Research Article

INTERNATIONAL JOURNAL OF ADVANCED ROBOTIC SYSTEMS

International Journal of Advanced Robotic Systems
November-December 2017: 1–12

DOI: 10.1177/1729881417746114
journals.sagepub.com/home/arx
SAGE

A compact representation of human actions by sliding coordinate coding

Runwei Ding[1], Qianru Sun[2], Mengyuan Liu[3] and Hong Liu[1]

Abstract

Human action recognition remains challenging in realistic videos, where scale and viewpoint changes make the problem complicated. Many complex models have been developed to overcome these difficulties, while we explore using low-level features and typical classifiers to achieve the state-of-the-art performance. The baseline model of feature encoding for action recognition is bag-of-words model, which has shown high efficiency but ignores the arrangement of local features. Refined methods compensate for this problem by using a large number of co-occurrence descriptors or a concatenation of the local distributions in designed segments. In contrast, this article proposes to encode the relative position of visual words using a simple but very compact method called sliding coordinates coding (SCC). The SCC vector of each kind of word is only an eight-dimensional vector which is more compact than many of the spatial or spatial–temporal pooling methods in the literature. Our key observation is that the relative position is robust to the variations of video scale and view angle. Additionally, we design a temporal cutting scheme to define the margin of coding within video clips, since visual words far away from each other have little relationship. In experiments, four action data sets, including KTH, Rochester Activities, IXMAS, and UCF YouTube, are used for performance evaluation. Results show that our method achieves comparable or better performance than the state of the art, while using more compact and less complex models.

Keywords

Human action recognition, bag-of-words model, local feature

Date received: 31 May 2017; accepted: 27 October 2017

Topic: Vision Systems
Topic Editor: Antonio Fernandez-Caballero
Associate Editor: Wenhui Wang

Introduction

Human action recognition has shown its significance in a large amount of applications from video surveillance to human–machine interaction.[1] Nevertheless, to achieve high efficiency and robustness remains challenging due to the difficulties caused by variations, such as actor appearances, view angles, object scales, and so on. Some previous approaches aim at directly representing how human body moves using silhouette motion,[2] pose matching,[3] and shape templates.[4] The recent tendency is to summarize the overall video appearance, often utilizing spatial–temporal interest points (STIPs)[5–9] and local features, such as 3D gradients,[10] 3D Scale Invariant Feature Transform (SIFT),[11] and Histogram of Oriented Gradient (HOG)–Histograms of Oriented Optical Flow (HOF)[12] to build a global representation. These methods have shown great robustness against the scale variance and

[1] Key Laboratory of Machine Perception, Peking University, Shenzhen Graduate School, China
[2] Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany
[3] School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore

Corresponding author:
Hong Liu, Key Laboratory of Machine Perception, Peking University, Shenzhen Graduate School, China.
Email: hongliu@pku.edu.cn

background noises in realistic videos such as Hollywood movies.[12] One key consideration in previous works is known as the bag-of-words (BoW)[13] representation, where local features are quantized to a dictionary of codewords by the clustering algorithm, for example, *K*-means, then each video is summarized as a distribution histogram, which means only one vector is enough for the representation of the action video. Therefore, BoW has become rapidly popular and the modified BoW models have been constantly developed.[14,15,8,16]

The success of BoW model is due to the avoidance of preprocessing, such as background subtraction, body modeling, and motion estimation, and their robustness to slight scale variance and background noises. However, the key problem of BoW is the ignorance of any spatial–temporal arrangement such as motion trajectory, before–after arrangement, or relative layout of human–object. This problem could cause different classes of action with similar sub-actions to be wrongly recognized as one. In order to solve this problem, many alternatives attempt to model the relationships between neighbor or co-occurrence visual words. Related work could be roughly divided into three categories: (1) using space–time binning method in video segments formed globally at the level of video[12,17–19]; (2) using the word-centered way where multiple sub-bins or top ranked words are used to describe the local neighborhood of the centered word;[14,20] and (3) coding the co-occurrence (on a single frame or in the whole video) information between all pairwise words.[8,15,21,22]

Most of these methods concentrate on the local or pairwise neighborhood information. In contrast, this article explores the global view of visual words in the video space. We propose the sliding coordinates coding (SCC) to encode the relative arrangement around each kind of word (the first contribution). In each sliding step, we regard the video space as a 3D Cartesian coordinate system, assign a word as the origin, and compute the counting weight of every other word according to the temporal distance between this word and the origin. Then, the relative arrangement around the origin word is the accumulation of the counting weights of distinct-labeled words in each quadrant. Note that we use a Gaussian mixture model to decide which quadrant a word belongs to, getting a soft assignment of the quadrant label. After sliding the origin across all words in the video, each word gets an eight-dimensional vector representing the relative arrangement around it. Summing the vectors of the same-labeled words results in the final descriptor called SCC vector. Some important aspects of SCC are summarized:

1. The spatial–temporal information is embedded into the SCC vector. Therefore, it does not require any post-processing in the scenarios of action recognition.
2. SCC encodes the arrangement of visual words by relative information; hence, it has high robustness to different action situations with scene changes.
3. SCC representation is more compact than many other spatial or spatial–temporal pooling methods.
4. SCC works with sparse sampling or dense sampling for local feature extraction, and with hard assignment or soft assignment for clustering, as long as they can provide the set of visual words.
5. The dimensionality of SCC vector is just eight times of the size of visual dictionary, that is, the number of clusters.
6. SCC relies on the relative arrangement of visual words, which is entirely objective in each action class. Hence, it dose not involve any extra parameter, for example, the pyramid level[14] or the size of temporal bins.[19]

Recent action data sets mostly contain long-range human activities, for example, getting close and giving a punch to another person,[19] taking out the cup then pooling water then drinking the water off,[23] etc. In order to deal with such data sets, we propose to segment the long video into non-overlapped clips by using a temporal cutting scheme (the second contribution). This scheme involves the computation of error processing criterion and naturally relies on the density change of visual words during the observation of the activity. Temporal cutting scheme aims at packaging nearby words that belong to the same small action. Using such a cutting scheme enables the local SCC in one clip to record not only the relative arrangement but also the co-occurrence/nearby relationships of visual words. After computing the SCC vectors in clips, all vectors are summarized into a global SCC vector by summing the vector elements according to the label indices.

The rest of this article is organized as follows. In "Related work" section, we reviewed the relative human action recognition works. The computation of the SCC vector in an unsegmented video is presented in "Shifting coordinates coding" section. Temporal cutting scheme and global SCC are implemented in "Temporal cutting scheme" section. In "Experiments and discussions" section, we evaluate the performance of the proposed method in four public data sets. In "Conclusion and future work" section, we concluded the method and lined out future work.

Related works

Human action recognition has been studied extensively in the field of computer vision. Given the significant literature, we focus on the work which is most relevant to our proposal. The basis of the proposed method is the set of visual words derived from clustering. Different actions may be composed by the same appearance of visual words in different space–time arrangement. Therefore, the most recent progress is due to new local features and models capturing the spatial–temporal arrangement of local features or between human and objects. Related works are roughly divided into two categories in this article.

Space-time binning based methods

This kind of method captures high-level structure by the space–time binning on the whole set of local features. For example, the video is partitioned globally into short clips, then the histogram of prior frames is different from that of after frames.[24,17,19] Ryoo[19] proposed to use integral histograms to model how the word distribution changes over a sequence of segments. Such kind of temporal binning method has the problem of making the mode sensitive to the time shifts. Besides the temporal binning, Brendel and Todorovic[25] introduced a generative model describing the space–time structure as a weighted directed graph defined on the over-segmentation of a video. This method learns the structure of complex human activities, in terms of relevant subactivities and their spatiotemporal relations. However, this model lacks the ability to discriminatively disambiguate between repeated structures in videos and the actual parts of the activity. Chen and Grauman[26] proposed an irregular subgraph model in which local topological changes are allowed. Using fixed-size bins always assumes that the proper scale is known or can be estimated for all actions. However, such uniformity is not an inherent character of local features, especially when feature extraction methods are very different.

To deal with scale changes, grid representation in conjunction with the pyramid coding has become very popular.[27,12,28,29] The common idea is to cut the video into a sequence of increasingly finer grids and computes the BoW/Fisher vector in each grid, finally concatenating all vectors. However, the final vector always has high dimensionality depending not only on the cluster number but also on the bin size and the pyramid level. In contrast, the proposed method is much more compact, since the dimension of SCC vector is just the linear times of the cluster number. Moreover, the proposed method has an advantage that we do not have to choose additional parameters for binning and pyramiding, since we use the coordinate system to segment the video space automatically during the sliding process.

Neighborhood based methods

Several alternatives of BoW model attempt to capture the word arrangement by the statistics of neighborhood/co-occurrence pairwise words. Savarese et al.[15] introduced a spatial–temporal correlogram, which records the co-occurrences of words in local space–time regions. It used multiple scaled local regions as neighborhood scopes where the size of the region is manually specified. In order to encode the global structure of the action video, Liu et al.[8] computed the directional relationship between pairwise words in the range of the whole video.

Ryoo and Aggarwal[22] designed a spatio-temporal match kernel to measure the structural similarity between two feature sets extracted from different videos. This kernel explicitly compares the temporal relationships (e.g. before and during) as well as the spatial relationships (e.g. near and far) between local features in the 3D space. To avoid using space–time bins of local features, Kovashka and Grauman[14] proposed to use ranked nearby words for describing the neighborhood around the center word. Similarly to pyramid methods, they extracted the neighborhood information using multilevel vocabularies. However, the pyramid number should be decided in prior, and the feature dimensionality cannot be controlled by the method itself. In contrast, our proposed method offers a global view at the word arrangement by computing the relative position between distinct words, for which the dimensionality of the action descriptor is constrained by the size of vocabulary (i.e. the number of clusters). Moreover, our sliding manner ensures the final descriptor to represent the holistic word arrangement in the video space.

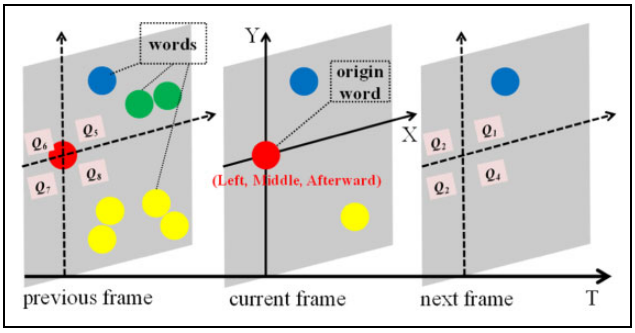

Figure 1. The intuition of relative position in a three-frame clip. With respect to other words, the origin word is located on the left horizontally, in the middle vertically, and occurs after most words in the time axis. "Q_1" indicates quadrant 1.

Shifting coordinates coding

For classifying images, Penatti et al.[30] extracted the arrangement of visual words in images using a method called word spatial arrangement (WSA). WSA divides the image plane into four quadrants every time it meets a word then counts the number of features in each quadrant. It is simple but effective for computing the relative arrangement of image parts. Inspired by this work, we propose the SCC to encode the relative arrangement of spatial–temporal words in videos. The intuition of relative position in a three-frame clip is shown in Figure 1. Nevertheless, SCC is not a simple 3D extension of WSA. We have three refined points: (1) we use the soft assignment to decide which quadrant the word belongs to, and we show our difference with WSA in Figure 2; (2) since the action video is time-related and the relative relationship between visual words occurring too far from each other has little meaning, we give a counting weight to each word based on the temporal distance between the word and the current origin, while Penatti's method counts "1" each time it meets a word; and (3) for computing SCC vector in long-range human activities, we propose a temporal cutting scheme as a complementary.

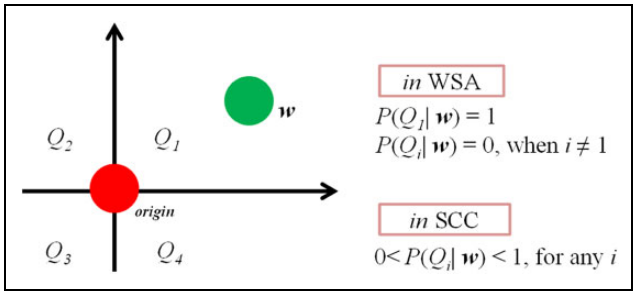

Figure 2. The difference between SCC and the WSA. *P* is the conditional probability. SCC: sliding coordinates coding; WSA: word spatial arrangement.

Prior assumption

The basis of the proposed method is the set of visual words. Hence, it is assumed that local features have been extracted and labeled through preprocessing of feature extraction and clustering. Let us assume that the video $\mathcal{V}$ contains a set of N visual words $\boldsymbol{w}_i$ at absolute pixel position $\boldsymbol{p}_i = (x_i, y_i, t_i)$ with the feature vector $\boldsymbol{f}_i$, and each word has been assigned a discrete label $c_i \in [1, 2, ..., K]$ from a visual vocabulary containing K entries. The location of the visual words can be determined using STIPs/cuboids/dense trajectories.[6,5,31] Local feature vector can be obtained by 3D-SIFT/3D-gradient/HoG–HoF[11,10,32,31] and so on. Once the set of local features is extracted, they are quantized into a set of discrete labels using a vocabulary generated by the clustering algorithm.

In the following algorithms, we assume that all visual words are detected in a strict order of the observation sequence, that is, $\forall i, j$, if $i < j$, then $t_i \leq t_j$. Thus, each video clip V is initially described as a sequence of word tuples

$$\mathcal{V} = \{\boldsymbol{w}_i\}_{i=1}^{N} = \{(\boldsymbol{p}_i, \sigma_i^s, \sigma_i^t, \boldsymbol{f}_i, c_i)\}_{i=1}^{N} \quad (1)$$

where σ^s and σ^t are the spatial and temporal scale generated during the feature extraction.[12] The absolute position in the video space is represented by $\boldsymbol{p}_i = (x_i, y_i, t_i)$, but the feature vector $\boldsymbol{f}_i$ indicates the location in the feature space.

Coding algorithm

In the video space, SCC first assigns one word as the origin, divides the clip into M quadrants by X-Y-T axis, for example, it is the 3D Cartesian coordinate system when $M = 8$, then counts the occurrences of distinct-labeled words.

Assuming that word $\boldsymbol{w}_i$ is the current origin and the quadrant index is $m \in \{1, 2, ..., M\}$, the SCC vector element of $\boldsymbol{w}_i$ is computed as follows

$$\mathrm{scc}(i, m) = \sum_{\substack{c_j \neq c_i \\ \boldsymbol{w}_j \in \mathcal{Q}_{i,m}}} h(\boldsymbol{w}_i, \boldsymbol{w}_j) \quad (2)$$

where $\mathcal{Q}_{i,m}$ represents the set of words located in the mth quadrant when $\boldsymbol{w}_i$ is the origin. The function $h(\boldsymbol{w}_i, \boldsymbol{w}_j)$ is the counting weight of $\boldsymbol{w}_j$ computed by equation (5).

In the algorithm of WSA,[30] each word is assigned one quadrant index according to its position in the 2D image plane. In contrast, we use a mixture model to compute a soft assignment of the quadrant to make the proposed method more robust to the rotations of view angle. For a word $\boldsymbol{w}_j$, we have its absolute angle $\phi_{j,i}$ in the coordinate system where $\boldsymbol{w}_j$ is the origin. In order to compute the probability of a word belonging to a certain quadrant, the following mixture model is formed

$$P(\phi_{j,i}) = \sum_{m=1}^{M} \psi_m f(\phi_{j,i} | \mu_m, \kappa_m) \quad (3)$$

where ψ_m, $m = 1, ..., M$, are the mixture weights, satisfying the constraint that $\sum_{m=1}^{M} \psi_i = 1$, and $f(\phi_{j,i} | \mu_m, \kappa_m)$, $m = 1, 2, ..., M$, are the von Mises components. Each component density is a von Mises probability density function of the form as follows

$$f(\phi_{j,i} | \mu_m, \kappa_m) = \frac{e^{\kappa_m \cos(x - \mu_m)}}{2\pi I_0(\kappa_m)} \quad (4)$$

where I_0 is the modified Bessel function of order 0, μ_m is a measure of location (the distribution is clustered around μ_m), and κ_m is a measure of concentration/dispersion. In the formulation above, the parameter set $\{\psi_m, \mu_m, \kappa_m\}$, $m = 1, 2, ..., M$, are estimated by the maximum likelihood algorithm, for which angle samples are computed by random pairs of visual words.

In another aspect, the counting weight always equals to 1 in the algorithm of WSA.[30] In the case of video, an intuition is that the words far away from the origin word should decrease their weights in counting. For images, it is important to measure the spatial nearness for recognizing one object from the whole image which contains backgrounds and other objects. In our case, the nearness is measured just on the T-axis, since it is assumed that local features are extracted without backgrounds. The formula to compute the counting weight of $\boldsymbol{w}_j$ when $\boldsymbol{w}_i$ is the origin word is given as

$$h(\boldsymbol{w}_i, \boldsymbol{w}_j) = \frac{1}{\sqrt{2\pi} \cdot a\sigma_i^t} \exp\left(-\frac{(t_i - t_j)^2}{2 \cdot (a\sigma_i^t)^2}\right) \quad (5)$$

where σ_i^t is the temporal scale obtained in local feature extraction, and a is the tolerance multiplier chosen in experiments.

SCC assigns the origin from one word to another in a sliding manner until all words are traversed. Algorithm 1 gives the details of this coding process locally from the words on the ath frame to the words on the bth frame. Since the relative position involves a pair of words, the "opp" index of the mth quadrant in Line 8 is used to avoid the

Algorithm 1: Local shifting coordinate coding

Input: the set of visual words $\mathcal{V}^{a,b} = \{\boldsymbol{w}_i\}_{i=1}^{n}$ in the local video from the a-th frame to the b-th frame, the mixture weights ψ_m, the measure of location μ_m and the measure of concentration/dispersion κ_m , $m = 1, 2, ..., M$

Output: local SCC vector $\boldsymbol{scc}^{a,b}$

1 **for** $i \leftarrow 1$ *to* $(n-1)$ **do**
2 Construct the M-quadrant coordinate system using $\boldsymbol{w}_i$ as the origin word.
3 **for** $j \leftarrow (i+1)$ *to* n **do**
4 Compute the angle $\phi_{j,i}$ of $\boldsymbol{w}_j$ in this coordinate system.
5 Calculate the posterior $f(\phi_{j,i}|\mu_m, \kappa_m)$ and $f(\phi_{j,i}|\mu_{m_{opp}}, \kappa_{m_{opp}})$ for $m \in \{1, 2, ..., M\}$.
6 **for** $m \leftarrow 1$ *to* 4 **do**
7 $\boldsymbol{scc}^{a,b}(c_i, m) \leftarrow \boldsymbol{scc}^{a,b}(c_i, m) + f(\phi_{j,i}|\mu_m, \kappa_m) \cdot h(\boldsymbol{w}_i, \boldsymbol{w}_j)$;
8 $\boldsymbol{scc}^{a,b}(c_j, m_{opp}) \leftarrow \boldsymbol{scc}^{a,b}(c_j, m_{opp}) + f(\phi_{j,i}|\mu_{m_{opp}}, \kappa_{m_{opp}}) \cdot h(\boldsymbol{w}_j, \boldsymbol{w}_i)$;
9 **end**
10 **end**
11 **end**
12 **return** $\boldsymbol{scc}^{a,b}$;

duplicate traverses when the opposite word is the origin. Formula "$j \leftarrow (i+1)$ to N" in Line 3 is under the assumption that the set of visual words has the time order, that is, t_i is non-monotone increasing with i.

Temporal cutting scheme

Long-range human activities are always composed by multiple sub-actions, where each sub-action means there is no interruption of the motion appearance. We propose to compute the local SCC inside the independent sub-action clip. Then, it is a problem that how to cut the video into such clips. Big cutting size is not enough for segment sub-actions, while small size always results in over segmentation. Basically, a video can be cut into equal sizes, but the parameter involved such as the clip size is fixed and cannot adapt to all action classes. A good cutting method should be robust to different video scales and action speeds. In this section, we propose a temporal cutting scheme based on the temporal changes of the feature appearance. The key idea is that a continuous sub-action guarantees the continuous appearance of motion features. The end of a sub-action must generate some null video frames which have no feature, before the next sub-action occurs. A sequence of continuous null frames could be used as a gap between the before and after clips.

The realistic situation is that local feature extraction always involves random errors. Frames having noisy features should also be classified as null frames, for which we propose to estimate the feature sparsity.

Sparsity estimation

We utilize the information statistic method to set the threshold for the sparsity degree of being a null frame. In Algorithm 2, we define a sparsity factor according to the error processing criterion which states that if the measurement error is threee times larger than the standard error, the current data point will be regarded as a singular data (deleted and substituted with arithmetic average value). While the observation continues, we update the temporary variables μ and σ, which represent the average number of features and the standard error, respectively. Then, the new input frame, which has equal or fewer number of features than the current *sparseFactor*, will be regarded as a null frame. Original features on the null frame seem to be noises and will be removed from the feature set of the video.

Algorithm 2: Sparsity filter

Input: the set of visual words $\mathcal{V} = \{\boldsymbol{w}_i\}_{i=1}^{N}$ in the video

Output: the filtered visual word set $\mathcal{V}'$

1 $pointNum \leftarrow 1$, $frameNum \leftarrow 0$, total number of words $sum \leftarrow 1$, average number of words $\mu \leftarrow 0$, standard error $\sigma \leftarrow 0$;
2 **for** $i \leftarrow 1$ *to* $(N-1)$ **do**
3 **if** $t_{i+1} == t_i$ **then**
4 $pointNum \leftarrow pointNum + 1$;
5 *continue*;
6 **end**
7 $frameNum \leftarrow frameNum + 1$;
8 $sum \leftarrow sum + pointNum$;
9 $\mu \leftarrow sum/frameNum$;
10 $\sigma \leftarrow \sqrt{(\frac{frameNum-1}{frameNum}) * \sigma + (pointNum - \mu)^2}$;
11 $sparseFactor \leftarrow \lceil \mu - 3\sigma \rceil$;
12 **if** $pointNum > sparseFactor$ **then**
13 $\mathcal{V}' \leftarrow \mathcal{V}' \bigcup \{\boldsymbol{w}_{(i-pointNum+1)}, \boldsymbol{w}_{(i-pointNum+2)}, ..., \boldsymbol{w}_i\}$
14 **end**
15 **end**
16 **return** $\mathcal{V}'$;

Cutting and coding

It is intuitive to use a sequence of continuous null frames as a gap between two clips. Illustrations of finding gaps are shown in Figure 3. For each video clip, we compute the local SCC as shown in Algorithm 1. Then, we sum up local SCC vectors to construct the final descriptor of the whole video. Coding details are given in Algorithm 3. The minimum gap size ε indicates that a gap should contain at least ε null frames. Here, $\min\{\varepsilon\} = 1$, and $\varepsilon = +\infty$ indicates there is no gap, or no cut in the video. The complexity of Algorithm 3 is $O(n^2 \cdot M + n)$. Since the value of M is fixed as an empirical value, the complexity is simplified as $O(n^2)$, where the number of words determines the time cost. In other words, if the number of words is limited to be no larger than a constant value, our Algorithm 3 can be efficiently implemented.

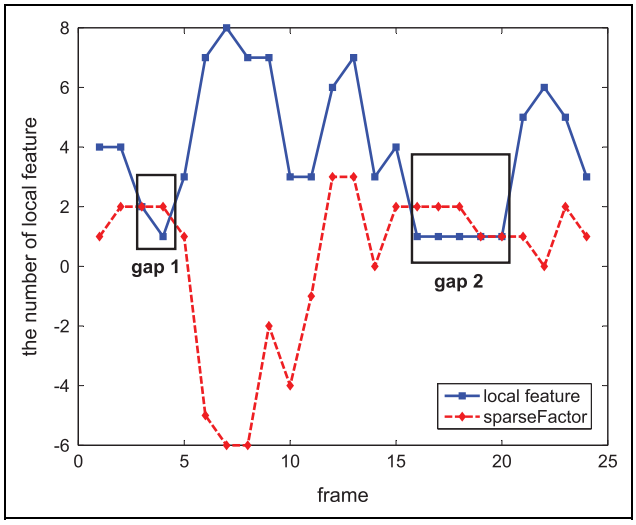

Figure 3. A sequence of null frames is used as a gap. The size of gap-1 is 2 and the size of gap-2 is 5.

Algorithm 3: Video cutting and global SCC coding

Input: the filtered set of visual words $\mathcal{V}' = \{\boldsymbol{w}_i\}_{i=1}^{N'}$, with $N' \leq N$, the minimum gap size ε.
Output: the global SCC vector ***gscc***

```
cursor ← t_1 % a temporary variable to record the start of each video clip.
for i ← 1 to N do
    if (t_{i+1} − t_i) < ε then
        continue;
    end
    gscc ← gscc + lscc^{cursor,t_i};
    cursor ← t_{i+1};
end
return gscc;
```

Experiments and discussions

The experiments are conducted as follows. The first step is to extract local feature descriptors from action sequences. Then, *K*-means clustering is employed to generate visual words, and our SCC method is used to encode the relative position of visual words, generating compact action representations. At last, we utilize a nonlinear Support Vector Machine (SVM) for classification.

In this section, we first describe the action data sets we used and basic settings for feature extraction. Then, we present the comparisons with baseline and state-of-the-art methods. Finally, we present the influences brought by the parameters in the proposed method.

Data sets

To evaluate the effectiveness and robustness of the proposed method for human action recognition, experiments are conducted in four public data sets: KTH, Rochester Activities, IXMAS, and UCF YouTube. These data sets are very different from each other. KTH[6] involves difficulties, such as shadows and scale changes, whereas Rochester Activities[23] contains the videos that are long-range living activities. IXMAS[33] is filmed at multiple view angles, and UCF YouTube[34] includes low-resolution videos collected from websites.

KTH is an important milestone in the literature of human action recognition, and it has been widely tested, for example, in the works of Kovashka and Grauman,[14] Savarese et al.,[15] Messing et al.,[23] and Bregonzio et al.[35] The KTH database contains 2391 sequences and all sequences were taken over homogeneous backgrounds with a static camera with 25 frames per second (fps) frame rate. The spatial resolution is 160 × 120 pixels. It contains six types of human actions (walking, jogging, running, boxing, hand waving, and hand clapping) performed by 25 people in four different scenarios: outdoors, outdoors with scale variation, outdoors with varying clothes, and indoors.

Rochester Activities is a high-resolution video data set of activities of daily living. Video was taken at 1280 × 720 pixel resolution, at 30 fps. It contains 10 classes of daily activities: answering a phone, chopping a banana, dialing a phone, drinking water, eating banana, eating snacks, looking up a phone number in a book, peeling a banana, eating food with silverware, and writing on a white board. Each activity is a long-range sample composed by a sequence of sub-actions. Videos are divided into five sets based on five different actors, and each set contains three repetitions of 10 classes. This data set has been tested in the works of Messing et al.,[23] Escorcia and Niebles,[36] Glaser and Zelnik-Manor,[37] and Liu et al.[8]

IXMAS is used to evaluate the robustness of the proposed method in videos of multiple view angles. It contains 1980 videos including 13 daily-life motions, each of which is performed three times by 12 individuals. The video sequences were taken at 390 × 291 pixel resolution, at 23 fps, under original views of five cameras. The actors arbitrarily choose position and orientation. We use the first four angles at which the whole body can be seen. Note that the definition of action classes strictly accords to the newest article.[38]

UCF YouTube includes personal action videos collected from Youtube websites. It contains 1168 video sequences in total. This data set has the following properties: (1) steady cameras and shaky cameras; (2) clutter backgrounds; (3) variations in action speed and scale; (4) varying view point; and (5) low resolution. There are 11 categories: basketball shooting, volleyball spiking, trampoline jumping, soccer juggling, horse-back riding, cycling, diving, swinging, golf swinging, tennis swinging, and walking (with a dog). There exist some difficulties that first four actions are easily confused with "jumping," and all the "swing" actions share some common motions.

Basic settings

The proposed method is transparent to the selection of local feature detector and descriptor. However, different features

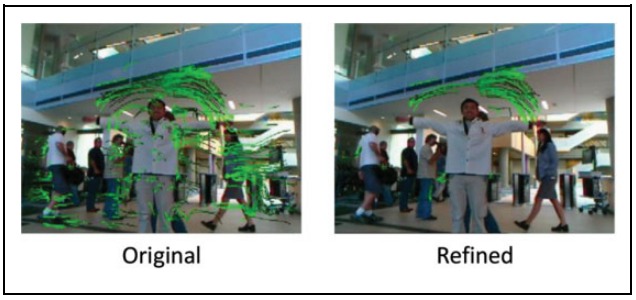

Figure 4. Comparing the original dense trajectory with our refined version. A man is waving his hands in a cluttered scene. It is clear that our refined detector removes the background noises while reserving the trajectories of "waving" which is the target.

are used for different data sets for a fair comparison with other works. Laptev's detector[39] is applied to KTH, Rochester Activities, and IXMAS data sets, to generate sparse STIPs. For the most challenging UCF YouTube, we use a refined detector of the dense trajectory.[31] As shown in Figure 4, the refined detector removes very straight trajectories which are usually detected on passing objects. Both sparse and dense features are described by the 162-dimensional HoG–HoF, following previous work.[14,12]

K-means clustering is applied to generate visual words, with 800 clusters for KTH, 500 clusters for Rochester Activities, 4000 clusters for IXMAS, and 4000 clusters for UCF YouTube. These cluster numbers are referred to or close to the settings proposed by Savarese et al.,[15] Glaser and Zelnik-Manor,[37] Wang et al.,[31] and Liu et al.[8] Recognition is conducted uniformly using a nonlinear SVM with a χ^2 kernel.[40] Leave-one-out-cross-validation is adopted for the training testing. In IXMAS, we use the videos of the first four angles where the whole body can be observed. In each round, the videos at one viewpoint are used for testing, and the rest for training. The recognition accuracy is the ratio of the number of correct predictions and the total number of tests. For performance measure, we report the average accuracy over all classes as the final result. Since random initializations are involved in algorithms, for example, *K*-means, all reported accuracies are averaged over 10 running results.

It should be noted that there are two key parameters in SCC: the tolerance multiplier a and the minimum gap width ε. In subsection "Baseline evaluations" and "Comparison with the state of the arts", these parameters have default values $a = 0.5$ and $\varepsilon = 4$ for KTH, $a = 1$ and $\varepsilon = 4$ for Rochester Activities, $a = 4$ and $\varepsilon = +\infty$ for IXMAS (no video cut), and $a = 4$ and $\varepsilon = 2$ in UCF YouTube. The sensitivity of the proposed method to the parameters above is tested in subsection "Comparison with the state of the arts".

Baseline evaluations

To evaluate the superiority of the proposed method, comparisons are implemented with: BoW model using HoG–HoF features[12]; BoW model using dense trajectories[31]; the 3D version of WSA,[30] and the component testing in the proposed method using: SCC without soft assignment in the quadrant indexing (i.e. SCC no soft assignment); SCC with all counting weight equals to 1 (i.e. SCC no weighting); SCC without temporal cutting scheme (i.e. SCC no cut); and SCC in conjunction with temporal cutting scheme (i.e. Ours).

Corresponding performances are shown in Table 1. The proposed method outperforms the baseline BoW model[31] by 6.14% in the challenging data set UCF YouTube. Moreover, the proposed method gives surprising improvements of 20.66–32.37% in the multiple view IXMAS, which indicates that the relative arrangement information encoded by SCC is highly robust to the view angle changes. Also, we attribute much to the soft assignment of quadrant indexing, since the improvement in angle 2 drops off from 20.98% to 5.50% if "SCC no soft assignment" method is adopted. It is noted that the highest results of IXMAS are obtained in no cut videos. The reason is that IXMAS actions are mostly very simple and short motions such as seeing the watch once with a hand up and down. Temporal segmentation on such simple actions may result in over-segmentation and makes nearby features have no contribution to the SCC vector of each other.

Additionally, we present the results per action class of UCF YouTube, see Table 2. The proposed method gives

Table 1. Performance comparisons with baselines.

Method	KTH	Rochester Activities	IXMAS				UCF YouTube
			angle1	angle2	angle3	angle4	
BoW using sparse STIPs[12]	0.918	0.85	0.5196	0.5234	0.5196	0.5177	0.6137
BoW using dense trajectories[31]	0.942	0.9133	0.6692	0.6711	0.6597	0.6635	0.842
3D version of WSA[30]	0.8833	0.8725	0.6309	0.5781	0.5003	0.6433	0.6967
SCC no soft assignment	0.925	0.8567	0.7096	0.5784	0.6690	0.6412	0.7725
SCC no weighting	0.9350	0.9333	0.7901	0.6567	0.6773	0.7570	0.8513
SCC no cut	0.9413	0.9533	0.8433	0.7332	0.7262	0.8086	0.7967
Ours	0.9883	0.9967	—	—	—	—	0.9034

BoW: bag-of-words; SCC: sliding coordinates coding; STIP: spatial–temporal interest point; WSA: word spatial arrangement.

Table 2. Accuracy per action class for UCF YouTube, comparing with the results of baseline methods.[31,12]

	SCC cut	denseTraj.[31]	STIPs[12]
basketballShooting	0.6930	0.430	0.3070
cycling	0.9917	0.917	0.6505
diving	0.9785	0.990	0.6920
golfSwinging	0.9934	0.970	0.7498
horsebackRiding	0.9028	0.850	0.6500
soccerJuggling	0.8211	0.760	0.6733
swing	0.9358	0.880	0.5767
tennisSwing	0.8050	0.710	0.4432
trampolineJumping	0.9352	0.940	0.6509
volleyballSpiking	0.9896	0.950	0.7044
walking	0.8917	0.870	0.6526
Accuracy	0.9034	0.842	0.6137

SCC: sliding coordinates coding; STIP: spatial–temporal interest point.

best results for 9 out of total 11 classes when compared with baseline methods.[31,12]

Comparison with the state of the arts

Table 3 compares our results with the state-of-the-art methods. For different data sets, the state of the arts are different mainly because that some papers only done experiments on specific data sets. Most of them are unsupervised methods using local motion features, except Zhu and Shao[45] and Escorcia and Niebles.[36] In KTH, we obtain 98.83% which is slightly higher than 98.2% in the works of Sadanand and Corso,[43] where the experiments were implemented with linear SVM classifiers. Sadanand and Corso[43] proposed a template model called Action Bank using many high-level action detectors to construct the final representation. Our advantage over this model is that SCC encodes the arrangement/structure of the action video simply by using low-level features. Usually, low-level feature can be extracted very effectively and is flexible to different kinds of video scenario. From this respect, it can be concluded that the proposed method has better applicability than the Action Bank.

Each video in Rochester Activities contains a sequence of sub-actions. SCC with temporal cutting scheme achieves the best accuracy in Rochester, demonstrating that the proposed method can describe the spatial–temporal arrangement of human actions more efficiently.

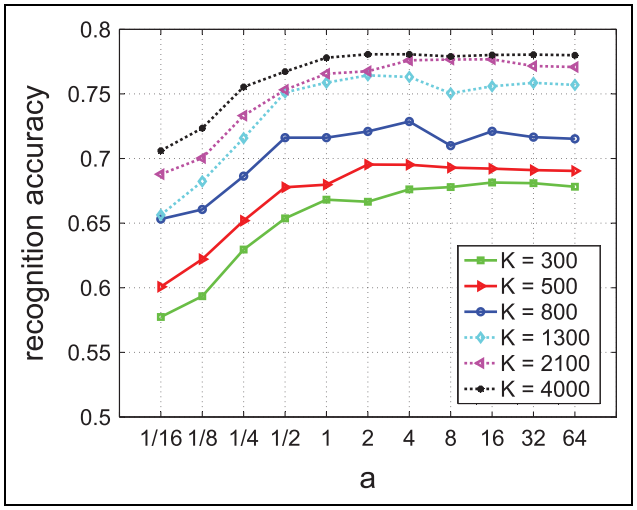

Figure 5. Recognition accuracies correspond to different values of K and a in IXMAS.

For recognizing the multiple view videos in IXMAS, our best rate is 77.78% and is lower than the highest record in the study by Ciptadi et al.[38] Ciptadi et al.[38] designed the specific algorithm to solve the problem of cross-view action recognition; however, the proposed method is relatively general to be applies to different scenarios of action recognition.

Zhu and Shao[45] utilized a semi-supervised learning method and obtained the highest accuracy 91.11% in UCF YouTube. Though the proposed method is unsupervised, it still obtains a very comparable accuracy 90.34% in this data set.

Parameter evaluation

In IXMAS, the minimum gap size ε equals $+\infty$, indicating that videos are used without any cut. For parameter evaluation, K is tested with $K \in \{300, 500, .., 4000\}$, and the tolerance multiplier a is in $\{2^{-3}, 2^{-2}, ..., 2^{4}\}$. Corresponding recognition accuracies are given in Figure 5. As we can see, when the size of vocabulary K ranges from 300 to 4000,

Table 3. Performance comparisons with the state-of-the-art methods.

KTH		Rochester Activities		IXMAS		UCF YouTube	
Bregonzio et al.[35]	0.9317	Messing et al.[23]	0.89	Yan et al.[41]	0.78	Liu et al.[34]	0.712
Kovashka and Grauman[14]	0.9453	Glaser and Zelnik-Manor[37]	0.911	Reddy et al.[42]	0.7260	Wang et al.[31]	0.842
Sadanand and Corso[43]	0.982	Escorcia and Niebles[36]	0.98	Glaser and Zelnik-Manor[37]	0.716	Oneata et al.[29]	0.89
Beaudry et al.[44]	0.9532	Liu et al.[8]	0.9133	Ciptadi et al.[38]	0.82	Zhu and Shao[45] (semi-sup.)	0.9111
Shao et al.[46]	0.975	Sun et al.[47]	0.984	—	—	Shao et al.[46]	0.876
Ours	0.9883	Ours	0.9967	Ours	0.7778	Ours	0.9034

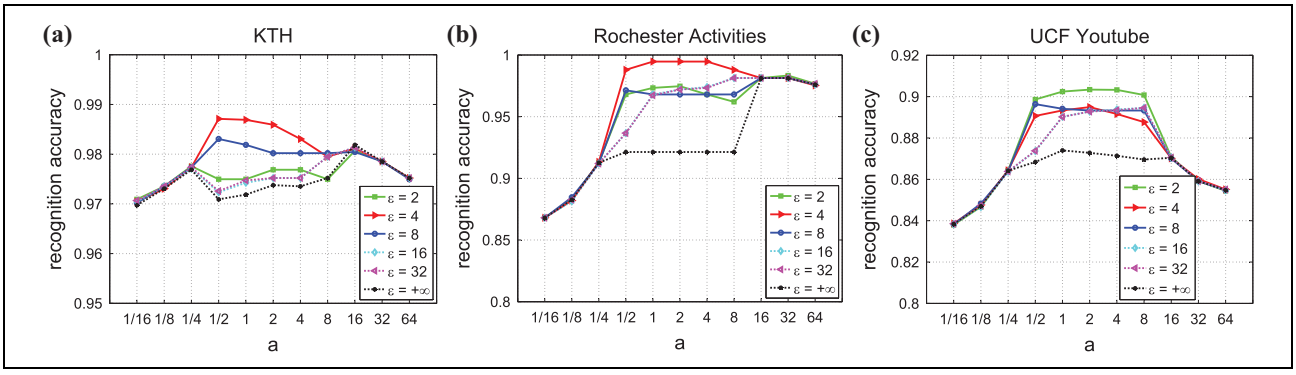

Figure 6. Recognition accuracies correspond to different values of ε and a in KTH, Rochester, and UCF YouTube.

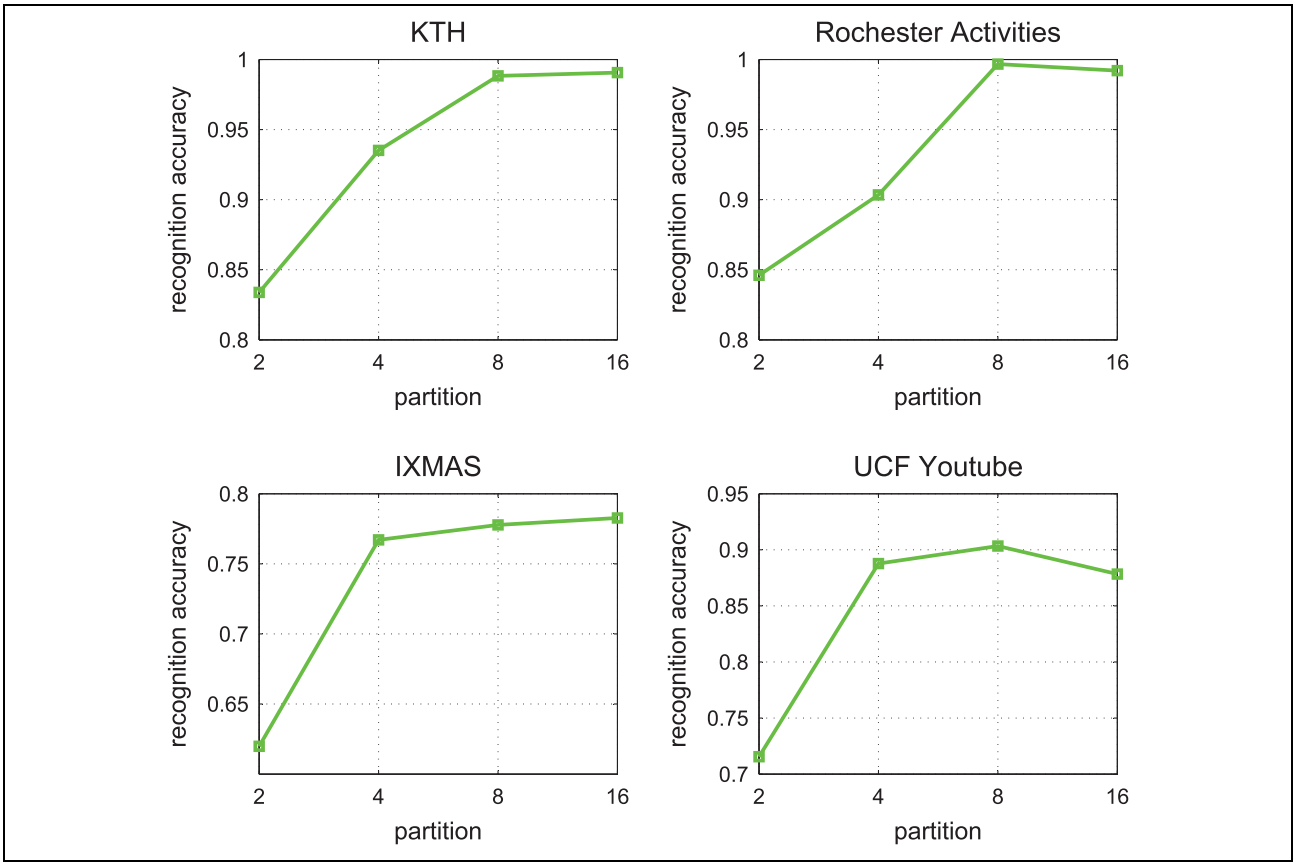

Figure 7. Recognition accuracies using 2, 4, 8, and 16 partitions of the video space.

using bigger size vocabulary always brings better result. The highest accuracy 77.78% is achieved in $K = 4000$ and $a = 4$ (or $a = 2$). However, it can also be noticed that when $a >= 4$, increasing K from 2100 to 4000 does not bring much improvement of recognition accuracy, so we do not still increase the vocabulary size. Note that Paper 76 constrains the size of vocabulary referred to the setting of related work.

In KTH, Rochester Activities, and UCF YouTube, we use the default vocabulary sizes which are referred to the settings of related works. Then, we compare the results using the minimum gap size $\varepsilon \in \{1, 2, ..., 2^5\}$ and the tolerance multiplier $a \in \{2^{-3}, 2^{-2}, ..., 2^4\}$. Figure 6(a) to (c) presents the results of three data sets. It can be observed that most curves have a tendency of increasing from $a = 1/16$ to 1 and decreasing from $a \geq 16$.

The algorithm of SCC is based on dividing the space–time domain into eight quadrants, that is, a standard 3D Cartesian coordinate system. Intuitive idea may rise that different partition ways can also be used in the proposed method. Therefore, we vary the size of partition and compare the performance of the proposed method using eight quadrants with the performances using 2, 4, and 16 partitions, see in Figure 7. Specifically, the 2-quadrant

system is just the partition in time axis, that is, the relative positions include "before" and "after" only. The 4-quadrant system discards the time-axis partition of the standard 3D Cartesian coordinate system, while the 16-quadrant system reserves the time axis and doubles the spatial division into eight partitions. As it is shown in Figure 7, more defined partitions always get better performances for KTH and IXMAS data sets. However, this phenomenon is not the same for recognizing long-range activities in Rochester Activities and the clustered videos in UCF YouTube. The reason might be over-segmentation of the video space brings about too much align deviation during the calculation of SCC vector distances, especially when the vectors are obtained from complicated or noisy videos.

Conclusion and future work

Instead of working toward complex models, this article proposes a simple but efficient way to code the arrangement of local visual words into a compact action descriptor. Basic idea is that using a local SCC to encode the relative positions of visual words in a statistic manner. The descriptor of the whole video is obtained by applying global SCC on all video clips generated by temporal cutting scheme. These steps mainly depend on the neighborhood distribution around each visual word, hence they are free from tedious parameter selection. The proposed method is based on the relative position, so it is robust to the variations of video scale and view angle. And by using a temporal cutting scheme, the proposed method can deal with the data sets which contain long-range human activities. Another advantage of the proposed method lies in that the SCC vector is much more compact than that of co-occurrence or pyramid coding method since its dimensionality is only several times of the cluster number. The performance of SCC vector has been evaluated in four public data sets containing of very different difficulties. We find that using the arrangement of basic visual words and typical SVM classifiers can also achieve the state-of-the-art performance than designing complex action representation. More importantly, the framework of the proposed method is available for using different kinds of local features and clustering algorithms if only they can provide the set of visual words.

The proposed SCC can efficiently encode local spatial–temporal information among short video clips. In future work, we plan to encode long-term spatial–temporal information by combining SCC and temporal pyramid structure.[12] To tackle with speed variations, energy-based temporal pyramid structure[48] can also be used to divide long-term video into multi-scale short clips. The temporal information among clips can be captured by decision level fusion of hierarchical SCC representations.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by National Natural Science Foundation of China (NSFC grant nos 61340046, 61673030, and U1613209), Natural Science Foundation of Guangdong Province (grant no 2015A030311034), Scientific Research Project of Guangdong Province (grant no. 2015B010919004), Science and Technology Innovation Commission of Shenzhen Municipality (grant no. JCYJ20170306164738129), Specialized Research Fund for the Strategic and Prospective Industrial Development of Shenzhen City (grant no. ZLZBCXLJZI20160729020003), Scientific Research Project of Shenzhen City (grant no. JCYJ20170306164738129), and Shenzhen Key Laboratory for Intelligent Multimedia and Virtual Reality (grant no. ZDSYS201703031405467).

References

1. Liu M, Liu H, Sun Q, et al. Salient pairwise spatiotemporal interest points for real-time activity recognition. *CAAI Trans Intell Technol* 2016; 1(1): 14–29.
2. Bobick AF and Davis JW. The recognition of human movement using temporal templates. *IEEE Trans Pattern Analy Mach Intell* 2001; 23(3): 257–267.
3. Lv F and Nevatia R. Single view human action recognition using key pose matching and Viterbi path searching. In: *IEEE conference on computer vision and pattern recognition*, 17–22 June 2007, Minneapolis, MN, USA, pp. 1–8. IEEE.
4. Blank M, Gorelick L, Shechtman E, et al. Actions as space-time shapes. In: *10th IEEE international conference on computer vision*, 17–21 October 2005, Beijing, China, Vol. 2, pp. 1395–1402. IEEE.
5. Dollár P, Rabaud V, Cottrell G, et al. Behavior recognition via sparse spatio-temporal features. In: *2nd joint IEEE international workshop on visual surveillance and performance evaluation of tracking and surveillance*, 15–16 October 2005, Beijing, China, pp. 65–72. IEEE.
6. Schuldt C, Laptev I, and Caputo B. Recognizing human actions: a local svm approach. In: *17th IEEE international conference on pattern recognition*, 26–26 August 2004, Cambridge, UK, Vol. 3, pp. 32–36. IEEE.
7. Oikonomopoulos A, Patras I, and Pantic M. Spatiotemporal salient points for visual recognition of human actions. *IEEE Trans Syst Man Cybern B Cybern* 2005; 36(3): 710–719.
8. Liu M, Liu H, and Sun Q. Action classification by exploring directional co-occurrence of weighted STIPs. In: *IEEE international conference on image processing*, 27–30 October 2014, Paris, France, pp. 1460–1464. IEEE.
9. Liu H, Liu M, and Sun Q. Learning directional co-occurrence for human action classification. In *IEEE international conference on acoustics, speech and signal processing*, 4–9 May 2014, Florence, Italy, pp. 1235–1239. IEEE.

10. Klaser A, Marszalek M, and Schmid C. A spatio-temporal descriptor based on 3D-gradients. In: *British machine vision conference*, 1–4 September 2008, Leeds, England, pp. 275–271. British Machine Vision Association.
11. Scovanner P, Ali S, and Shah M. A 3-dimensional SIFT descriptor and its application to action recognition. In: *15th ACM international conference on multimedia*, 24–29 September 2007, Augsburg, Germany, pp. 357–360. ACM.
12. Laptev I, Marszalek M, Schmid C, et al. Learning realistic human actions from movies. In: *IEEE conference on computer vision and pattern recognition*, 23–28 June 2008, Anchorage, AK, USA, pp. 1–8. IEEE.
13. Peng X, Wang L, Wang X, et al. Bag of visual words and fusion methods for action recognition: comprehensive study and good practice. *Comput Vis Image Underst* 2016; 150: 109–125.
14. Kovashka A and Grauman K. Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. In: *IEEE conference on computer vision and pattern recognition*, 13–18 June 2010, San Francisco, CA, USA, pp. 2046–2053. IEEE.
15. Savarese S, DelPozo A, Niebles JC, et al. Spatial-temporal correlatons for unsupervised action classification. In: *IEEE workshop on motion and video computing*, 8–9 January 2008, Copper Mountain, CO, USA, pp. 1–8. IEEE.
16. Wang Y, Tran D, and Liao Z. Learning hierarchical poselets for human parsing. In: *IEEE conference on computer vision and pattern recognition*, 20–25 June 2011, Colorado Springs, CO, USA, pp. 1705–1712. IEEE.
17. Marszalek M, Laptev I, and Schmid C. Actions in context. In: *IEEE conference on computer vision and pattern recognition*, 20–25 June 2009, Miami, FL, USA, pp. 2929–2936. IEEE.
18. Han D, Bo L, and Sminchisescu C. Selection and context for action recognition. In: *IEEE 12th international conference on computer vision*, 29 September–2 October 2009, Kyoto, Japan, pp. 1933–1940. IEEE.
19. Ryoo MS. Human activity prediction: early recognition of ongoing activities from streaming videos. In: *IEEE international conference on computer vision*, 6–13 November 2011, Barcelona, Spain, pp. 1036–1043. IEEE.
20. Gilbert A, Illingworth J, and Bowden R. Fast realistic multi-action recognition using mined dense spatio-temporal features. In: *2009 IEEE 12th international conference on computer vision*, 29 September–2 October 2009, Kyoto, Japan, pp. 925–931. IEEE.
21. Morioka N and Satoh S. Compact correlation coding for visual object categorization. In: *IEEE international conference on computer vision*, 6–13 November 2011, Barcelona, Spain, pp. 1639–1646. IEEE.
22. Ryoo MS and Aggarwal JK. Spatio-temporal relationship match: video structure comparison for recognition of complex human activities. In: *IEEE international conference on computer vision*, 29 September–2 October 2009, Kyoto, Japan, pp. 1593–1600. IEEE.
23. Messing R, Pal C, and Kautz H. Activity recognition using the velocity histories of tracked keypoints. In: *12th IEEE international conference on computer vision*, 29 September–2 October 2009, Kyoto, Japan, pp. 104–111. IEEE.
24. Sun J, Wu X, Yan S, et al. Hierarchical spatio-temporal context modeling for action recognition. In: *IEEE conference on computer vision and pattern recognition*, 20–25 June 2009, Miami, FL, USA, pp. 2004–2011. IEEE.
25. Brendel W and Todorovic S. Learning spatiotemporal graphs of human activities. In: *IEEE international conference on computer vision*, 6–13 November 2011, Barcelona, Spain, pp. 778–785. IEEE.
26. Chen CY and Grauman K. Efficient activity detection with max-subgraph search. In: *IEEE conference on computer vision and pattern recognition*, 16–21 June 2012, Providence, RI, USA, pp. 1274–1281. IEEE.
27. Lazebnik S, Schmid C, and Ponce J. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: *IEEE computer society conference on computer vision and pattern recognition*, 17–22 June 2006, New York, NY, USA, Vol. 2, pp. 2169–2178. IEEE.
28. Harada T, Ushiku Y, Yamashita Y, et al. Discriminative spatial pyramid. In: *IEEE conference on computer vision and pattern recognition*, 20–25 June 2011, Colorado Springs, CO, USA, pp. 1617–1624. IEEE.
29. Oneata D, Verbeek J, and Schmid C. Action and event recognition with fisher vectors on a compact feature set. In: *IEEE international conference on computer vision*, 1–8 December 2013, Sydney, NSW, Australia, pp. 1817–1824. IEEE.
30. Penatti O, Valle E, and da S Torres R. Encoding spatial arrangement of visual words. *Prog Patt Recog Image Analy Comput Vis Appl* 2011; 7042: 240–247.
31. Wang H, Kläser A, Schmid C, et al. Action recognition by dense trajectories. In: *IEEE conference on computer vision and pattern recognition*, 20–25 June 2011, Colorado Springs, CO, USA, pp. 3169–3176. IEEE.
32. Wang H, Ullah MM, Klaser A, et al. Evaluation of local spatiotemporal features for action recognition. In: *British machine vision conference*, 7–10 September 2009, London, UK, pp. 124.1–124.11. BMVA Press.
33. Weinland D, Ronfard R, and Boyer E. Free viewpoint action recognition using motion history volumes. *Comput Vis Image Underst* 2006; 104(2): 249–257.
34. Liu J, Luo J, and Shah M. Recognizing realistic actions from videos in the wild. In: *IEEE conference on computer vision and pattern recognition*, 20–25 June 2009, Miami, FL, USA, pp. 1996–2003. IEEE.
35. Bregonzio M, Gong S, and Xiang T. Recognising action as clouds of space-time interest points. In: *IEEE conference on computer vision and pattern recognition*, 20–25 June 2009, Miami, FL, USA, pp. 1948–1955. IEEE.
36. Escorcia V and Niebles J. Spatio-temporal human-object interactions for action recognition in videos. In: *IEEE international conference on computer vision workshops*, 2–8 December 2013, Sydney, NSW, Australia, pp. 508–514.
37. Glaser T and Zelnik-Manor L. Incorporating temporal context in bag-of-words models. In: *IEEE international*

conference on computer vision workshops, 6–13 November 2011, Barcelona, Spain, pp. 1562–1569. IEEE.
38. Ciptadi A, Goodwin MS, and Rehg JM. Movement pattern histogram for action recognition and retrieval. In: *European conference on computer vision*, 6–12 September 2014, Zurich, Switzerland, pp. 695–710. Springer.
39. Laptev I and Lindeberg T. On space-time interest points. On space-time interest points. *Int J Comput Vis* 2005; 64(2-3): 107–123.
40. Vedaldi A and Fulkerson B. Vlfeat: An open and portable library of computer vision algorithms. In: *18th International Conference on Multimedea*, 25–29 October 2010, pp. 1469–1472. ACM.
41. Yan P, Khan SM, and Shah M. Learning 4D action feature models for arbitrary view action recognition. In: *IEEE conference on computer vision and pattern recognition*, 23–28 June 2008, Anchorage, AK, USA, pp. 1–7. IEEE.
42. Reddy KK, Liu J, and Shah M. Incremental action recognition using feature-tree. In: *12th IEEE international conference on computer vision*, 29 September–2 October 2009, Kyoto, Japan, pp. 1010–1017. IEEE.
43. Sadanand S and Corso JJ. Action bank: a high-level representation of activity in video. In: *IEEE conference on computer vision and pattern recognition*, 16–21 June 2012, Providence, RI, USA, pp. 1234–1241. IEEE.
44. Beaudry C, Péteri R, and Mascarilla L. Action recognition in videos using frequency analysis of critical point trajectories. In: *IEEE international conference on image processing*, 27–30 October 2014, Paris, France, pp. 1445–1449. IEEE.
45. Zhu F and Shao L. Weakly-supervised cross-domain dictionary learning for visual recognition. *Int J Comput Vis* 2014; 109(1–2): 42–59.
46. Shao L, Liu L, and Yu M. Kernelized multiview projection for robust action recognition. *Int J Comput Vis* 2016; 118(2): 115–129.
47. Sun Q, Liu H, Liu M, et al. Human activity prediction by mapping grouplets to recurrent self-organizing map. *Neurocomputing* 2016; 177: 427–440.
48. Liu M, Liu H, and Chen C. 3D action recognition using multiscale energy-based global ternary image. *IEEE Trans Circuits Syst Video Technol* 2017. DOI: 10.1109/TCSVT.2017.2655521.