Vis Comput (2018) 34:391–403
https://doi.org/10.1007/s00371-016-1345-6

ORIGINAL ARTICLE

Motion keypoint trajectory and covariance descriptor for human action recognition

Yun Yi[1,2,3] · **Hanli Wang**[1,2]

Published online: 9 January 2017

Abstract Human action recognition from videos is a challenging task in computer vision. In recent years, histogram-based descriptors that are calculated along dense trajectories have shown promising results for human action recognition, but they usually ignore motion information of the tracking points, and the relationship between different motion variables is not well utilized. To address this issue, we propose a motion keypoint trajectory (MKT) approach and a trajectory-based covariance (TBC) descriptor, which is calculated along the motion keypoint trajectories. The proposed MKT approach tracks motion keypoints at multiple spatial scales and employs an optical flow rectification algorithm to reduce the influence of camera motions and thus achieves better performance than the improved dense trajectory (IDT) approach well known in the literature. In particular, MKT is faster than IDT, because MKT does not need to use human detection and extracts fewer trajectories than IDT. Furthermore, the TBC descriptor outperforms the classical histogram-based descriptors, such as the Histogram of Oriented Gradient, Histogram of Optical Flow and Motion Boundary Histogram. Experimental results on three challenging datasets (i.e., Olympic Sports, HMDB51 and UCF50) demonstrate that our approach is able to achieve better recognition performances than a number of state-of-the-art approaches.

Keywords Human action recognition · Motion keypoint trajectory · Optical flow rectification · Trajectory-based covariance descriptor

1 Introduction

The past few years have witnessed a great success of social networks and multimedia technologies, leading to the generation of vast videos. Therefore, it is increasingly important to design automatic approaches for analyzing video contents. Among all these studies, human action recognition is one of the most attractive research directions, as it has extensive applications in video retrieval, video surveillance, human–computer interaction, and so on.

Recently, local spatial–temporal descriptors with the classical Bag-of-Words (BoW) model have shown high action recognition performance. In particular, histogram-based descriptors, which are calculated along dense trajectories [35], obtain promising results for human action recognition. Based on these low-level descriptors, a number of approaches [14,26,36,40] further promote the recognition performance. Although impressive progresses in human action recognition are achieved by recent studies, it is still challenging to recognize human actions from realistic videos owing to complex background, camera motion, view angle variation, etc.

In general, previous studies ignore the motion information of the tracking points and the relationship between different motion variables. To address these issues, we propose the motion keypoint trajectory (MKT) and trajectory-based covariance (TBC) descriptor, which can be utilized as the

✉ Hanli Wang
hanliwang@tongji.edu.cn

[1] Department of Computer Science and Technology, Tongji University, Shanghai 201804, People's Republic of China

[2] Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 200092, People's Republic of China

[3] Department of Mathematics and Computer Science, Gannan Normal University, Ganzhou 341000, People's Republic of China

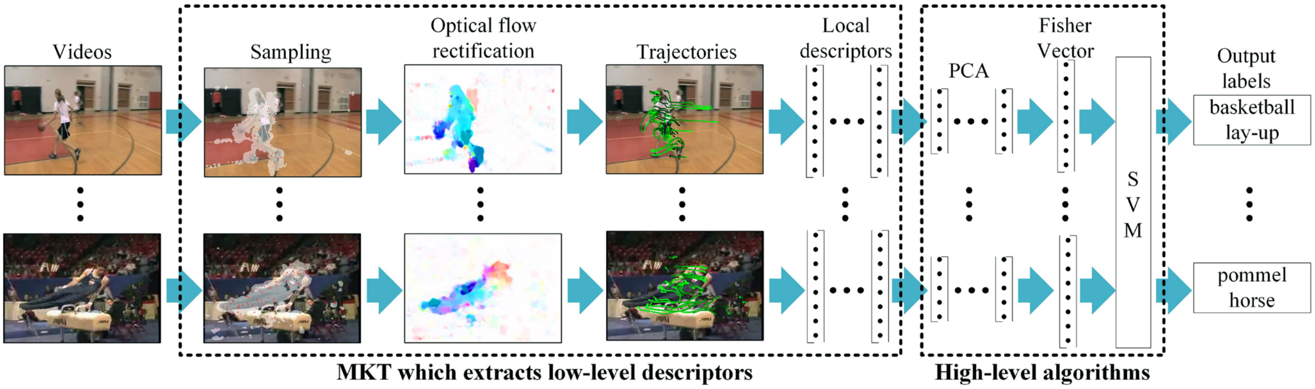


Fig. 1 An overview of the proposed system

basis of high-level algorithms for action recognition systems. After the extraction of local descriptors, the vectors of a video are separately encoded into a video-level signature vector by Fisher Vector (FV) model [29] for each descriptor, and then linear Support Vector Machine (SVM) [10] is utilized to classify actions. Extensive experiments are carried out to evaluate our approach on three challenging datasets, including Olympic Sports [25], HMDB51 [18] and UCF50 [28]. The experimental results demonstrate that TBC is able to achieve better performances than the classical histogram-based descriptors (i.e., HOG [7], HOF [20] and MBH [8]), and the proposed MKT outperforms the state-of-the-art approaches of dense trajectory (DT) [35] and improved dense trajectory (IDT) [37]. An overview of the proposed system is shown in Fig. 1, and the major contributions of this work are summarized as follows.

- The proposed MKT approach tracks motion keypoints at multiple spatial scales and dispels the influence of camera motions by an optical flow rectification approach.
- Unlike other covariance-based descriptors, the proposed TBC descriptor is formulated along trajectories and can be clustered or classified in the Euclidean space by utilizing the Log-Euclidean Riemannian metric [1].

Note that a preliminary investigation of TBC descriptor has been made in our previous work [38]. This work is different from [38] in the following aspects.

- The TBC descriptor is extracted along motion keypoint trajectories, which are better than the previous method for action recognition, and the dimension of TBC is further reduced as described in Sect. 4.3.
- We propose three hypotheses about the selection of covariance variables and perform experiments to validate these hypotheses in Sect. 5.3. Experimental results demonstrate that our selection offers a good trade-off between speed and accuracy.
- We provide more detailed description about the TBC descriptor and verify our method on more challenging dataset for action recognition.

The rest of this paper is organized as follows: Section 2 gives an overview of the related works. Section 3 describes the MKT approach in detail. The TBC descriptor is elaborated in Sect. 4. The experimental results and discussion are reported in Sect. 5. Finally, we conclude this paper in Sect. 6.

2 Related work

During the past decade, human action recognition has attracted a lot of attentions in the computer vision community. A number of researchers focus on this challenging topic. Readers are referred to [4,9,34] for comprehensive surveys of human action recognition techniques. In the next subsections, we only focus our discussion on the studies that investigate human action tracking techniques and local spatial–temporal descriptors.

2.1 Human action tracking

Owing to the wonderful ability of capturing local motion information, trajectory-based approaches have been shown to be very efficient for video representation. Messing et al. [24] used dense clouds of Kanade–Lucas–Tomasi feature tracker for action recognition. Sun et al. [31] proposed a dense long-duration trajectory extraction scheme. Wang et al. [35] designed the dense trajectory (DT) model, which tracked dense sampling points by optical flow at multiple spatial scales. Wu et al. [40] improved the DT approach by a temporal pyramid model and latent SVM. To address the camera motion problem, Wang and Schmid [37] improved the DT method by explicitly estimating camera motions. With an excellent human detection algorithm, the improved dense trajectory (IDT) method obtained state-of-the-art experimental results. In [15], the performance of DT was promoted by compensating camera motions, where the 2D affine motion model was utilized to calculate the affine flow vector, and the compensated flow was obtained by removing the affine flow vector from the optical flow vector. The performance of IDT was further promoted by utilizing the spatial–temporal pyramid and spatial FV in [36].

As opposed to these existing human action tracking techniques, the key distinctions of our tracking strategy are given below. First, we consider both salient and motion information when selecting tracking points. This sampling scheme ensures a good trade-off between computational complexity and performance. Second, the sampled motion keypoints are tracked by dense optical flow at multiple spatial scales, and four local descriptors (i.e., HOG, HOF, MBH and TBC) are employed to depict trajectories. Furthermore, we explore the potential of the approach of Vector Field Consensus (VFC) [23] as a robust point matching technique and propose a VFC-based optical flow rectification algorithm to eliminate the influence of camera motions. In particular, the VFC-based approach obtains good performance without utilizing human detection, which is usually computationally expensive.

2.2 Local spatial–temporal descriptors

The local spatial–temporal descriptors have been shown to be excellent for capturing the intrinsic characteristic of human actions. To represent the detected events, Laptev [19] proposed a scale-adapted space-time interest points (STIP) descriptor. Willems et al. [39] proposed a dense spatial–temporal feature detector. A combination of time-series representation was introduced in [16]. To encode the temporal information, a temporal sparse representation was proposed in [41]. Based on the Laban movement analysis model, a 3D descriptor was proposed in [32]. To capture the structural information, Li et al. [22] proposed a cumulative probability histogram descriptor. Among all local spatial–temporal descriptors, the histogram-based descriptors (i.e., HOG [7], HOF [20] and MBH [8]) which were calculated along dense trajectories [35] obtained excellent performances. Unlike histogram-based descriptors, Tuzel et al. [33] introduced covariance matrix as region descriptors and achieved excellent performance on object detection and texture classification. Then, the covariance-based descriptors were utilized in other fields [13,27]. For action recognition, Guo et al. [13] proposed a covariance-based descriptor to depict videos. Bilinski and Bremond [3] proposed the Video Covariance Matrix Logarithm (VCML) descriptor based on pixel-level appearance features to recognize actions. In addition to the above descriptors, there are also high-level features based on low-level descriptors such as [5,6]. To recognize human activities, Brendel and Todorovic [5] represented videos by spatiotemporal graphs, and four types of 10-bin histograms were utilized as the low-level descriptors. In [6], the temporal structures of the trajectory components were employed for action recognition, and the trajectory-based descriptors were used as the low-level features.

Unlike the existing spatial–temporal descriptors, the key characteristics of the proposed TBC descriptor are given below. First, the TBC descriptor is formulated along trajectories and does not need to utilize action segmentation or background subtraction. The TBC descriptor can be extracted with different tracking strategies, e.g., DT and IDT. Second, motion variables (i.e., the derivations of dense optical flow) are employed as covariance variables, and experiments are performed to select suitable variables for action recognition. Finally, with the Log-Euclidean metric, the matrices of TBC descriptor are projected to the Euclidean space. Extensive experiments demonstrate that the TBC descriptor obtains better performances than a number of state-of-the-art trajectory-based descriptors.

3 Proposed motion keypoint trajectory

In this section, we introduce the major components of the proposed MKT approach, including sampling and tracking, optical flow rectification and trajectory descriptors.

3.1 Sampling and tracking

The first step of MKT is to select tracking points. In general, salient points represent more information than points in homogeneous regions. Because the approach of speeded-up robust features (SURF) [2] provides a fast keypoint detector and can obtain excellent results on keypoint matching, we utilize the SURF detector to detect keypoints. Given a point $P = (x, y)$ in a frame, $\mathcal{H}(x, y)$ is the Hessian matrix at P. If the determinant of $\mathcal{H}(x, y)$ is larger than a given salient threshold Th_s, then P is selected as a candidate tracking point.

To well obtain the motion information of human actions, it is important to select the points which are moving as time goes on. For each frame, we compute its dense optical flow $w = (u, v)$, where u and v are the horizontal and vertical components of the dense optical flow. Given a point $P = (x, y)$ in a frame, its motion magnitude is defined as $M(x, y) = \sqrt{(u|_{x,y})^2 + (v|_{x,y})^2}$, where $u|_{x,y}$ and $v|_{x,y}$ are the horizontal and vertical motion value at (x, y). In general, the motion magnitudes of background points are smaller than the mean motion magnitude of a frame because foreground points are usually fewer than background points which are often static. As shown in Fig. 2, the magnitudes of most background points are smaller than the mean motion magnitude of a frame. Since the background points have less contribution

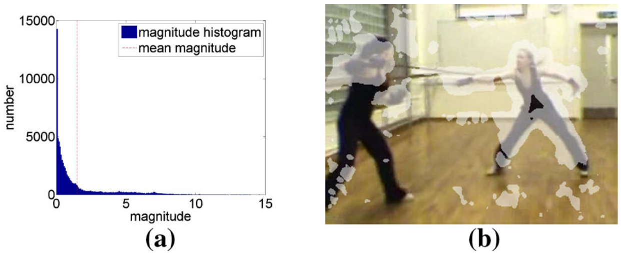


Fig. 2 Histograms of the motion magnitude and corresponding frame. **a** Motion magnitude histogram for a frame. **b** Corresponding frame where the white regions are those with the motion magnitude larger than the mean magnitude

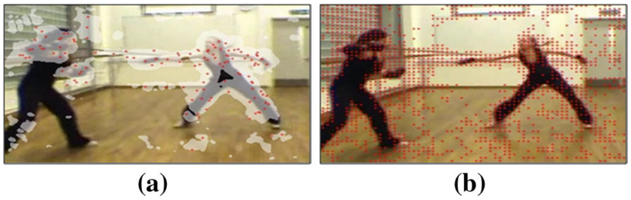


Fig. 3 Comparison between **a** the proposed motion keypoint sampling and **b** dense sampling

to action recognition, we compute the adaptive threshold of a frame as $\mathrm{Th}_m = \overline{M}$, where $\overline{M}$ is the mean motion magnitude of this frame.

For a point $P = (x, y)$, if $M(x, y) > \mathrm{Th}_m$ and $|\mathcal{H}(x, y)| > \mathrm{Th}_s$, then P is selected as a tracking point. A comparison between the dense sampling strategy and the proposed motion keypoint sampling strategy is illustrated in Fig. 3, where the white regions are those with the motion magnitude larger than Th_m, and the red points are the selected tracking points. In Fig. 3b, the points are densely sampled on a grid spaced by 5 pixels as the same as in [35]. The experimental results in Sect. 5 further demonstrate that MKT extracts fewer but more salient trajectories than IDT, since the motion keypoint sampling strategy obtains fewer but more significant points for tracking than the dense sampling strategy.

In order to obtain scale-invariant features, the sampling points are projected to multiple spatial scales, and the spatial scales are decreased by a factor of $\sqrt{2}$. The max number of spatial scales is set to 8, and the size of each spatial scale must be larger than that of the space region, which is defined in Sect. 3.3. After the dense optical flow is calculated, these points are individually tracked with optical flow at each spatial scale. Given a point $P_{m:t} = (x_{m:t}, y_{m:t})^{\mathrm{T}}$ in the spatial scale layer m of frame I_t, its tracked position in the same layer of the subsequent frame is calculated as

$$P_{m:t+1} = (x_{m:t}, y_{m:t})^{\mathrm{T}} + (K \cdot \omega_t)|_{(x_{m:t}, y_{m:t})^{\mathrm{T}}}, \tag{1}$$

where $(x_{m:t}, y_{m:t})$ is the coordinate of the given point at scale m of frame I_t, K is the 3×3 median filter, and ω_t is the 2-channel optical flow matrix of frame I_t. In particular, the matrix ω_t is first split into its horizontal and vertical components, and then, they are separately smoothed by the 3×3 median filter. To realize these functions, we utilize the OpenCV[1] toolbox.

3.2 Optical flow rectification

In realistic videos, the static background may become dynamic via camera motions; therefore, human actions may be confused by these motions. Previous researches utilize the 2D affine motion model [15] or the homography estimation with RANSAC [37] to avoid this confusion. In order to address this issue, we rectify the current frame before calculating dense optical flow. Because the global motion between two consecutive frames is usually small, we assume that two consecutive frames are related by a perspective transformation as formulated as

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} \sim \mathcal{M} \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ 1 \end{bmatrix}, \tag{2}$$

where $\mathcal{M}$ is a 2×3 perspective transformation matrix. Then, the key step of optical flow rectification is to find the perspective transformation matrix $\mathcal{M}$. To obtain matched points for computing $\mathcal{M}$, we employ two complementary strategies including optical-flow-based matches and SURF-based matches. First, the SURF keypoints are matched between the current frame I_t and its previous frame I_{t-1} by brute-force descriptor matcher. Second, the Shi and Tomasi [30] corner points are detected in each frame, and the matched points are calculated by dense optical flow. To reduce the computation complexity, we reuse the dense optical flow and SURF keypoints, which have been calculated during sampling and tracking.

Due to the complexity of unconstrained videos, there are usually a large number of false matches. To establish robust matches between points in consecutive frames, the VFC algorithm [23] is utilized to filter out the false matches, which is an efficient algorithm to establish robust correspondences between two sets of points. Because SURF-based and optical-flow-based matches represent different types of matches, we separately filter out the outliers by VFC. After the false matched points are filtered out, a normalized direct linear transform is applied to calculate the perspective transformation matrix $\mathcal{M}$. Then, the current frame is rectified by Eq. (2), and the warped frame I_t' is generated. Finally, the Gunnar Farneback algorithm [11] is utilized to calculate the dense optical flow between frame I_t' and I_{t-1}.

The effect of the proposed rectification method is visualized in Fig. 4 with two major advantages observed. First,

[1] http://opencv.org/.

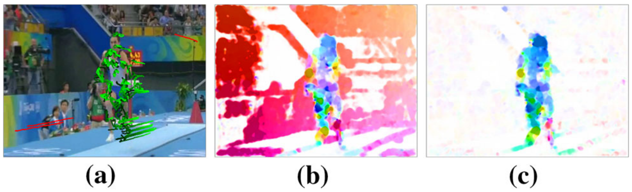


Fig. 4 An example of comparison between the original optical flow and the rectified optical flow. **a** Visualization of trajectories. **b** Optical flow before rectification. **c** Optical flow after rectification

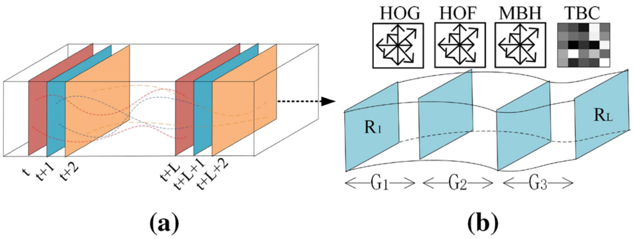


Fig. 5 Illustration of trajectory-based descriptors. **a** Trajectories. **b** A trajectory with $n_\mathrm{t} = 3$

invalid trajectories generated by camera motions can be removed by the proposed rectification method. Second, the background movements caused by camera motions are suppressed, and the foreground is enhanced as seen from the comparison between Fig. 4b and c.

3.3 Trajectory descriptors

In order to well depict the tracking information, four descriptors are calculated along trajectories, including HOG [7], HOF [20], MBH [8] and the proposed TBC descriptor. The HOF and MBH descriptors capture the local motion information, the HOG descriptor depicts the local appearance information, and the TBC descriptor represents the relationships between different motion variables. An illustration of the trajectory-based descriptors is given in Fig. 5.

To utilize the temporal information, a trajectory $T = \{G_i : i \in [1, n_\mathrm{t}]\}$ is represented by n_t temporal grids. As shown in Fig. 5b, a temporal grid $G = \{R_i(N) : i \in [1, n_\mathrm{s}]\}$ consists of n_s space regions, where N is the width and height of the square space region R aligned with a tracking point. So the frame number of a trajectory (i.e., trajectory length) is $L = n_\mathrm{t} \times n_\mathrm{s}$. In general, there are small differences between consecutive frames, so a temporal grid is described as the average feature vector of each space region within the corresponding temporal grid. To depict a trajectory, the descriptor vectors of each temporal grid are linearly concatenated according to the time stamp. In practice, we fix $n_\mathrm{t} = 3$, $n_\mathrm{s} = 5$ and $N = 32$. The HOG, HOF and MBH descriptors are computed with the same parameters as used in [37]; then, the final dimensions of HOG, HOF and MBH are 96, 108 and 192, respectively. The detailed information of our TBC descriptor is introduced in the next section.

4 Proposed trajectory-based covariance descriptor

The proposed TBC descriptor is calculated along trajectories, and it can be extracted by different tracking approaches, e.g., SIFT tracking [31], DT [35] and IDT [37].

4.1 Description of region

As we know histogram-based descriptors obtain state-of-the-art performances, but they ignore the relationships between different variables. Because the covariance matrix reflects the correlation of variables, it is utilized to depict space region. Let F denote a $W \times H \times d$ dimensional feature extracted from dense optical flow where W and H are the width and height of a frame, d is the number of variables, a mapping function Φ is defined as

$$F(x, y) = \Phi(u, v, x, y), \tag{3}$$

where u and v are the horizontal and vertical components of optical flow. There are many choices about the variables in function Φ, e.g., partial derivation, motion magnitude and motion orientation. Given a rectangular region $R \subset F$ which is selected around a tracking point, let $\{z_k, k \in [1, S]\}$ be the d-dimensional feature points inside R; an estimate of the covariance matrix for R is given by

$$\operatorname{cov}(R) = \frac{1}{S-1} \sum_{k=1}^{S} (z_k - \mu)(z_k - \mu)^\mathrm{T}, \tag{4}$$

where $S = N \times N$ is the size of region R, and μ is the mean of feature points.

Because Euclidean operations on covariance matrices suffer from several shortcomings [1], we utilize a Riemannian metric instead. In general, there are two classical distance metrics for covariance matrices, including the affine-invariant Riemannian metric [12] and the Log-Euclidean Riemannian metric [1]. As analyzed in [1], both the affine-invariant metric and the Log-Euclidean metric obtain similar performances, but the Log-Euclidean metric is much simpler and faster than the affine-invariant metric. Therefore, we utilize the Log-Euclidean metric to project covariance matrices to the Euclidean space. Let the singular value decomposition of a covariance matrix $\mathcal{X}$ be $\mathcal{X} = \mathcal{U}\Sigma\mathcal{V}^\mathrm{T}$; the matrix logarithm $\log(\mathcal{X})$ is calculated as

$$\begin{aligned}\log(\mathcal{X}) &= \mathcal{U} \cdot \ln(\Sigma) \cdot \mathcal{V}^\mathrm{T} \\ &= \mathcal{U} \cdot \operatorname{diag}(\ln(\lambda_1), \ln(\lambda_2), \ldots, \ln(\lambda_d)) \cdot \mathcal{V}^\mathrm{T},\end{aligned} \tag{5}$$

where $\Sigma = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_d)$ is a diagonal matrix of the singular value of $\mathcal{X}$; meanwhile, $\mathcal{U}$ and $\mathcal{V}$ are the orthogonal matrices. With the covariance matrix and the Log-Euclidean metric, a space region is represented as a $d \times d$ symmetric matrix.

4.2 Selection of covariance variables

Motion information is important to classify human actions from videos. A visualization of optical-flow-based variables is shown in Fig. 6, from which we can discover that both the horizontal and vertical components of optical flow contain the motion information of human actions. Furthermore, the magnitude and orientation of optical flow components and the corresponding first-order partial derivatives with respect to x and y (i.e., motion boundaries) also reveal motion cues.

The selection of covariance variables is vital to descriptors based on covariance. We propose three hypotheses about selecting covariance variables. First, the magnitude and orientation of optical-flow-based variables are effective for capturing motion information. Second, unlike the first-order partial derivatives of optical flow with respect to x and y, the second-order partial derivatives of optical flow with respect to x and y have few motion cues. Finally, the partial derivatives of optical flow with respect to time have little influence on action recognition. Massive experiments in Sect. 5.3 have been performed to verify the aforementioned hypotheses.

Let the magnitude function be defined as $\mathrm{mag}(x, y) = \sqrt{x^2 + y^2}$, and the orientation function as $\mathrm{atan}(x, y) = \arctan(x, y)$ where $\arctan(\cdot)$ is the arc tangent function; we select the optical-flow-based variables to form the mapping function Φ as

$$\begin{aligned}\Phi = \Bigg[& x, y, u, v, \mathrm{mag}(u, v), \mathrm{atan}(u, v), \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, \\ & \frac{\partial v}{\partial x}, \frac{\partial v}{\partial y}, \mathrm{mag}\left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right), \mathrm{atan}\left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right), \\ & \mathrm{mag}\left(\frac{\partial v}{\partial x}, \frac{\partial v}{\partial y}\right), \mathrm{atan}\left(\frac{\partial v}{\partial x}, \frac{\partial v}{\partial y}\right)\Bigg]^{\mathrm{T}}, \end{aligned} \tag{6}$$

where x and y indicate the location of the dense optical flow, u and v are the horizontal and vertical components of the optical flow. In order, the follow-up optical-flow-based variables include the magnitude and orientation of u and v, the first-order partial derivatives of u and v with respect to x and y, the magnitude and orientation of $\frac{\partial u}{\partial x}$ and $\frac{\partial u}{\partial y}$, the magnitude and orientation of $\frac{\partial v}{\partial x}$ and $\frac{\partial v}{\partial y}$.

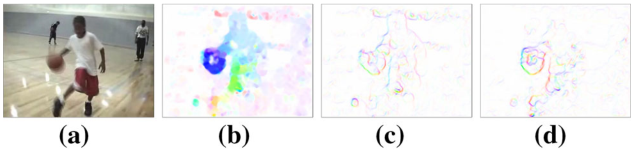


Fig. 6 Visualization of optical-flow-based variables. **a** Original frame. **b** Optical flow. **c** Horizontal motion boundaries. **d** Vertical motion boundaries

4.3 Covariance description of trajectory

After the covariance matrix is projected to the Euclidean space, it is further converted to a vector by using its upper triangular matrix elements. As introduced in Sect. 4.2, the first and second variables of the covariance matrices $\mathcal{X}$ represent the horizontal and vertical positions, so the elements of $\mathcal{X}_{(1,1)}$, $\mathcal{X}_{(1,2)}$, $\mathcal{X}_{(2,1)}$ and $\mathcal{X}_{(2,2)}$ are the same in each matrix $\mathcal{X}$. Therefore, these elements are deleted when vectorizing matrix, and the dimension is shortened to $(d(d+1)/2 - 3)$. Then, we calculate the mean vector of space regions within a temporal grid. The final TBC descriptor for a trajectory is the linear concatenation of these vectors of temporal grids along the corresponding trajectory.

5 Experimental results

In this section, we report the comparison and analysis of the proposed MKT and TBC approaches on three challenging action recognition datasets including Olympic Sports (OlycSpos) [25], HMDB51 [18] and UCF50 [28].

5.1 Datasets

The introductions and experimental protocols for the three datasets are described in this section. Figure 7 shows some examples from these datasets. We follow the standard evaluation protocols by reporting the mean average precision (mAP) over all classes for OlycSpos and average accuracy for HMDB51 and UCF50.

The OlycSpos dataset [25] contains videos of different sports. There are 16 kinds of actions represented by a total of 783 video sequences. We use 649 video sequences for training and the other 134 sequences for testing. The performance is evaluated with the mAP over all classes as recommended in [25].

The HMDB51 dataset [18] is collected from a variety of sources. There are a total of 6766 videos distributed in 51 action categories. For evaluation, there are three distinct training and testing splits. We follow the original protocol using three train–test splits [18] and report average accuracy over these three splits.

The UCF50 dataset [28] has 50 action categories and 6618 videos, which are downloaded from YouTube. For all of these 50 categories, the videos are split into 25 groups. As suggested in [28], we apply the Leave-One-Group-Out cross-

Fig. 7 Sample frames from three human action recognition datasets. The *top row* is from OlycSpos, the *middle row* is from HMDB51 and the *bottom row* is from UCF50

validation experimental setup and report average accuracy over all classes.

5.2 Experimental setup

In order to classify videos, the conventional approach to describing a video is to extract feature vectors with low-level local descriptors. Then, these vectors are encoded into a high-dimensional video-level signature vector to represent this video. Among all encoding techniques, the Fisher Vector (FV) encoding method [29] achieves excellent performances on image classification and action recognition. According to this conventional approach, we first extract feature vectors of TBC and three baseline descriptors (i.e., HOG, HOF and MBH). In order to fairly compare with DT [35] and IDT [37], the same parameters are utilized to extract these baseline descriptors as used in [37].

After feature extraction, the principle component analysis (PCA) is individually applied to reduce the dimensionality of these descriptors (i.e., HOG, HOF, MBH and TBC) by a factor of two as suggested in [29,37], so as to better fit the diagonal covariance matrix assumption [29]. For each video, these PCA-reduced vectors are separately encoded into a signature vector by the FV method [29]. For each descriptor, we randomly select 256,000 training samples to learn the PCA projection as suggested in [37], and the Gaussian mixture models (GMM) are, respectively, learned based on these PCA-reduced vectors. In all experiments, we set the number of GMM for FV generation to 256 as the same as in [37].

After encoding, each vector is normalized with the signed square root and ℓ_2 normalization. To utilize the spatial–temporal location information of video contents, we employ the Spatial–Temporal Pyramid (STP) representation [21] and divide one video in two temporal parts and three spatial parts as used in [36]. When utilizing STP, we separately encode the local features in different spatial–temporal grids to obtain the related FV representations and then concatenate these FVs with the vector computed over the entire video.

Data augmentation (DA) is an efficient scheme to increase the amount of training samples, which has been utilized in image classification [17] and video classification [14]. Based on the observation that a video and its left–right mirrored video depict the same human action, we double the amount of training samples by adding videos obtained by left–right flipping.

When using multiple descriptors, we utilize early fusion to linearly concatenate the normalized vectors together. To achieve the balance of training samples, we weight positive and negative samples in an inverse manner. In the following experiments, the standard linear SVM [10] is used with the penalty parameter C equal to 100, which is the same configuration applied in [37].

5.3 Evaluation of parameters

The selection of covariance variables is important for covariance-based descriptors. To evaluate the aforementioned three hypotheses discussed in Sect. 4.2, three additional mapping functions are defined as follows.

$$\Phi_1 = \left[x, y, u, v, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial v}{\partial x}, \frac{\partial v}{\partial y}\right]^{\mathrm{T}}, \tag{7}$$

$$\Phi_2 = \left[\Phi, \frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 u}{\partial y^2}, \frac{\partial^2 v}{\partial x^2}, \frac{\partial^2 v}{\partial y^2}\right]^{\mathrm{T}}, \tag{8}$$

$$\Phi_3 = \left[\Phi, \frac{\partial u}{\partial t}, \frac{\partial v}{\partial t}\right]^{\mathrm{T}}, \tag{9}$$

where x and y indicate the location of the dense optical flow, u and v are the horizontal and vertical components of the optical flow, $\frac{\partial u}{\partial x}$, $\frac{\partial u}{\partial y}$, $\frac{\partial v}{\partial x}$ and $\frac{\partial v}{\partial y}$ are the first-order partial derivatives of u and v with respect to x and y, Φ is the mapping function defined in Eq. (6), $\frac{\partial^2 u}{\partial x^2}$, $\frac{\partial^2 u}{\partial y^2}$, $\frac{\partial^2 v}{\partial x^2}$ and $\frac{\partial^2 v}{\partial y^2}$ are the second-order partial derivatives of u and v with respect to x and y, $\frac{\partial u}{\partial t}$ and $\frac{\partial v}{\partial t}$ are the first-order partial derivatives of u and v with respect to time.

Table 1 Comparison of variable selection on HMDB51

Approach	Number	Accuracy (%)
TBC with Eq. (7)	8	50.6
TBC with Eq. (6)	14	54.3
TBC with Eq. (8)	18	54.5
TBC with Eq. (9)	16	50.3

To evaluate the performance of different variables, we calculate the TBC descriptor with the aforementioned four mapping functions on the HMDB51 dataset. Except the difference of mapping functions, we fix other parameters to the default values as described in Sect. 5.2. The comparison results are listed in Table 1. On this dataset, "TBC with Eq. (6)" achieves better performances than "TBC with Eq. (7)." This demonstrates that the additional variables in Eq. (6) including magnitude and orientation of optical flow derivatives promote the performance for action recognition. By adding four second-order partial derivatives, "TBC with Eq. (8)" is 0.2% better than "TBC with Eq. (6)." This testifies that the second-order partial derivatives in Eq. (8) have a few influence on action recognition. Furthermore, the first-order partial derivatives of optical flow components with respect to time in Eq. (9) reduces the performance for action recognition. In practice, we utilize Eq. (6) as the mapping function for TBC in the next experiments, because Eq. (6) offers a good trade-off between computational efficiency and accuracy.

To quantify the improvement of the proposed optical flow rectification (OFR), we employ the techniques of camera motion compensation (CMC) [15] and camera motion estimation (CME) [37] as the baseline approaches for comparison. We implement these baseline approaches with our proposed method according to [15,37] and select TBC descriptor to depict trajectories. In particular, we utilize the publicly available Motion2D software[2] to calculate the affine flow vector as in [15], use the code of CME and the bounding boxes of human detection[3] provided by [37].

The evaluation results are listed in Table 2, where CMEHD is the CME approach with human detection. By comparing "TBC without OFR" and "TBC with CMC," we find that the performance of TBC is not improved with CMC, and the similar results of MBH are also obtained in Table 3 of [15]. This may be due to the reason that CMC is not suitable for descriptors which cancel the constant motion, such as MBH and TBC. From Table 2, we also discover that both CME and OFR improve the performances of TBC descriptor, and "TBC with OFR" achieves the best results on these datasets.

5.4 Comparison with baseline descriptors

A number of experiments have been carried out to quantify the improvement obtained by our TBC descriptor as compared to three baseline descriptors (i.e., HOG, HOF and MBH). To compare in a fair manner, both TBC and baseline descriptors are extracted along the proposed MKT trajectories and employ the same parameters as presented in Sect. 5.2. The comparison between our TBC descriptor and baseline descriptors is given in Table 3, where "HOG + HOF + MBH" is the method of combining these three descriptors and "Combined All" is the combination of the four descriptors (i.e., HOG, HOF, MBH and TBC). In this experiment, we utilize the early fusion strategy to directly concatenate vectors of these descriptors before classification.

As given in Table 3, the TBC descriptor outperforms the other three baseline descriptors, since it captures more information of human actions than the baseline descriptors. In particular, TBC, respectively, obtains 6.9, 16.9 and 6.2% higher recognition performance than HOG on OlycSpos, HMDB51 and UCF50. This experiment verifies that the covariance of optical-flow-based variables along trajectories is more robust than histogram-based descriptors. Furthermore, the combination of TBC with baseline descriptors achieves the best performance, which demonstrates that the TBC descriptor and baseline descriptors complement each other. Figure 8 shows the confusion matrices for combined descriptors on HMDB51 and UCF50. The errors mainly occur between classes which are visually similar, like "sword exercise" and "draw sword" on HMDB51, "Swing" and "Tennis Swing" on UCF50.

5.5 Comparison with baseline trajectories

Due to the excellent performances obtained by DT [35] and IDT [37], they are selected as the baseline tracking approaches. The default parameters of the baseline approaches are set as the same as in [35,37]. To obtain the best performance of IDT, human detection is utilized. For a fair comparison, the other parameters of both baseline approaches and MKT are configured as the same as presented in Sect. 5.2. The combination of the four descriptors (i.e., "Combined All" in Table 3) is utilized to evaluate the performance, and the results are reported in Table 4, where we report the mAP over all classes for OlycSpos [25], average accuracy over three train–test splits for HMDB51 [18], and average accuracy over all classes for UCF50 [28].

As seen from the results, the recognition performances of MKT, respectively, outperform DT by 4.8, 5.4 and 3.3% on the three datasets. As compared with DT and IDT, MKT obtains the best recognition performance, since MKT tracks

[2] http://www.irisa.fr/vista/Motion2D/.

[3] http://lear.inrialpes.fr/people/wang/improved_trajectories.

Table 2 Evaluation of optical flow rectification

Approach	OlycSpos (%)	HMDB51 (%)	UCF50 (%)
TBC without OFR	83.5	49.4	86.7
TBC with CMC [15]	82.3	48.6	86.4
TBC with CME [37]	86.7	52.2	88.6
TBC with CMEHD [37]	87.8	53.6	88.8
TBC with OFR	89.4	54.3	89.2

Table 3 Comparison of TBC with baseline descriptors

Approach	OlycSpos (%)	HMDB51 (%)	UCF50 (%)
HOG	82.5	37.4	83.0
HOF	87.6	50.7	87.4
MBH	89.3	53.0	89.0
TBC	89.4	54.3	89.2
HOG + HOF + MBH	92.5	59.2	92.1
Combined all	93.2	60.2	92.8

motion keypoints and utilizes a VFC-based optical flow rectification algorithm to eliminate the influence of camera motions.

5.6 Computational complexity

To fast calculate the covariance matrices, the integral image is employed in this work to compute TBC descriptor. As described in [33], the computational complexity of constructing the integral images is $O(d^2WH)$, where d is the number of covariance variables, W and H are the width and height in pixel of a video frame. After the construction of integral images, the TBC descriptor of a region can be calculated in the complexity of $O(d^2)$. So the computation of TBC is fast owing to the use of integral image.

To obtain good recognition performance, the IDT approach uses human detection (HD), while MKT obtains better performances than IDT without using HD. Generally speaking, the process of HD is time-consuming, so MKT is faster than IDT with HD. Moreover, we also compare the feature extraction speed of MKT and IDT without HD and show the comparison on the HMDB51 dataset in Table 5 for example. In particular, all the experiments are run at a PC with Intel I7 (3.6 GHz CPU) and only a single CPU core is used.

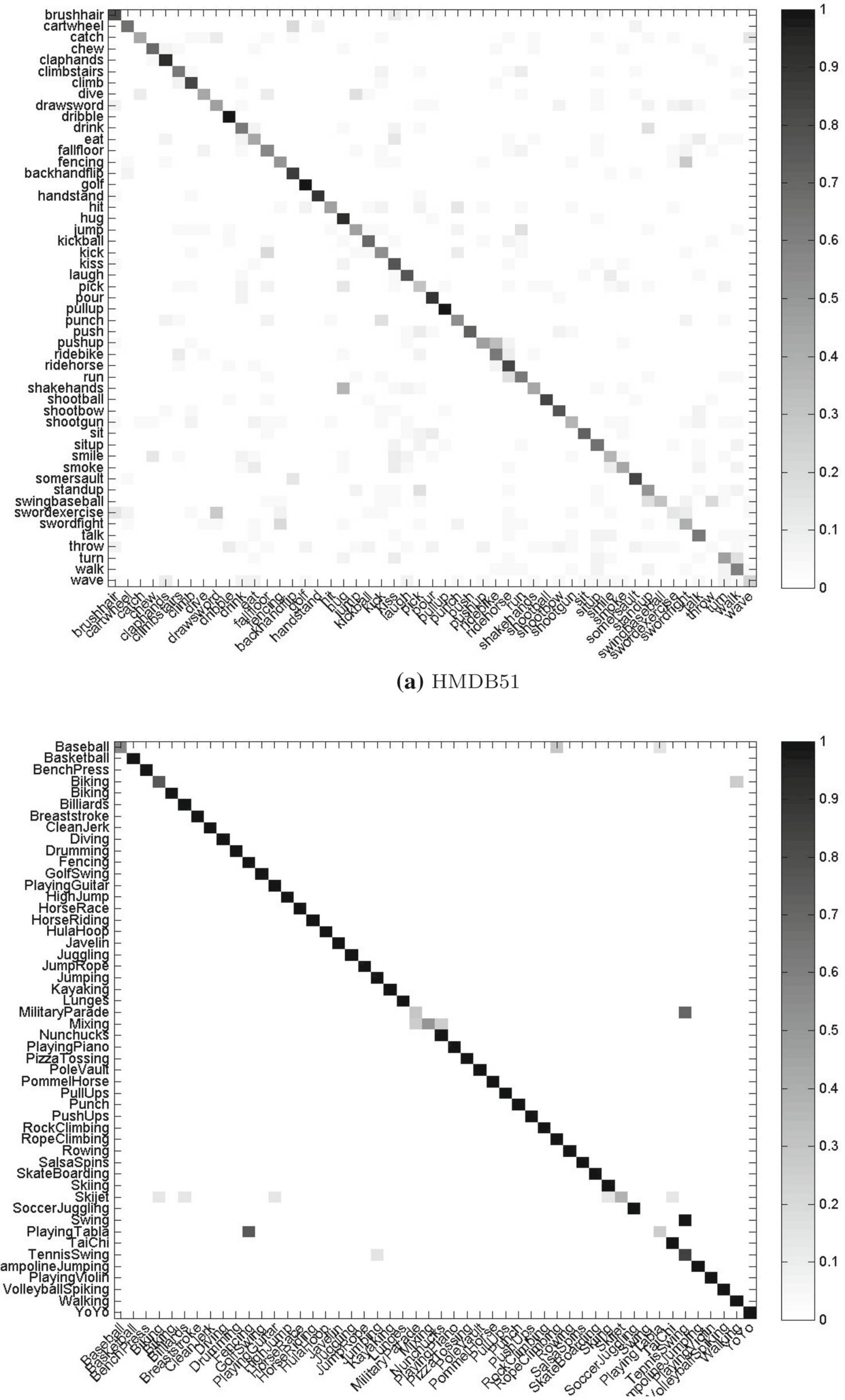
In Table 5, “Total” represents the total number of extracted trajectories, “Percent” stands for the percentage of valid trajectories, and the processing speed is reported in frames-per-second (fps). As observed from the results, MKT extracts fewer trajectories than IDT without HD since the motion keypoint filter is utilized by MKT, so the MKT approach is faster than IDT without HD.

Regarding memory consumption, it is mainly determined by the resolution of videos and the number of extracted descriptors. In order to quantify the memory requirements, experiments are performed on a 240 × 320 video with four descriptors (i.e., HOG, HOF, MBH and TBC) extracted, for instance. For this video, the maximum memory consumption required by MKT is about 0.25 GB and that of IDT is approximate 0.29 GB. The memory requirements of IDT are higher than those of MKT, because IDT extracts more trajectories than MKT.

5.7 Comparison with state of the arts

The MKT approach with data augmentation (DA) and Spatial–Temporal Pyramid (STP) is compared with other state-of-the-art approaches in Table 6, where “$A + B$” is the approach combining the techniques of A and B, and “–” indicates that no available results are reported by the cited publications. On all of the three datasets, the MKT approach outperforms other competing methods. As listed in Table 6, most state-of-the-art approaches are based on STIP [19], DT [35] or IDT [37]. The performance of DT is improved by utilizing Spatial Fisher Vector (SFV) and STP in [26]. In [15], the performance of DT is promoted by CMC and Divergence–Curl–Shear (DCS) descriptor. In [40], the recognition accuracy of DT is boosted by using Temporal Pyramid Model (TPM) and latent SVM. In [14], two complementary techniques are proposed to improve the recognition performance of IDT, including the Subsequence-Score Distribution (SSD) and Relative Class Scores (RCS). To capture the structural information, the Cumulative Probability Histogram (CPH) descriptor is proposed in [22] based on STIP. The performance of IDT is improved by the Video Covariance Matrix Logarithm (VCML) in [3]. As introduced in [36], the performance of IDT is further improved by utilizing SFV and STP.

Fig. 8 Confusion matrices for combined descriptors on **a** HMDB51 and **b** UCF50



(a) HMDB51

(b) UCF50

Table 4 Comparison of MKT with baseline trajectories

Approach	OlycSpos (%)	HMDB51 (%)	UCF50 (%)
DT	88.4	54.8	89.5
IDT	91.2	58.3	92.0
MKT	93.2	60.2	92.8

Table 5 Speed comparison of feature computation on HMDB51

Approach	Total	Percent (%)	fps
IDT without HD	180,987,836	44.8	2.16
MKT	77,457,827	66.9	3.22

As given in Table 6, only TBC descriptor achieves promising results, as all state-of-the-art methods in this table utilize multiple descriptors and some high-level algorithms (e.g., DA, STP and SFV) are used to improve their performance. From the results in the bottom of this table, our "MKT + DA" is superior to the MKT approach on three datasets, because the training samples are doubled. On the HMDB51 and UCF50 datasets, the method of "MKT + DA + STP" outperforms the method of "MKT + DA," but the STP strategy fails to promote the performance on the OlycSpos dataset. This phenomenon can also be observed by comparing the results reported in [36,37]. On the whole, the experimental results of our approach can be further promoted by the DA and STP strategies. As we currently focus on extracting low-level descriptors, the performance of the proposed approach may be further promoted by other high-level strategies, e.g., SFV, SSD and RCS.

6 Conclusion

In this paper, a new tracking approach MKT is proposed and a novel descriptor TBC is designed for human action recognition. The MKT approach tracks motion keypoints at multiple spatial scales, and the VFC-based optical flow rectification algorithm is designed to eliminate the influence of camera motions on action recognition. Experimental results demonstrate that the proposed MKT outperforms the baseline approaches (e.g., DT and IDT). In recent years, many excellent systems utilize the baseline approaches to extracting trajectory-based descriptors as low-level features and develop high-level algorithms based on these low-level features. In order to further improve the recognition performance, action recognition systems can utilize MKT instead of these baseline approaches to extract low-level descriptors (e.g., HOG, HOF, MBH and TBC).

Regarding TBC, it is based on the covariance matrix representation of trajectory and captures the linear relationships between the derivations of dense optical flow. Note that the TBC descriptor can be calculated not only along MKT trajectories but also other trajectories. The experiments demonstrate that the TBC descriptor outperforms three classical trajectory-based descriptors, and these descriptors are complementary to each other.

Furthermore, the recognition performance of our approach can be further promoted by the improving strategies (i.e., DA and STP). Extensive experiments demonstrate that the proposed approach is superior to other state-of-the-art approaches. Our approach is easy to implement, and the source code of the approach will be made available to the public. In the future, we will focus our research on designing more discriminative features for human action recognition.

Table 6 Comparison with state-of-the-art approaches, where we report mAP for OlycSpos and average accuracy for HMDB51 and UCF50

Approach	OlycSpos (%)	HMDB51 (%)	UCF50 (%)
DT + STP [35]	77.2	48.3	85.6
DT + SFV + STP [26]	82.1	54.8	90.0
DT + CMC + DCS [15]	83.2	52.1	–
IDT + FV [37]	**91.1**	57.2	91.2
DT + TPM [40]	84.3	47.1	–
IDT + DA + RCS [14]	–	**60.8**	–
STIP + CPH [22]	–	29.6	–
IDT + VCML [3]	–	58.6	**92.1**
IDT + SFV + STP [36]	90.4	60.1	91.7
TBC	89.4	54.3	89.2
MKT	93.2	60.2	92.8
MKT + DA	**94.1**	62.5	93.2
MKT + DA + STP	93.3	**64.2**	**93.7**

Best results of other approaches and our approach are in bold, respectively

Acknowledgements This work was supported in part by the National Natural Science Foundation of China under Grants 61472281 and 61622115, the Program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning (No. GZ2015005), and the NSF of Jiangxi Province under Grant 20161BAB202069.

References

1. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Log-Euclidean metrics for fast and simple calculus on diffusion tensors. Magn. Reson. Med. **56**(2), 411–421 (2006)
2. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. In: European Conference on Computer Vision, pp. 404–417 (2006)
3. Bilinski, P., Bremond, F.: Video covariance matrix logarithm for human action recognition in videos. In: International Joint Conference on Artificial Intelligence, pp. 2140–2147 (2015)
4. Borges, P.V.K., Conci, N., Cavallaro, A.: Video-based human behavior understanding: a survey. IEEE Trans. Circuits Syst. Video Technol. **23**(11), 1993–2008 (2013)
5. Brendel, W., Todorovic, S.: Learning spatiotemporal graphs of human activities. In: International Conference on Computer Vision, pp. 778–785 (2011)
6. Cheng, G., Huang, Y., Wan, Y., Buckles, B.P.: Exploring temporal structure of trajectory components for action recognition. Int. J. Intell. Syst. **30**(2), 99–119 (2015)
7. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 886–893 (2005)
8. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: European Conference on Computer Vision, pp. 428–441 (2006)
9. Dawn, D.D., Shaikh, S.H.: A comprehensive survey of human action recognition with spatio-temporal interest point (STIP) detector. Vis. Comput. **32**(3), 289–306 (2016)
10. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: a library for large linear classification. J. Mach. Learn. Res. **9**, 1871–1874 (2008)
11. Farnebäck, G.: Two-frame motion estimation based on polynomial expansion. In: Image Analysis, pp. 363–370 (2003)
12. Förstner, W., Moonen, B.: A metric for covariance matrices. In: Geodesy—The Challenge of the 3rd Millennium, pp. 299–309 (2003)
13. Guo, K., Ishwar, P., Konrad, J.: Action recognition from video using feature covariance matrices. IEEE Trans. Image Process. **22**(6), 2479–2494 (2013)
14. Hoai, M., Zisserman, A.: Improving human action recognition using score distribution and ranking. In: Asian Conference on Computer Vision, pp. 3–20 (2014)
15. Jain, M., Jegou, H., Bouthemy, P.: Better exploiting motion for better action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2555–2562 (2013)
16. Junejo, I.N., Junejo, K.N., Aghbari, Z.A.: Silhouette-based human action recognition using sax-shapes. Vis. Comput. **30**(3), 259–269 (2014)
17. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Annual Conference on Neural Information Processing Systems, pp. 1097–1105 (2012)
18. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: International Conference on Computer Vision, pp. 2556–2563 (2011)
19. Laptev, I.: On space-time interest points. Int. J. Comput. Vis. **64**(2), 107–123 (2005)
20. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
21. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2169–2178 (2006)
22. Li, Y., Ye, J., Wang, T., Huang, S.: Augmenting bag-of-words: a robust contextual representation of spatiotemporal interest points for action recognition. Vis. Comput. **31**(10), 1383–1394 (2015)
23. Ma, J., Zhao, J., Tian, J., Yuille, A.L., Tu, Z.: Robust point matching via vector field consensus. IEEE Trans. Image Process. **23**(4), 1706–1721 (2014)
24. Messing, R., Pal, C., Kautz, H.: Activity recognition using the velocity histories of tracked keypoints. In: International Conference on Computer Vision, pp. 104–111 (2009)
25. Niebles, J.C., Chen, C.W., Fei-Fei, L.: Modeling temporal structure of decomposable motion segments for activity classification. In: European Conference on Computer Vision, pp. 392–405 (2010)
26. Oneata, D., Verbeek, J., Schmid, C.: Action and event recognition with fisher vectors on a compact feature set. In: International Conference on Computer Vision, pp. 1817–1824 (2013)
27. Pang, Y., Yuan, Y., Li, X.: Gabor-based region covariance matrices for face recognition. IEEE Trans. Circuits Syst. Video Technol. **18**(7), 989–993 (2008)
28. Reddy, K.K., Shah, M.: Recognizing 50 human action categories of web videos. Mach. Vis. Appl. **24**(5), 971–981 (2013)
29. Sanchez, J., Perronnin, F., Mensink, T., Verbeek, J.: Image classification with the fisher vector: theory and practice. Int. J. Comput. Vis. **105**(3), 222–245 (2013)
30. Shi, J., Tomasi, C.: Good features to track. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 593–600 (1994)
31. Sun, J., Mu, Y., Yan, S., Cheong, L.F.: Activity recognition using dense long-duration trajectories. In: IEEE International Conference on Multimedia and Expo, pp. 322–327 (2010)
32. Truong, A., Boujut, H., Zaharia, T.: Laban descriptors for gesture recognition and emotional analysis. Vis. Comput. **32**(1), 83–98 (2016)
33. Tuzel, O., Porikli, F., Meer, P.: Region covariance: a fast descriptor for detection and classification. In: European Conference on Computer Vision, pp. 589–600 (2006)
34. Vishwakarma, S., Agrawal, A.: A survey on activity recognition and behavior understanding in video surveillance. Vis. Comput. **29**(10), 983–1009 (2013)
35. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Dense trajectories and motion boundary descriptors for action recognition. Int. J. Comput. Vis. **103**(1), 60–79 (2013)
36. Wang, H., Oneata, D., Verbeek, J., Schmid, C.: A robust and efficient video representation for action recognition. Int. J. Comput. Vis. **119**(3), 219–238 (2016)
37. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: International Conference on Computer Vision, pp. 3551–3558 (2013)
38. Wang, H., Yi, Y., Wu, J.: Human action recognition with trajectory based covariance descriptor in unconstrained videos. In: ACM international conference on Multimedia, pp. 1175–1178 (2015)
39. Willems, G., Tuytelaars, T., Gool, L.V.: An efficient dense and scale-invariant spatio-temporal interest point detector. In: European Conference on Computer Vision, pp. 650–663 (2008)
40. Wu, J., Hu, D., Chen, F.: Action recognition by hidden temporal models. Vis. Comput. **30**(12), 1395–1404 (2014)
41. Zhou, L., Lu, Z., Leung, H., Shang, L.: Spatial temporal pyramid matching using temporal sparse representation for human motion retrieval. Vis. Comput. **30**(6), 845–854 (2014)

Yun Yi is a Ph.D. candidate at the Department of Computer Science and Technology, Tongji University, Shanghai, China. He received the B.S. and M.S. degrees in Computer Science and Technology from Jiangxi Normal University, Jiangxi, China, in 2004 and 2007, respectively. His current research interests include computer vision and multimedia content analysis.

Hanli Wang received the B.E. and M.E. degrees in Electrical Engineering from Zhejiang University, Hangzhou, China, in 2001 and 2004, and the Ph.D. degree in Computer Science from City University of Hong Kong, Hong Kong, in 2007. From 2007 to 2008, he was a Research Fellow with the Department of Computer Science, City University of Hong Kong. From 2007 to 2008, he was also a Visiting Scholar with Stanford University, Palo Alto, CA. From 2008 to 2009, he was a Research Engineer with Precoad, Inc., Menlo Park, CA. From 2009 to 2010, he was an Alexander von Humboldt Research Fellow in University of Hagen, Hagen, Germany. In 2010, he joined the Department of Computer Science and Technology, Tongji University, Shanghai, China, as a Professor. His current research interests include multimedia data processing, computer vision and machine learning.