording setting for these action clips. For the other action classes, especially the Diving, Kicking and HorseRiding, the drastic background change yields many background trajectories irrelevant to the motion due to the camera's frequent fast-moving along with the actors' position. These background trajectories can further make severe noise interference to the subsequent extraction of action part and action classification.

The recognition accuracy increased from 75.21% to 84.63% after the motion compensation in the YouTube dataset. The improvement over each action class reaches 3–25% except Swinging and Diving due to their similar backgrounds. Accordingly, the motion and appearance features from the background are conducive to the recognition of these two actions. It indicates that the background of some specific action classes may contain a multitude of complementary information which benefits the foreground motion recognition.

For the Kisses/Slaps and Hollywood datasets, the improvement of mean recognition accuracy after the motion compensation reaches about 3%. Compared with the UCF Sports and the YouTube, most of the video sequences are created indoor and the actors occupy the most area of video frame. In this situation, the effect of the motion compensation using the camera affine model is not so obvious.

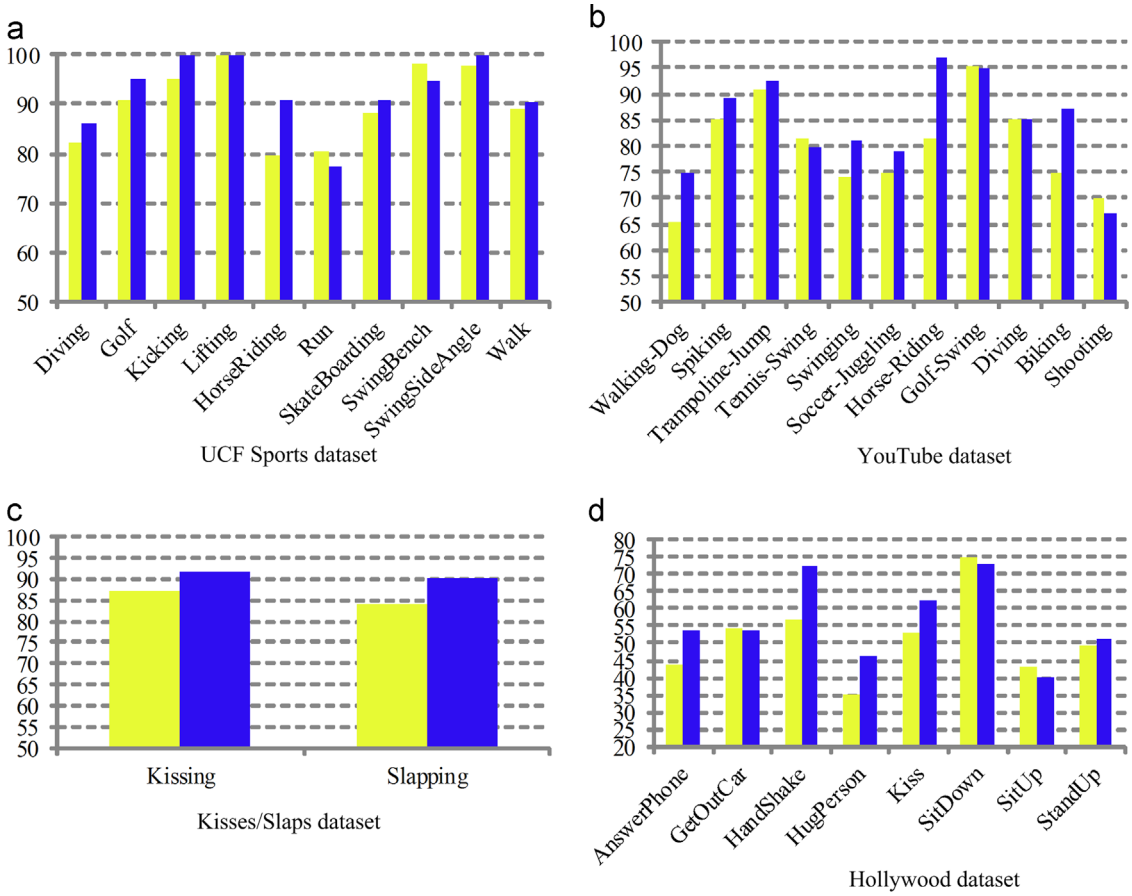

Fig. 14. Recognition performance (%) on four datasets. The performance before merging the trajectory clusters is depicted in yellow and the performance after merging the trajectory clusters is depicted in blue.

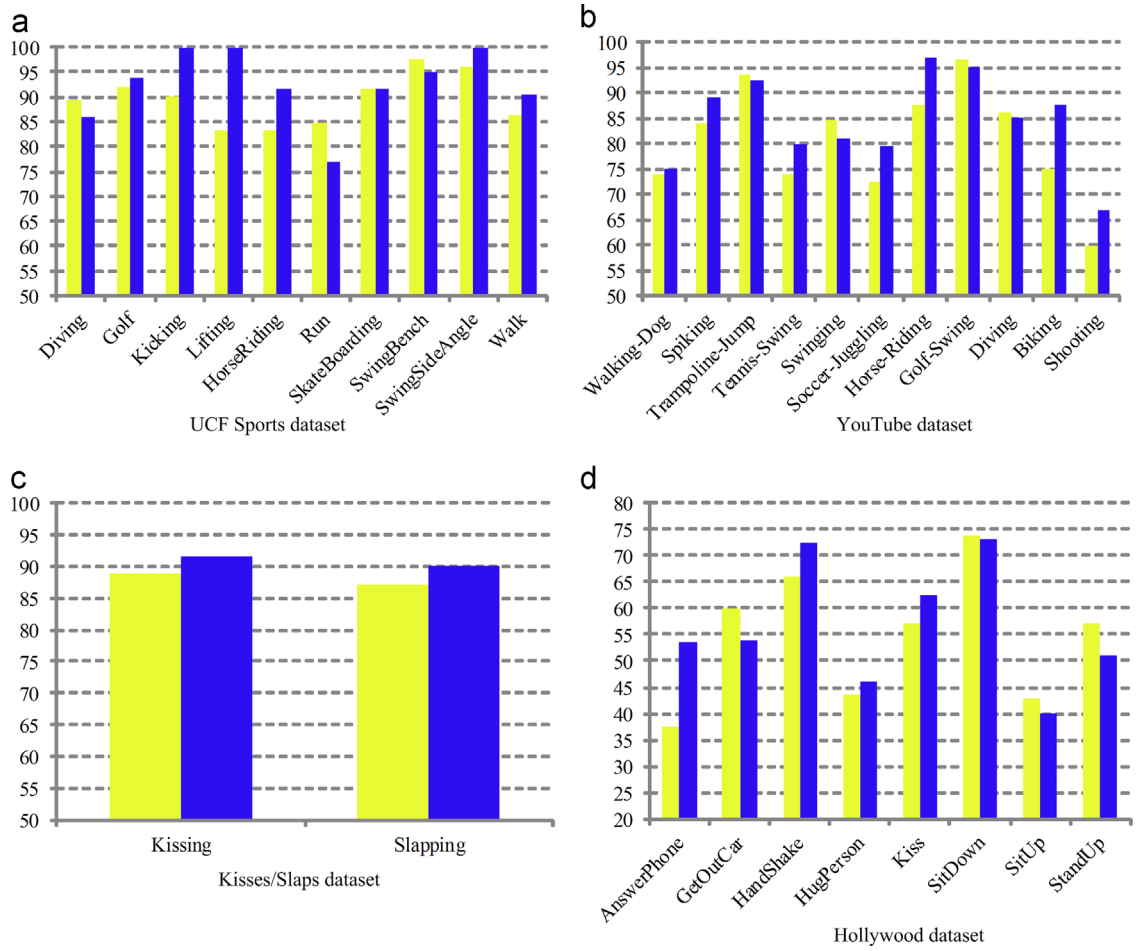

Fig. 15. Recognition performance (%) on four datasets. The performance of MIL is depicted in yellow and the performance of STG-MIL is depicted in blue.

The experimental results show that the motion compensation has a significant improvement on recognition performance. In addition, compared with the use of the whole optical flow field, the employ of the compensated optical flow to extract dense trajectories can prune the foreground trajectories by 30.81%, 23.64%, 7.32% and 8.16% in UCF Sports, YouTube, Kisses/Slaps and Hollywood, respectively. Thus, on one hand, the subsequent cost of storage and time is saved, e.g., fewer trajectories can speed up the calculation of the trajectory affinity matrix $\hat{\boldsymbol{W}}$ and the process of spectral clustering. On the other hand, the dense trajectories focused on the foreground area are also beneficial to obtain more compact and action-related motion parts for the mid-level representation.

4.4. *Effect of the density discontinuity detector*

A density discontinuity detector from the trajectory spectrum space is addressed for merging the over-segmented trajectory clusters. In this section, the effectiveness of the proposed detector is evaluated. The effect of merging trajectory clusters is influenced by two parameters: the threshold of Delaunay neighborhood trajectory number n and the threshold of the discontinuity between clusters η. The former guarantees that the merged trajectory clusters achieve a certain degree of spatial affinity, and the latter restrains that the merged trajectory clusters satisfy the condition that their boundaries should be relatively weak. Figs. 12 and 13 give the mAPs on the four datasets changing with values of parameters n and η, respectively. Here, the default value of η is fixed to 0.4 when testing n, and n is fixed to the default value of 50 when testing η.

As shown in Fig. 12, an ideal mAP of each dataset is obtained when the value of n is 40 or 50. Too small value of n (<40) would merge the trajectory clusters that are not spatio-temporal adjacent, which leads to the lack of spatio-temporal coherence on the action parts, whilst too large value of n (>50) can gradually undermine the effect of the density discontinuity detector, which results in no significant difference before and after merging.

For the parameter η, its optimal value is 0.4 for the UCF Sports and the YouTube, and 0.6 for the Kisses/Slaps and the Hollywood, respectively. As shown in Fig. 13, too many trajectory clusters are merged when η increases gradually approaching to 1, which leads to a weakening in the region segmentation information. And a significant downward trend for all the mAPs on the four datasets happens. An extreme situation is that all the trajectory clusters in a video are merged by $\eta=1$, leaving only one action part. In this case, the proposed BoH feature representation model will degenerate to the traditional BoF model (i.e. only one histogram represents a video sequence). Thus, a conclusion can be drawn that it helps to improve the action recognition under complex environments by fusing the nonlocal region segmentation information of the video sequence into the BoF model. Fig. 14 compares the recognition performance before and after merging the trajectory clusters, so as to validate the effectiveness of the density discontinuity detector. Compared with the trajectory clustering based on the embedded spectrum, merging the over-segmented trajectory clusters via utilizing the density discontinuity detector can achieve the raise of mAP by 3–5% for all the four datasets. The

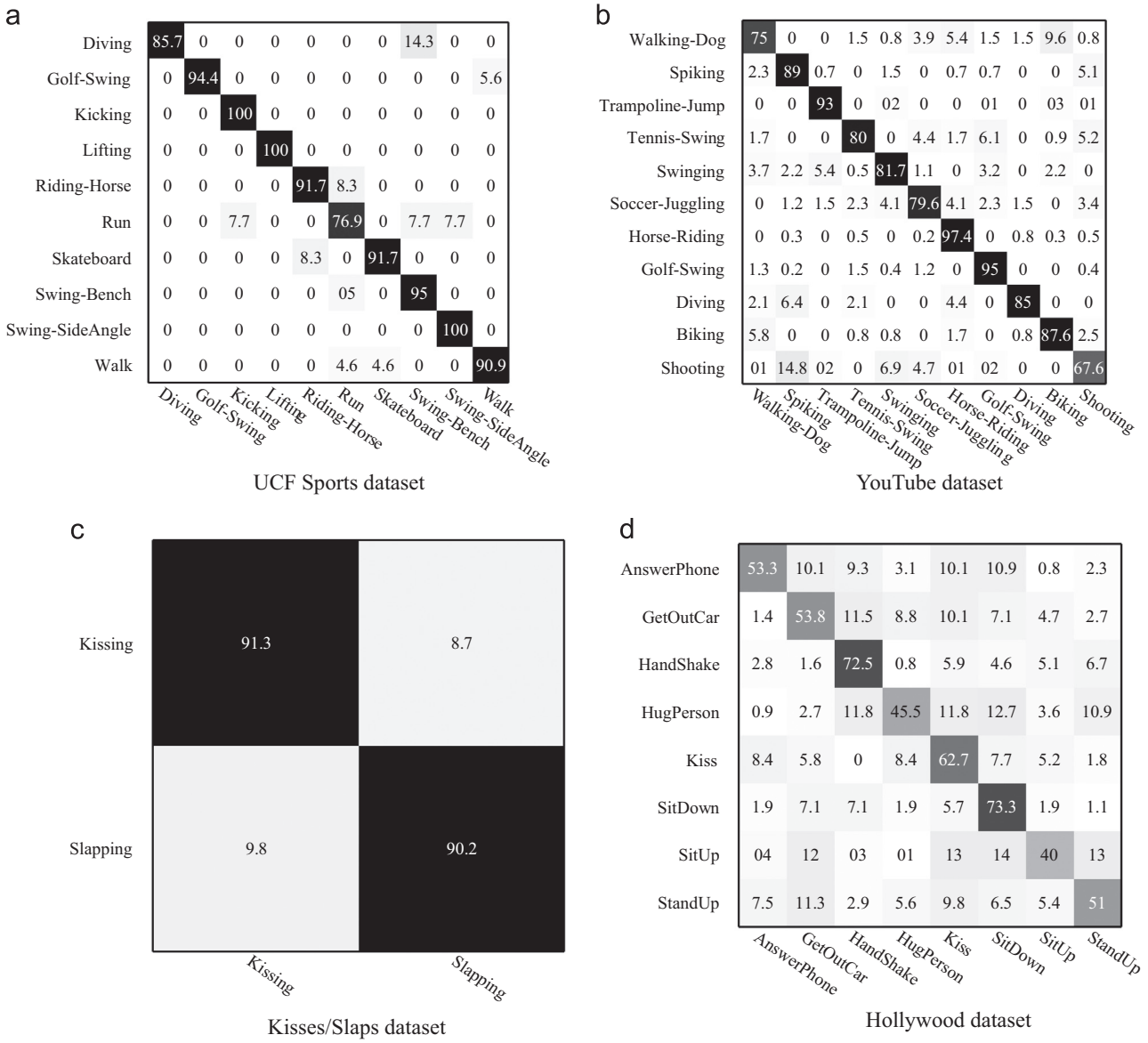

Fig. 16. Confusion matrixes by the proposed approach on four datasets. The mAPs are (a) 92.63%, (b) 84.63%, (c) 90.75%, and (d) 56.51%, respectively.

Table 1
Comparison of mAP (%) on the UCF Sports dataset.

Method	Year	mAP (%)	Representation
Liu et al. [41]	2014	**92.7**	Local
Cho et al. [42]	2014	89.7	Local
Yi et al. [43]	2013	90.08	Local
Wang et al. [39]	2013	89.1	Local
Wang et al. [44]	2013	87.3	Local
Wang et al.[45]	2012	86.6	Local
Shabani et al. [46]	2011	91.5	Local
Wu et al. [47]	2011	89.7	Local
Wang et al. [32]	2011	88.2	Local
Le et al.[48]	2011	86.5	Local
Wang et al. [49]	2013	90	Global
Yuan et al. [50]	2013	87.33	Fusion
Wu et al. [51]	2011	91.3	Fusion
The proposed method	–	92.63	Mid-level

Table 2
Comparison of mAP (%) on the YouTube dataset.

Method	Year	mAP (%)	Representation
Liu et al. [41]	2014	**85.4**	Local
Peng et al. [52]	2013	**86.56**	Local
Wang et al. [39]	2013	84.1	Local
Liu et al. [53]	2012	71.5	Local
Bhattacharya et al. [54]	2011	76.5	Local
Le et al. [48]	2011	75.8	Local
Oshin et al. [55]	2011	72.5	Local
Liu et al. [5]	2009	71.2	Local
Peng et al. [17]	2014	**93.77**	Mid-level
Zhu et al. [19]	2013	**89.4**	Mid-level
Sapienza et al. [18]	2012	**86.1**	Mid-level
Ikizler-Cinbis et al. [12]	2010	75.21	Mid-level
The proposed method	–	84.63	Mid-level

Table 3
Comparison of mAP (%) on the Kisses/Slaps dataset.

Method	Year	mAP (%)			Representation
		Kissing	Slapping	Average	
Wang et al. [44]	2013	86.3	89.6	87.95	Local
Oshin et al. [55]	2011	82.4	77.2	79.8	Local
Yeffet et al. [56]	2009	77.3	84.2	80.75	Local
Rodriguez et al. [3]	2008	66.4	67.2	66.8	Global
The proposed method	–	**91.3**	**90.2**	**90.75**	Mid-level

Table 4
Comparison of mAP (%) on the Hollywood dataset.

Method	Year	mAP (%)	Representation
Kulkarni et al. [57]	2015	**59.9**	Global
Shabani et al. [58]	2013	53.5	Local
Shabani et al. [59]	2012	**62**	Local
Wu et al. [47]	2011	47.6	Local
Raptis et al. [60]	2010	32.1	Local
Gilbert et al. [61]	2009	**56.8**	Local
Yeffet et al. [56]	2009	36.8	Local
Laptev et al. [1]	2008	38.4	Local
Raptis et al. [13]	2012	40.1	Mid-level
Han et al. [62]	2009	47.5	Mid-level
The proposed method	–	56.51	Mid-level

actions of Horse-Riding, Biking, HandShake, HugPerson and Slapping have got the most obvious promoting effect. The spectral clustering of trajectories originally has a high error rate of over-segmentation in these action sequences, bringing about a lot of noise in the subsequent feature representation, then the density discontinuity detector is introduced to effectively merge these "fragments", thereby reducing the feature noise.

4.5. Effect of the spatio-temporal graph-based multiple-instance learning

The STG model is adopted to construct the interaction relationship among the multiple action parts of video sequences and then is fused into the MIL framework. This section compares the difference in recognition performance before and after fusing the STG model by experiments, verifying the advantage of STG-MIL as compared to the MIL.

As given in Section 3.2, the corresponding kernel function of SVM merged with the STG model is the STG kernel K_G defined in Eq. (11). Before fusing STG, due to the lack of graph edge information, the corresponding SVM kernel function degenerates back to K_{MIL} by removing the second half of Eq. (11):

$$K_{MIL}(\boldsymbol{B}_i, \boldsymbol{B}_j) = \sum_{a=1}^{|\boldsymbol{V}_i|} \sum_{b=1}^{|\boldsymbol{V}_j|} k_v(\boldsymbol{x}_{ia}, \boldsymbol{x}_{jb}) \tag{14}$$

Essentially, Eq. (14) happens to be the MI-Kernel by using χ^2 kernel which is proposed by Gartner et al. [40].

Fig. 15 shows the comparison of the recognition performance before and after fusing the STG model. The mAP on UCF Sports, YouTube, Kisses/Slaps and Hollywood is increased by 3.2%, 3.9%, 3.9% and 3.2%, respectively, by fusing STG into the MIL framework. For some action classes such as Kicking, Lifting, Horse-Riding, AnswerPhone, Slapping and HandShake, the increase of recognition accuracy is significant. Note that all these actions contain obvious human–object interaction relationship, and their corresponding instance bags contain only a few instances. Utilizing the interaction information between the action parts helps to increase the amount of motion feature information and discrimination between classes. However, it is also clear that the fusion of STG has led to a decrease of recognition accuracy for some action categories, such as Diving, Run, Swinging, GetOutCar and StandUp, since these categories are relatively complex and the STG-MIL model is relatively simple, which makes it difficult to describe the internal structure information of these video sequences. In this case, the interaction information between action parts has become a kind of noise feature.

4.6. Comparison with other state-of-the-art results

The proposed approach is compared with several state-of-the-art methods whose experimental settings are the same with this paper. Fig. 16 shows the confusion matrixes on the four datasets. And the comparisons of recognition performance on the four datasets are shown in Tables 1–4, respectively.

For the UCF Sports dataset, most of the existing methods adopted local features and representations, and the recognition accuracies among different methods are rather close. Compared with the spatio-temporal interest point representations by Le et al. [48], the proposed method obtains a higher recognition accuracy of 92.63%, exceeding by nearly 6%. Due to the much more complex classifier proposed by Liu et al. [41], their performance is 0.07% better.

In Table 2, the presented method obtains a good recognition accuracy (84.63%), better than most of the local representation approaches, which is worse than the local features based method [52] (86.56%) due to its novel context-based representations and the Gaussian process by Liu et al [41] (85.4%) owing to its more robust classifier. Besides, our approach is inferior to the other mid-level representation by [17–19] since these mid-level representations employ relatively rich structure information. e.g., Peng et al. [17] preserved global structure of video sequence by multi-layer fisher vector and Zhu et al. [19] developed a two-layer structure on action classification.

Table 3 compares the results by the presented approach with the state of the art on the Kisses/Slaps dataset. The proposed method obtains the best mAP of 90.75%, higher than the best published result by 2.8%, increasing 24% as compared to the Action MACH global representation method [3], and exceeding the LTP local representation method [56] by 10%. Meanwhile, the presented approach also obtains the best results on the two action classes of Kissing and Slapping, respectively. The comparative results show that the presented approach has better robustness to the inner-class variation of human actions in the movie scenes than other methods.

As shown in Table 4, the mAP of the presented approach on the Hollywood reaches 56.51%, which has a good approximation to the local representation method by Gilbert et al. [61]. And Kulkarni [57] adopted the per-metaframe representation in the dynamic time warping framework, obtaining a better performance of 59.9%. Compared to the best result obtained by Shabani et al. [59], the mean recognition accuracy of the presented approach decreases by 5.5% due to the dark scenes in most videos, which weakens the discrimination and the amount of local descriptors. Compared with the hierarchical combination of multiple types of local features proposed by Gilbert et al. [61], the per-metaframe accounting for the large varieties associated with actions in different contexts proposed by Kulkarni [57], and the asymmetric spatio-temporal motion features proposed by Shabani et al. [59], this study only utilizes the relatively simple local feature descriptors of HOG, HOF and MBH.

5. Conclusions

This paper has proposed a mid-level approach to represent and model the spatio-temporal relationship of video for the purpose of

human activity classification in unconstrained environments. The presented approach starts from a simple, efficient and effective motion compensation procedure to make the visual motion truly related to the foreground actions. Then a density discontinuity detector is proposed for locating action part boundaries by trajectory spectral embedding. The density discontinuity is incorporated into a spectral embedding discretization process, by which multiple semantically salient parts corresponding to the action naturally pop out. Finally, the action classification problem is formulated within the multiple-instance learning framework incorporated with a spatio-temporal graph model, which can simultaneously encode the individual part properties and their pairwise interactions. Experimental results demonstrate the effectiveness of the proposed approach on four challenging benchmarks.

The future work will focus on constructing a more complex interaction model for action parts since the STG-MIL is not enough to fully model the complicated human action due to its utilized information limited to spatial position and relative movement. The star schema will be considered to fuse the geometrical relationship between action parts.

Conflict of interest statement

None declared.

Acknowledgments

The authors would like to thank Zixin Guo M.Sc. for his insightful and inspirational comments which have greatly helped us to improve the technical contents and experiments of the study. And this work was partly supported by Guangzhou Science and Technology Project (No. 2013B090500030), and Guangdong Science and Technology Project (No. 2014B010112001).

References

[1] I. Laptev, M. Marszalek, C. Schmid, B. Rozenfeld, Learning realistic human actions from movies, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
[2] M. Marszalek, I. Laptev, C. Schmid, Actions in context, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 2929–2936.
[3] M.D. Rodriguez, J. Ahmed, M. Shah, Action MACH a spatio-temporal maximum average correlation height filter for action recognition, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2008, 1–8.
[4] J.C. Niebles, C.W. Chen, F.F. Li, Modeling temporal structure of decomposable motion segments for activity classification, in: Proceedings of European Conference on Computer Vision, 2010, pp. 392–405.
[5] J. Liu, J. Luo, M. Shah, Recognizing realistic actions from videos 'in the wild', in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1996–2003.
[6] L. Gorelick, M. Blank, E. Shechtman, M. Irani, R. Basri, Actions as space–time shapes, IEEE Trans. Pattern Anal. Mach. Intell. 29 (12) (2007) 2247–2253.
[7] M. Rochan, S. Rahman, N.D. Bruce, Y. Wang, Segmenting objects in weakly labeled videos, in: Proceedings of IEEE Canadian Conference on Computer and Robot Vision, 2014, pp. 119–126.
[8] K. Tang, R. Sukthankar, J. Yagnik, F. Li, Discriminative segment annotation in weakly labeled video, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp.2483–2490.
[9] A. Prest, C. Schmid, V. Ferrari, Weakly supervised learning of interactions between humans and objects, IEEE Trans. Pattern Anal. Mach. Intell. 34 (3) (2012) 601–614.
[10] A. Prest, V. Ferrari, C. Schmid, Explicit modeling of human–object interactions in realistic videos, IEEE Trans. Pattern Anal. Mach. Intell. 35 (4) (2013) 835–848.
[11] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Trans. Pattern Anal. Mach. Intell. 32 (9) (2010) 1627–1645.
[12] N. Ikizler-Cinbis, S. Sclaroff, Object, scene and actions: combining multiple features for human action recognition, in: Proceedings of European Conference on Computer Vision, 2010 pp. 494–507.
[13] M. Raptis, I. Kokkinos, S. Soatto, Discovering discriminative action parts from mid-level video representations, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1242–1249.
[14] J. Liu, B. Kuipers, S. Savarese, Recognizing human actions by attributes, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 3337–3344.
[15] A. Jain, A. Gupta, M. Rodriguez, L.S. Davis, Representing videos using mid-level discriminative patches, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2571–2578.
[16] L. Wang, Y. Qiao, X. Tang, Motionlets: mid-level 3D parts for human motion recognition, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2674–2681.
[17] X. Peng, C. Zou, Y. Qiao, Q. Peng, Action recognition with stacked fisher vectors, in: Proceedings of European Conference on Computer Vision, 2014, pp. 581–595.
[18] M. Sapienza, F. Cuzzolin, P.H.S. Torr, Learning discriminative space-time actions from weakly labelled videos, in: Proceedings of British Machine Vision Conference, 2012, pp. 1–12.
[19] J. Zhu, B. Wang, X. Yang, W. Zhang, Z. Tu, Action recognition with actons, in: Proceedings of IEEE International Conference on Computer Vision, 2013, pp. 3559–3566.
[20] A. Ravichandran, C. Wang, M. Raptis, S. Soatto, SuperFloxels: a mid-level representation for video sequences, in: Proceedings of European Conference on Computer Vision, 2012, pp.131–140.
[21] A. Gaidon, Z. Harchaoui, C. Schmid, Activity representation with motion hierarchies, Int. J. Comput. Vis. 107 (3) (2014) 219–238.
[22] T. Brox, J. Malik, Object segmentation by long term analysis of point trajectories, in: Proceedings of European Conference on Computer Vision, 2010, pp. 282–295.
[23] K. Fragkiadaki, J. Shi, Detection free tracking: exploiting motion and topology for segmenting and tracking under entanglement, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 2073–2080.
[24] Y. Hu, L. Cao, F. Lv, S. Yan, Y. Gong, T.S. Huang, Action detection in complex scenes with spatial and temporal ambiguities, in: Proceedings of IEEE International Conference on Computer Vision, 2009, pp. 128–135.
[25] S. Ali, M. Shah, Human action recognition in videos using kinematic features and multiple instance learning, IEEE Trans. Pattern Anal. Mach. Intell. 32 (2) (2010) 288–303.
[26] B. Babenko, M.H. Yang, S. Belongie, Visual tracking with online multiple instance learning, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 983–990.
[27] D. Li, J. Wang, X. Zhao, Y. Liu, D. Wang, Multiple kernel-based multi-instance learning algorithm for image classification, J. Vis. Commun. Image Represent. 25 (5) (2014) 1112–1117.
[28] M. Sapienza, F. Cuzzolin, P.H.S. Torr, Learning discriminative space–time action parts from weakly labelled videos, Int. J. Comput. Vis. 110 (1) (2014) 30–47.
[29] Z. Li, G. Geng, J. Feng, J. Peng, C. Wen, J. Liang, Multiple instance learning based on positive instance selection and bag structure construction, Pattern Recognit. Lett. 40 (2014) 19–26.
[30] J.M. Odobez, P. Bouthemy, Robust multiresolution estimation of parametric motion models, J. Vis. Commun. Image Represent. 6 (4) (1995) 348–365.
[31] A. Chambolle, T. Pock, A first-order primal-dual algorithm for convex problems with applications to imaging, J. Math. Imaging Vis. 40 (1) (2011) 120–145.
[32] H. Wang, A. Klaser, C. Schmid, L. Cheng Lin, Action recognition by dense trajectories, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2011) 3169–3176.
[33] S. Yu, J. Shi, Multiclass spectral clustering, in: Proceedings of IEEE International Conference on Computer Vision, 2003, pp. 313–319.
[34] U. von Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (4) (2007) 395–416.
[35] D. Cai, X. He, J. Han, Document clustering using locality preserving indexing, IEEE Trans. Knowl. Data Eng. 17 (12) (2005) 1624–1637.
[36] W.J. Li, D.Y. Yeung, MILD: multiple-instance learning via disambiguation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (1) (2010) 76–89.
[37] Z. Fu, A. Robles-Kelly, J. Zhou, MILIS: multiple instance learning with instance selection, IEEE Trans. Pattern Anal. Mach. Intell. 35 (5) (2011) 958–977.
[38] A. Erdem, E. Erdem, Multiple-instance learning with instance selection via dominant sets, in: Proceedings of the IEEE International Workshop on Similarity Based Pattern Recognition, 2011, pp. 177–191.
[39] H. Wang, A. Klaser, A., C. Schmid, L. Cheng Lin, Dense trajectories and motion boundary descriptors for action recognition, Int. J. Comput. Vis. 103 (1) (2013) 60–79.
[40] T. Gartner, P.A. Flach, A. Kowalczyk, A., A.J. Smola, Multi-instance kernels, in: Proceedings of Conference on Machine Learning, 2002 pp. 179–186.
[41] L. Liu, L. Shao, F. Zheng, X. Li, Realistic action recognition via sparsely-constructed Gaussian processes, Pattern Recognit. 47 (2014) 3819–3824.
[42] J. Cho, M. Lee, H.J. Chang, S. Oh, Robust action recognition using local motion and group sparsity, Pattern Recognit. 47 (2014) 1813–1825.
[43] Y. Yi, Y. Lin, Human action recognition with salient trajectories, Signal Process. 93 (11) (2013) 2932–2941.
[44] H. Wang, C. Yuan, G. Luo, W. Hu, C. Sun, Action recognition using linear dynamic systems, Pattern Recognit. 46 (6) (2013) 1710–1718.
[45] H. Wang, C. Yuan, W. Hu, C. Sun, Supervised class-specific dictionary learning for sparse modeling in action recognition, Pattern Recognit. 45 (11) (2012) 3902–3911.

[46] A.H. Shabani, D.A. Clausi, J.S. Zelek, Improved spatio-temporal salient feature detection for action recognition, in: Proceedings of British Machine Vision Conference, 2011, pp. 1–12.
[47] S. Wu, O. Oreifej, M. Shah, Action recognition in videos acquired by a moving camera using motion decomposition of Lagrangian particle trajectories, in: Proceedings of IEEE International Conference on Computer Vision, 2011, pp. 1419–1426.
[48] Q.V. Le, W.Y. Zou, S.Y. Yeung, A.Y. Ng, Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 3361–3368.
[49] C. Wang, Y. Wang, A.L. Yuille, An approach to pose-based action recognition, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 915–922.
[50] C. Yuan, X. Li, W. Hu, H. Ling, S. Maybank, 3D $\mathscr{R}$ transform on spatio-temporal interest points for action recognition, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp .724–730.
[51] X. Wu, D. Xu, L. Duan, J. Luo, Action recognition using context and appearance distribution features, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 489–496.
[52] X. Peng, Y. Qiao, Q. Peng, X. Qi, Exploring motion boundary based sampling and spatial-temporal context descriptors for action recognition, in: Proceedings of British Machine Vision Conference, 2013, pp. 1–11.
[53] J. Liu, J. Yang, Action recognition using spatiotemporal features and hybrid generative/discriminative models, J. Electron. Imag. 21 (2) (2012) 1–10.
[54] S. Bhattacharya, R. Sukthankar, R. Jin, M. Shah, A probabilistic representation for efficient large scale visual recognition tasks, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 2593–2600.
[55] O. Oshin, A. Gilbert, R. Bowden, Capturing the relative distribution of features for action recognition, in: Proceedings of IEEE Conference on Automatic Face & Gesture Recognition 2011 pp. 111–116.
[56] L. Yeffet, L. Wolf, Local trinary patterns for human action recognition in: Proceedings of IEEE International Conference on Computer Vision, 2009, pp. 492–497.
[57] K. Kulkarni, G. Evangelidis, J. Cech, R. Horaud, Continuous action recognition based on sequence alignment, Int. J. Comput. Vis. 112 (2015) 90–114.
[58] A.H. Shabani, J.S. Zelek, D.A. Clausi, Multiple scale-specific representations for improved human action recognition, Pattern Recognit. Lett. 34 (15) (2013) 1771–1779.
[59] A.H. Shabani, D.A. Clausi, J.S. Zelek, Evaluation of local spatio-temporal salient feature detectors for human action recognition, in: Proceedings of IEEE Conference on Computer and Robot Vision, 2012, pp. 468–475.
[60] M. Raptis, S. Soatto, Tracklet descriptors for action modeling and video analysis, in: Proceedings of European Conference on Computer Vision, 2010, pp. 577–590.
[61] A. Gilbert, J. Illingworth, R. Bowden, Action recognition using mined hierarchical compound features, IEEE Trans. Pattern Anal. Mach. Intell. 33 (5) (2009) 883–897.
[62] D. Han, L. Bo, C. Sminchisescu, Selection and context for action recognition, in: Proceedings of IEEE International Conference on Computer Vision, 2009, pp. 1933–1940.

Yang Yi received the B.Sc. degree in Electronic Engineering from Fudan University, China, in 1989, and the M.Sc. and Ph.D. degrees in Machine Learning and Optimization from Northeastern University, China, in 1996 and 2002, respectively. From September 2005 to August 2006, she was a visiting scholar with the Computer Science Department, Eastern Washington University, USA. Currently, Dr. Yang Yi is with the School of Data and Computer Science, Sun Yat-sen University, China, and the Xinhua College of Sun Yat-sen University, China. Meanwhile, she is also a Fellow of SYSU-CMU Shunde International Joint Research Institute, China. She has published over 80 papers in international conferences or journals. And her current research interests include computer vision and digital image processing, especially in human action recognition by machine learning and pattern recognition.

Maoqing Lin received the B.Eng. degree in Computer Science and Technology from Hunan Normal University, China, in 2014. And she was recommended for admission to the postgraduate. Now she is a M.Sc. candidate in Computer Science and Technology in Sun Yat-sen University, China. Her research interests mainly focus on human action recognition.