RESEARCH

Motion superpixels for temporal video classification

Novanto Yudistira[1*†] and Takio Kurita[2]

[*]Correspondence:
cbasemaster@gmail.com
[1]Graduate School of Information Engineering, Hiroshima University, Japan
Full list of author information is available at the end of the article
[†]Equal contributor

Abstract

Superpixels are a representation of still images as pixel grids because of their more meaningful information compared with atomic pixels. However, their usefulness for video classification has been given little attention. In this paper, rather than using spatial RGB values as low-level features, we use optical flows mapped into hue-saturation-value (HSV) space to capture rich motion features over time. We introduce motion superpixels, which are superpixels generated from flow fields. After mapping flow fields into HSV space, independent superpixels are formed by iteration of seeded regions. Every grid of a motion superpixel is tracked over time using nearest neighbors in the histogram of flow (HOF) for consecutive flow fields. To define the temporal representation, the evolution of three features within the superpixel region, namely the HOF, histogram of gradients (HOG), and the center of superpixel mass are used as descriptors. The bag of features algorithm is used to quantify final features, and generalized histogram-kernel support vector machines are used as learning algorithms. We evaluate the proposed superpixel tracking on first-person videos and action sports videos.

Keywords: motion; superpixel; temporal features; video classification

Introduction

Video classification, especially as it relates to human activity and animal behavior, has become an important area of research. Many methods based on the visual representation of RGB frames have been proposed, but exploration of superpixels in motion space has still not been discussed much. Videos contain scenes that change over time and that are sometimes distracted because of camera motion or occlusions. Therefore, the first step for most automatic video classification methods is to locate and extract the position of an object of interest in the scene. The traditional approach is to combine handcrafted features processed from low-level features such as pixels. Recent methods based on either lower-level pixels or higher levels of motion attempt to understand and classify videos in which superpixels may effectively define the localities of scenes or objects. It would be interesting to determine whether localities can be obtained by oversegmenting optical flows rather than spatial features to extract full motion-based features. Motion superpixels follow flow properties in which homogeneous directions and magnitudes gather into a single cluster under various user-defined thresholds. To make use of time-varying motions, it would be beneficial to track a sequence of consistent superpixels after motion superpixels are formed on the flow fields. Quantification methods such as the bag of features (BOF) algorithm can be used to quantify sparse features in spatio-temporal domains before feeding them into a learning algorithm such as a support vector machine (SVM).

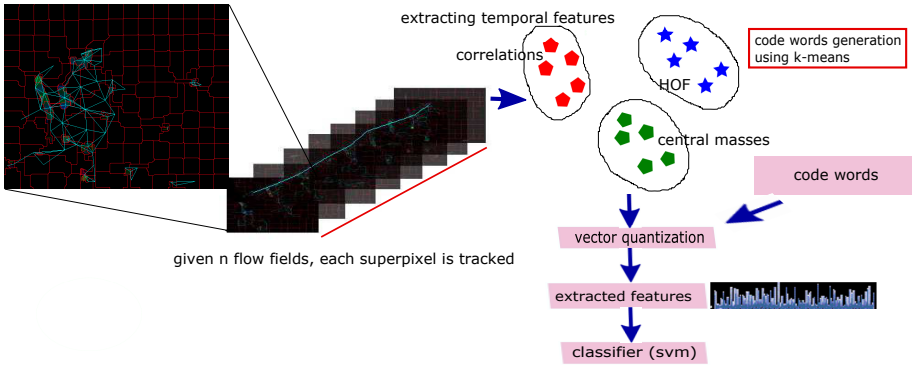


Figure 1 Given n flow fields, the proposed method constructs flows that form a sequence over time. Three features are extracted from each flow field for the tracked superpixels. These collections of temporal features form evolutionary features that are quantified by BOF and fed into the learning algorithm for classification.

Figure 1 shows the entire process, including tracking, extracting temporal features, quantification and feeding into the learning algorithm.

Related works

Superpixels have been used as features for classification, segmentation, and tracking tasks. In image classification or segmentation, they have been found to be useful for remote sensing [1], object classification [2], motion words for videos of hyperspectral images [3], medical imaging [4] and medical segmentation [5]. In the era of video analysis, segmentation has dominated superpixel-based feature approaches, such as video segmentation using superpixel flows [6], learning to segment moving objects [7], perceptual organization [8], and temporal superpixels using tracking figure-ground segments [9]. Superpixels are also useful for saliency detection [10]. However, in the case of video classification, there is still little research compared with video segmentation application, such as motion words [11], feature co-occurrence for activity classification [12], temporal superpixels using a generative approach [13], and spatial superpixels using motion descriptors for action recognition [14]. Our approach to video classification uses the superpixels extracted via energy-driven sampling (SEEDS) algorithm proposed by Bergh et al. as a pixel-wise algorithm for fast superpixel generation with reliable quality of representation based on hill-climbing optimization [5]. Video SEEDS, which improves on the original SEEDS algorithm, was introduced as a method for tracking superpixel continuity over time using temporal windows [15]. Energy-driven superpixels produced by growing segmented regions under iteration have been shown to be computationally efficient and robust, with low computational cost and promising results. Our approach uses SEEDS for superpixel generation and tracks superpixels over time. We do not consider the video SEEDS algorithm because our goal is not segmentation but extraction of atomic feature vectors, which are simple, fast and limited by the predefined vector dimension. While motion features are usually represented by optical flows, there are promising extension to such approaches, including the use of higher-order local autocorrelation [16] based on the analysis of autocorrelation between spatially and temporally neighboring pixels. Kim et al. [17] has proposed a higher-order

correlation of superpixels that merges disjoint homogeneous superpixels into more meaningful objects. Similarly, our tracker and descriptor also considers temporal correlations over a superpixel's neighborhood. Ke et al. [18] proposed localization and recognition of actions using the correlation of spatio-temporally segmented videos tested on the KTH dataset. Their results are not significant in that there is still room to improve on the correlations, especially in flow fields rather than RGB space. The viability of such approaches for this paper hinges on whether it is possible to adapt correlations in flow fields and the advantages of doing this in recognition exercises. In RGB space, Dong et al. [14] proposed action recognition using superpixel features extracted from RGB space, and attached motion features without tracking. We compare our method to RGB-based superpixels. Further research on superpixels extracted from RGB space considered temporal tracking to improve robustness. Using learned features, such as from convolutional neural networks (CNNs) [19], motion information such as optical flows helps to improve recognition of actions. However, this requires an abundance of training data compared to handcrafted features because of overfitting. To extract handcrafted features from motion, spatial and temporal dynamics of actions cannot be neglected in feature construction. Unfortunately, multi-scale analysis using differential operators is prone to losing low-frequency information. Recently developed and well-founded methods such as spatio-temporal interest points (STIP) [20], dense trajectories [21], and scale-invariant feature transforms (SIFT) [22] are prone to bias on coarse scales. More meaningful region-based tracking rather than using points or cuboids with a predefined size can give exclusive motion boundaries and extract features under different scales of regions. We evaluate our proposed method on first-person videos and sports videos because these two types of videos represent recent topics of interest in video classification. Little research has been conducted on first-person videos, examples being papers by Ryoo [23] and Iwashita [24]. Superpixel classification has been applied to optic cup segmentation [4] in glaucoma detection, using mean intensity, centered statistics and location as features for segmentation. This approach gives better results than previous method for glaucoma detection. Our features are similar, although they are richer because we use the histogram of flow (HOF) and consider neighborhood correlation over time. Another use of superpixels for change detection is multi-dimensional change analysis [25] for a static image that changes over time. Our problems are more challenging in that scenes and people in video datasets are dynamic over time. Temporal superpixels for video representation have been proposed [13] in generative probabilistic models for temporally consistent superpixels over time. Our method is different because we oversegment directly into optical flows and track consistent centers of mass of superpixels sequentially.

1 Motion superpixel

Motion superpixels are derived from motion space, in this case, from optical flows. The SEEDS algorithm creates segmented flows that are iterated using color distributions and boundary terms. Figure 2 shows the difference between spatial SEEDS, which oversegments RGB space, and motion SEEDS, which oversegments flow space. Motion superpixels are constructed where motion arises and remain in default form when there is no motion or very little motion, which depends on a threshold.

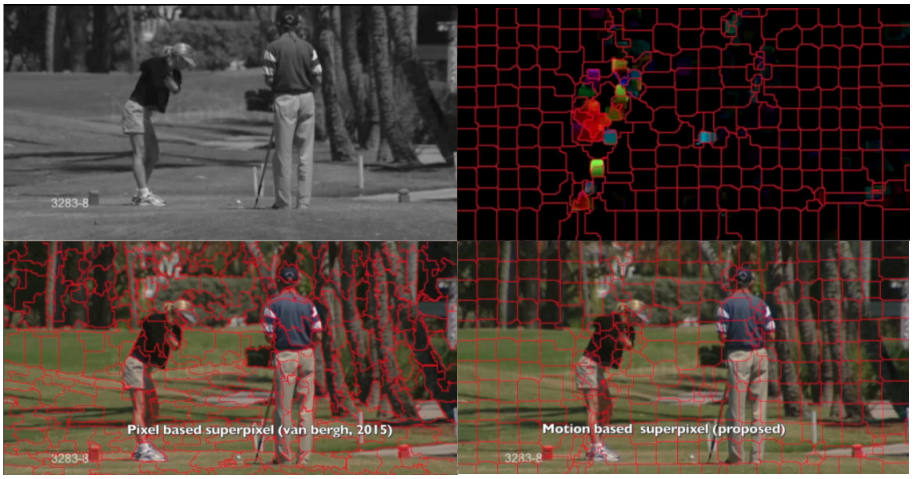


Figure 2 Top row: original image; motion superpixels on flow field. Bottom row: original superpixels from SEEDS; motion superpixels mapped to the original image (MPEG-4, 21 MB).

If a motion superpixel is mapped to the original RGB space, then the superpixel will react if there is a moving object. We can filter out stationary superpixels by selecting superpixels for which the average from the HOF is greater than zero. We start from the definition of flow field which motion vectors consis of of x direction, y direction, and its angular :

$$\begin{pmatrix} x-direction \\ y-direction \\ \arctan2(F_x, F_y) + \phi \end{pmatrix} = \begin{pmatrix} F_x \\ F_y \\ \theta \end{pmatrix} \tag{1}$$

To be able to oversegment, motion space is transformed into color space without losing its valuable informations. Motion vector is then transformed into HSV space using:

$$f^* = \begin{pmatrix} H \\ S \\ V \end{pmatrix} = \begin{pmatrix} 90\theta\phi \\ 255 \\ \sqrt{{F_x}^2 + {F_y}^2} \end{pmatrix} \tag{2}$$

Given N as the number of flow color vector f^*, and K as the number of constructed superpixels, we can map flow partition into superpixel such that:

$$s : (1, ..., N) \rightarrow (1, ..., K) \tag{3}$$

Once the K partitions are generated, we can represent flow color vectors f^* inside superpixel k as:

$$A_k = f^* : s(f^*) = k \tag{4}$$

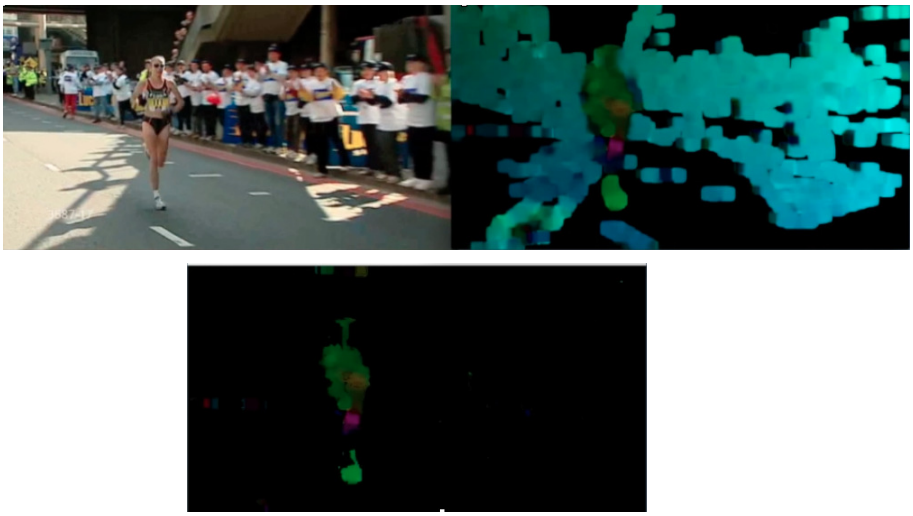


Figure 3 Clockwise from top left: original video of runner with camera following runner; flow field of optical flow which highlights the runner and the noise from camera motion; flow field after motion compensation to highlight only the runner's actions. The motions of an actor or an object of interest are considered to be local motions.

A_k contains flows inside superpixel k. The superpixels are then optimized iteratively using energy function maximization E until s^* such that:

$$s^* = argmax E(s) \tag{5}$$

2 Motion compensation

Motions or flows within a frame are considered to be motion features. It is important to apply motion compensation because of camera motion. Video captured using handheld devices have a high degree of freedom and often includes camera motion that distracts from actual motions. Moreover, it may be difficult to differentiate between actions of interest and the background from action videos taken in the field. To solve this problem, it is possible to use affine transformation and random sampling consensus. We use a similar consensus approach with a rigid transformation that estimates the affine transformation and removes those parts from the flow field. We use the rigid affine estimation in equation 6, in which i is a point inside the pixel region of the current frame X_t and the next frame X_{t+1}, and find a 2×2 matrix $A*$ and a 2×1 vector $b*$ such that:

$$(A*, b*) = argmin \sum_i ||X_{t+1}[i] - AX_t[i]^T - b||^2. \tag{6}$$

We find a transformation matrix from the reference frame to the next frame that represents the rigid transformation. In real action recognition, small camera movements can greatly impact flow alteration. By assuming that camera motion is rigid, the affine flow field can be removed from the flow field. After A and b have

been found, the rigid prediction of X_t can be determined. If we assume that the rigid prediction is $\hat{X}_{t+1}$, flow field space is X_{t+1}, and cleared flow field is $X_{t+1}^{compensated}$ then the following holds:

$$X_{t+1}^{compensated} = X_{t+1} - \hat{X}_{t+1} \tag{7}$$

Figure 3 shows an original image of a runner, where a moving camera is following the runner. Only the salient movements are required to be processed during image recognition. In human motion analysis, articulated motion is considered to be salient as it creates varied and dynamic flows that cannot be modeled using affine transformations.

3 Motion superpixel tracks

While dense trajectories track a reference point over time based on flows, we use superpixels of flows, which contain more meaningful information and track reference superpixels over time based on the nearest moving point. Similarly, just as dense trajectories are tracked over multiple scales, motion superpixels are tracked in varying sizes. Grids for the initial superpixels are seeded in every frame and are iterated until convergence. Every convergent superpixel is then tracked with its corresponding superpixels based on the value nearest to its center of mass. Based on experiments, it is found that using 250 superpixel seeds per flow field is enough to give baseline results. We used five different values for the number of superpixel seeds (1, 4, 16, 64 and 256 initial seeds). Temporal information is prominent in activity recognition tasks. To treat motion superpixels over time, we used the nearest neighbor to find the superpixel corresponding to a specific super-pixel for times t to $t+1$ between two consecutive flow fields. It is similar to mean shift algorithm in the sense of correspondence search between one frame to the next frame. Mean shift can use histogram or kernel density function to track object based on pixel distribution. It, however, works effectively on grid sampling. For region that have been clustered into superpixels, we can select correspondence by making use of avalaible superpixels. To build the corresponding network between superpixels, we defined the center of mass for a superpixel region as sum for all possible x and y inside superpixel A as follows:

$$m_{ij} = \sum_{x,y} A(x,y) x^j y^i, \tag{8}$$

$$\bar{x} = \frac{m_{10}}{m_{00}}, \bar{y} = \frac{m_{01}}{m_{00}}. \tag{9}$$

Equation 8 shows the spatial moments m_{ij} up to the first order for a polygon or superpixel. $A(x,y)$ is intensity value of flow color within superpixel A. While $(i,j) \in \{(1,0),(0,0),(0,1),(0,0)\}$ and $\{x,y\}$ represents points on the border of the superpixel, equation 9 gives the center of mass $(\bar{x},\bar{y})$ of a superpixel. Note that

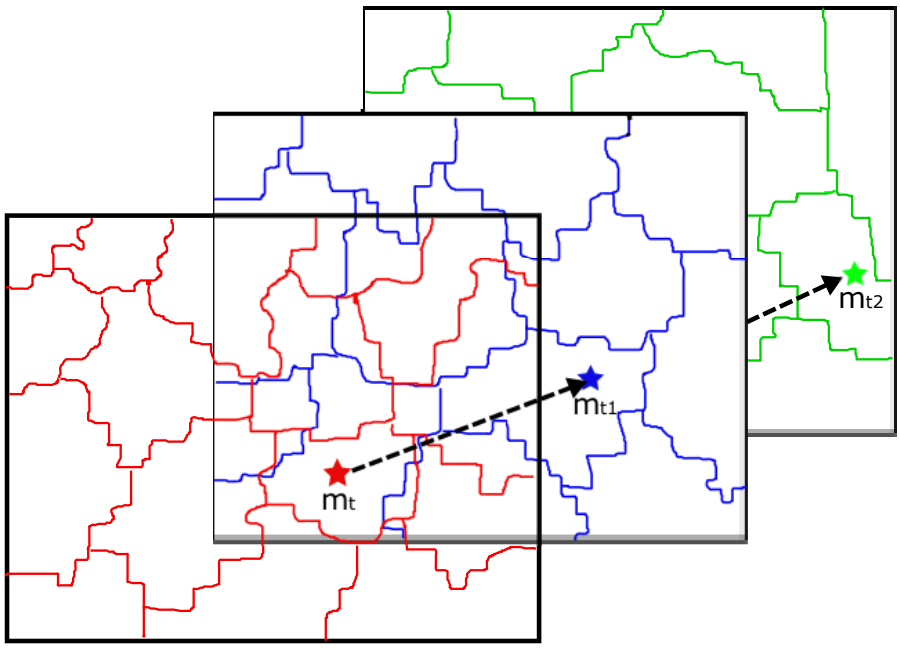


Figure 4 Tracking is done by following the nearest center of mass from a flow field frame m_t to the next frame m_{t1}.

each superpixel contains information about a set of motions. Each superpixel at time $t+1$ may have moved based on motions contained inside the superpixels at time t. Therefore, the new central moments at time $t+1$ ($m_s(t+1)$) are the central moments at time t ($m_s(t)$) summed according to the average flow color vectors's location of $(\bar{v}_x, \bar{v}_y)$ in the superpixels at time t as described in equation 10:

$$m_s(t+1) = m_s(t) + (\bar{v}_x, \bar{v}_y) \tag{10}$$

Under the assumption that motion superpixels in consecutive flow fields will appear in a position nearest to its reference superpixel whether its shape changes or not, tracking is done by finding the nearest central moment point at each iteration. This is similar to tracking by following the path of the center of mass of superpixels over time as in Figure 4. Along with several flow fields, this yields a time series of feature evolution. For each superpixel s, its corresponding superpixel s_{t+1} is selected based on the minimum distance between the central moments of:

$$s_{t+1} = \underset{s \in S}{argmin} ||(m_s(t) - m_s(t+1)||. \tag{11}$$

After computing the optical flows, seeds of the superpixels are constructed and each superpixel is tracked based on its center of mass. A collection of centers of mass for the given time interval forms a sequence of tracked motion superpixels $(m_t, m_{t+1}, m_{t+2}, ...)$. The tracking is restricted to a time interval because longer

track increase the probability of drift or bias from the initial point. In anticipation of this problem, we predefine the number of flow fields N in a sequence. If the next superpixel contains a flow field with all zero values or with no motion, then the sequence is terminated. If the length of the sequence is less than N, then the track is not saved as a feature. Conversely, if the track contains N flow fields and all superpixels contain motions, then the track is saved as a feature vector. In practice, a track length of $L = 10$ flow fields is used.

In general, superpixels with no motion are represented in HSV color space as zero values or in black. This means that there is no presence of motion in that superpixel region, or that the motion is removed because of motion compensation. We use the termination criterion of the absence of motion to prevent non-motion selection. As with dense trajectories [21], dense optical flows are more robust than sparse flows, and thus dense Farneback optical flows [26] are chosen as the base flows for the entire process.

Local motion segmentation to form superpixels contains information about motion flows. A sequence tracked over time will produce a sequence of motion flows that can be described as a rich motion pattern. Given a number of flow fields N, the sequence of moments over time $M = (\Delta m_t, \Delta m_{t+1}, ...\Delta m_N)$ has the displaced central moments $\Delta m_t = (m_{t+1} - m_t) = (x_{t+1} - x_t, y_{t+1} - y_t)$. The resulting vector is normalized by the sum of the magnitudes of the displacement vectors in the central moments sequence:

$$S^* = \frac{(\Delta m_t, \Delta m_{t+1}, ...\Delta m_N)}{\sum_{i=t}^{t+N-1} ||\Delta m_i||}. \tag{12}$$

The vector S^* is the final normalized vector in the superpixel tracking representation. In this research, we only consider $N = 10$ because varying N does not significantly influence the accuracy of the results. However, this does not improve the results in practice. Thus, using a fixed number of flow fields produces a reliable final superpixel track vector.

4 Two-way motion superpixel tracks (bi-tracks)

Tracking along temporal flows requires path selection for dense superpixels. By using the nearest centers of mass over time, there is a possibility that a track will lose the most salient path for describing temporal evolution. This situation often happens when there are many centers of masses within a small distance in the next flow field. Selecting the nearest center of mass is not enough in this case, so we consider multiple selections to ensure that the generated paths are adequate for temporal representation.

As in Figure 5, we select the two nearest centers of mass at the first iteration of tracking. The first (m_{t+1}) and second (m'_{t+1}) centers of mass are selected from the second flow field, and only the nearest center of mass is selected in subsequent iterations. This produces two paths, which we call a bi-track, for every superpixel in the flow field. It can be helpful in the case of motion compensation, as in the UCF Sports dataset, to not remove camera motion, as this can confuse track generation. There is a possibility of adding more paths by using multiple selections of centers of mass to enrich feature sampling.

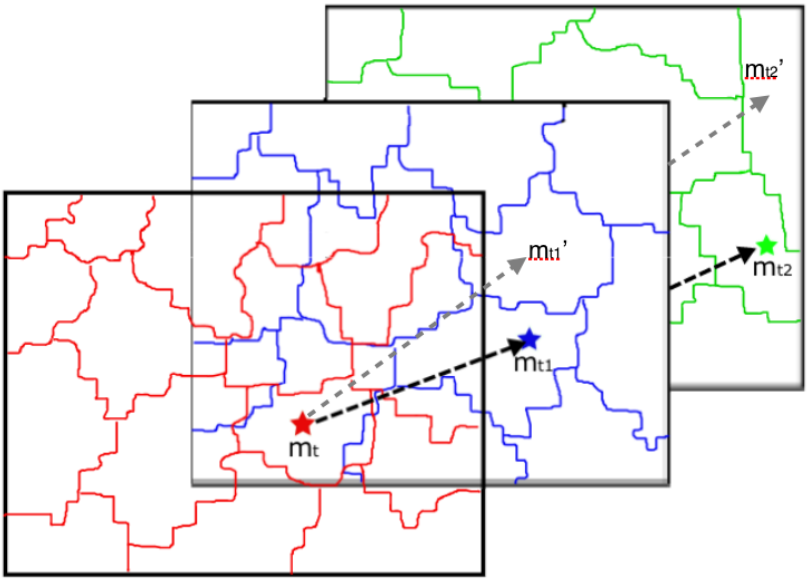


Figure 5 Bi-track produces two paths starting from m_t. central moment of m_t chooses both m_{t+1} and m'_{t+1} at first track.

5 Feature descriptors

We define four features for evolutionary representations, namely the HOF, position (center of mass), and the histogram of gradient (HOG). Every motion superpixel must form a region which consists of a boundary along with the flow field contained inside it.

5.1 Histogram of Flows

The HOF extracted from a flow field consists of several bins with direction and magnitude quantifiers. One of possibility to gather information about distribution is discretizing by using histogram. HF is a collection of bins which is equally spaced discretization of angular θ between 0 to 2π. In this histogram, Denote that h_f is histogram of flows. Within constructed superpixel k, angular informations of θ are quantized into bins its magnitude m is weight to count each angular into bin such that:

$$h_f(z) = \sum_{f \in F_k} m\Omega((f) \in HF_z) \tag{13}$$

Where Ω is 1 if flow vector f falls in bin z. We use 10 bins as default number of bins because based on several experiments within 8-20 bins, there is no significant change in accuracy.

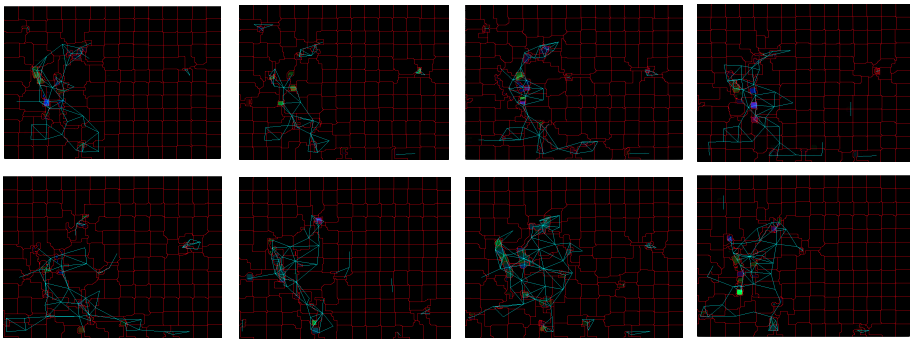


Figure 6 Temporal evolution of motion superpixels in sequence starting from the upper left corner to the upper right corner and from the lower left corner to the lower right corner.

5.2 Histogram of Gradients

In spatial space, frame has information about the change of shape. It is represented by spatial gradient. Suppose HG is a collection of bins consists of equally spaced discretization of gradient angular α between 0 to 90. Denote that h_g is histogram of gradients. Within constructed superpixel k, square 4×4 is used to sample pixels and quantize its gradients into bins such that:

$$h_g(l) = \sum_{i \in I_k} \delta(I(i) \in HG_l) \tag{14}$$

Where δ is 1 if flow vector f falls in bin l. We use 10 bins as default number of bins because same as HOF, based on experiments using 8-20 bins, there is no significant change in accuracy.

As in Figure 6, the evolution of a superpixel's neighborhood can be captured by the evolution of correlations in the relevant region. The communal representation of neighbors changes over time, which gives dynamic information about moving objects or parts of objects.

5.3 Local Bag of Features & Classification

The BOF is constructed from data gathered by temporal sampling. The sampling step for ten flow fields is used as the subsequent feature vector. We use a class-specific dictionary formed from three iterations of k-means clustering. Fast k-means clustering [27] is used, which is provably computationally cheap even without using a GPU. The number of clusters in the representation of features is usually determined heuristically by trial-and-error within a given range for the number of clusters. If the number of clusters is too small, then the representation may be shallow, whereas if the number of clusters is too large then the representation becomes nearly flat and is hard to generalize. We consider using separate bags for different descriptors and different superpixel sizes. Previous research has shown that separating bags is advantageous, especially for different coding scales [28].

For classification, we use an SVM with a generalized histogram kernel, which has been shown to be robust in terms of quantification-based features such as histograms

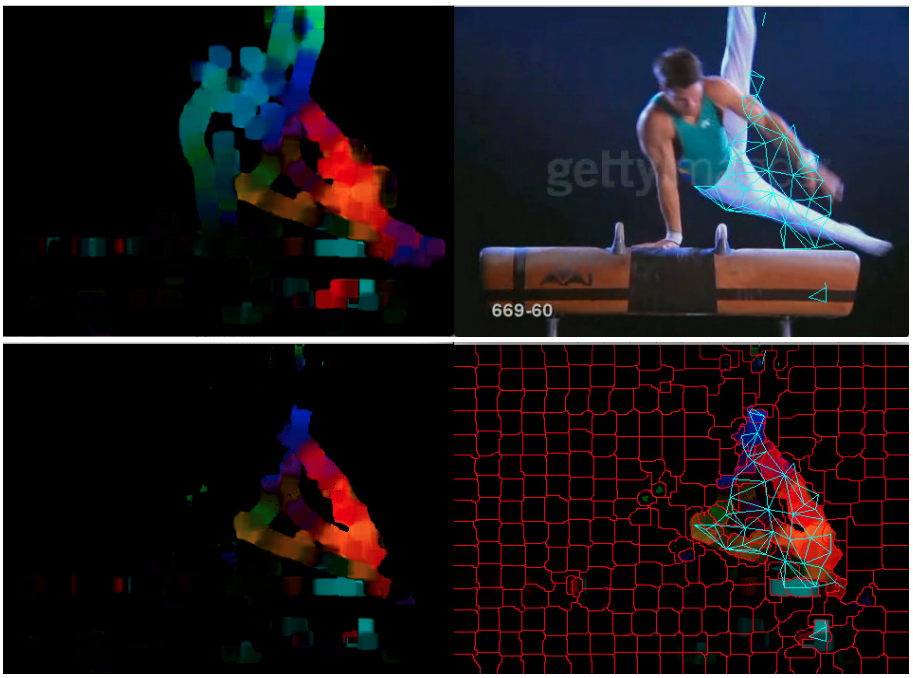


Figure 7 Motion superpixels for gymnastic movement.

[29]. The definition of a generalized kernel histogram is

$$K(x, x') = \sum_{i=1}^{m} min\{|x_i|^c, |x'_i|^b\}, \tag{15}$$

where x_i and x'_i are two different histograms that each contain m bins. This comparison is done for each element i and is iterated until there are m bins. The values c and b give generalized versions of the histogram kernel to handle fields of different sizes in histogram extraction. For example, the compared histograms are in general extracted from different sized superpixels. Based on Boughorbel et al. [29], we let $c = b = 0.25$, which gives good results in a large variety of contexts.

6 Experiments

Experiments are performed using the UCF Sports dataset [30], [31] in which the scenes are shot under real conditions for sport events and include camera motion. The camera follows the actors? motion as the objects of interest. Actors also appear at various scales and their motion is freely articulated with occlusions, making this dataset challenging for action recognition. The frame rate in the UCF Sports dataset averages ten frames per second. The dataset contains ten action classes (diving, golf swing, kicking, lifting, riding a horse, running, skateboarding, swing-bench, swing-side angle, and walking) with a resolution of 720×480 pixels.

We also try to evaluate the JPL First-Person Interaction dataset [23] which is challenging because of the robotic vision. Our framework is suitable for adaptation to active vision, because the vision of a robot changes dynamically and has characteristics that are similar to camera motion. There are seven activity classes to be

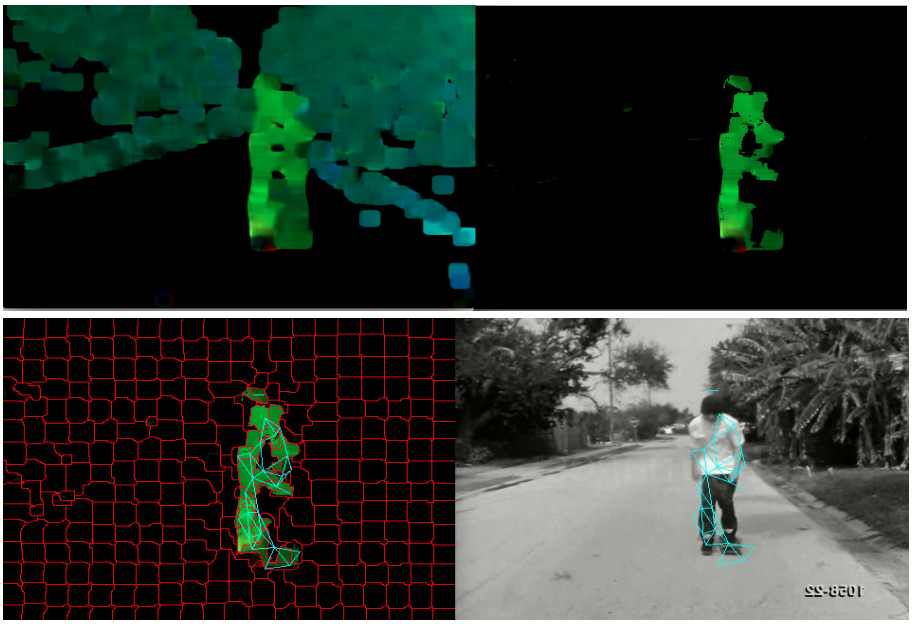


Figure 8 Motion superpixels for skateboarding.

differentiated, and there are positive, neutral, and negative interactions. In particular, the classes are shaking hands, hugging, petting, waving, pointing, punching, and throwing. Figures 7, 8, 9 show example of features generated from the UCF Sports dataset, and Figure 10 is generated from the JPL Interaction dataset.

For feeding features into the BOF algorithm, we sample five flow fields of length ten for both the JPL Interaction dataset and the UCF Sports dataset. Because of considerations of computing time, the number of initial seeds for the UCF Sports dataset is 16, 32, 64, and 128, while for the JPL Interaction dataset we use 50,150,250, and 350 and 1, 4, 16, 64 and 256. We found these sizes to be adequate for giving reasonable results. In specific cases, we try to explore how coarse or fine the flow information should be and how much locality of flow information is important for recognition in the JPL Interaction dataset. We also evaluate the datasets using tracks of length $L = 10$ and $L = 20$ to examine differences in the recognition rate. To this end, varying and fixed sizes of superpixels are compared to demonstrate the influence of large and small superpixels. The number of iterations of SEEDS is set into 50. A greater number of iterations will construct a more precise superpixel, but more computational time is required. Several experiments show that the gain in precision is not significant. We also tried various numbers of initial superpixels, but again the accuracy does not change significantly. For evaluation, leave-one-person-out classification is used for the JPL Interaction dataset, meanwhile leave-one-sample-out classification is used for the UCF Sports dataset.

For the experimental setup for the JPL Interaction dataset, we first decide how many clusters to compare between compensated and uncompensated motion. It is found that 1400-2800 clusters are suitable to give reliable accuracy with 150 and 250

Figure 9 Motion superpixels for two people skateboarding.

seeds, as in Table 1. With 1400 clusters there are 200 clusters for each class (seven classes), while with 2100 clusters there are 300 clusters for each class. Because the number of extracted features is 100000 on average, a reliable accuracy is achieved when the number of clusters per class is around 350. Therefore, we use the square root of the number of extracted features as the total number of clusters for the JPL Interaction dataset. For the UCF Sports dataset, a quarter of the square root of the number of extracted features for each class is used for the number of clusters, and therefore the total number of clusters in the codebook is the square root of the number of extracted features multiplied by the number of classes.

Table 1 Number of codewords relative to accuracy

Cluster number	700	1400	2100	2800	3500
Accuracy	0.78	0.82	0.82	0.82	0.78

Table 2 Effect of motion compensation

Superpixel seeds	Compensated motion	Uncompensated motion
250 seeds	0.82	0.75
150 & 250 seeds	0.88	0.79
50 & 150 & 250 seeds	0.85	0.75
50 & 150 & 250 & 350 seeds	0.88	0.81

To confirm that motion compensation is important, we compare results between compensated and uncompensated motion. Motion compensation has a significant impact in helping motion superpixels identify desirable features. Table 2 shows

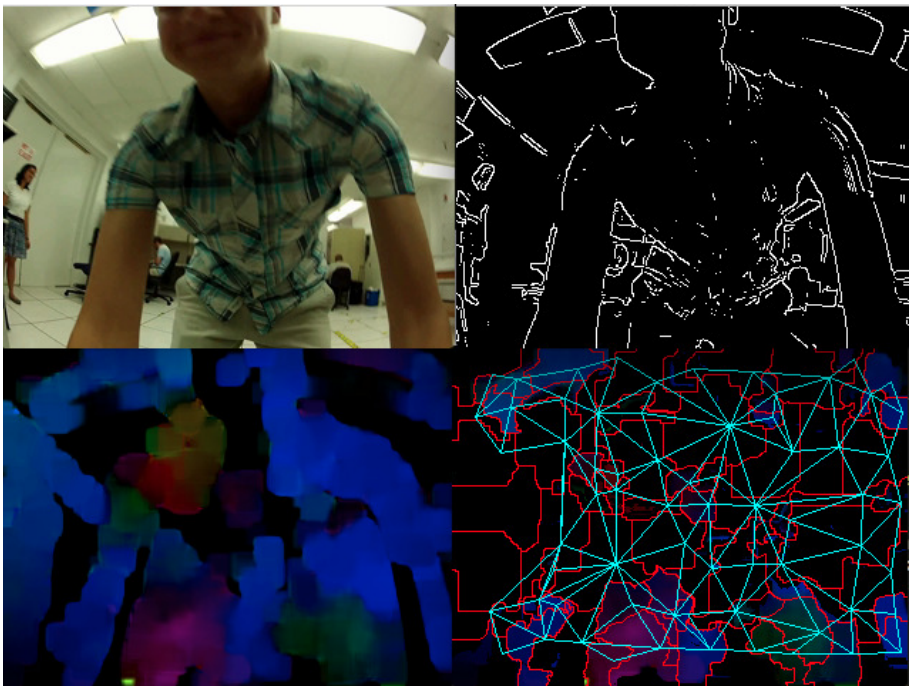


Figure 10 Motion superpixels for first-person video.

the differences in accuracy between compensated and uncompensated motion using various numbers of initial seeds.

Figure 11 shows confusion matrix results for given classes in the JPL Interaction dataset using $L = 10$. We conclude that using various superpixel sizes obtains significant results because it covers coarse to fine motions, multi-scale motions, and more sample features.

Table 3 Comparison with other methods for the JPL Interaction dataset

Method	Accuracy
Global motion descriptor	72 %
Local motion descriptor	69%
Global & Local $+\chi^2$ kernel	82 %
Local temporal motion superpixels	88 %

Compared with other state-of-the-art methods, temporal motion superpixels achieve comparable results for the JPL Interaction dataset. Table 3 shows that motion superpixels are better than global and local descriptors obtained from existing research [23].

Figure 12 shows confusion matrix results for motion superpixel tracks with $L = 20$. As in Figure 11, more varied and detailed superpixels give better recognition rates, which indicates that locality is important for capturing flow representations. Comparing Figures 11 and 12 shows that $L = 20$ gives better confusion results for all superpixel sizes. This reveals that longer tracks increase the information gain of time series, although this increase is not significant.

Figure 13 (a) shows the confusion matrix results for the UCF Sports dataset. Compared with the JPL Interaction dataset, the UCF Sports dataset has a larger

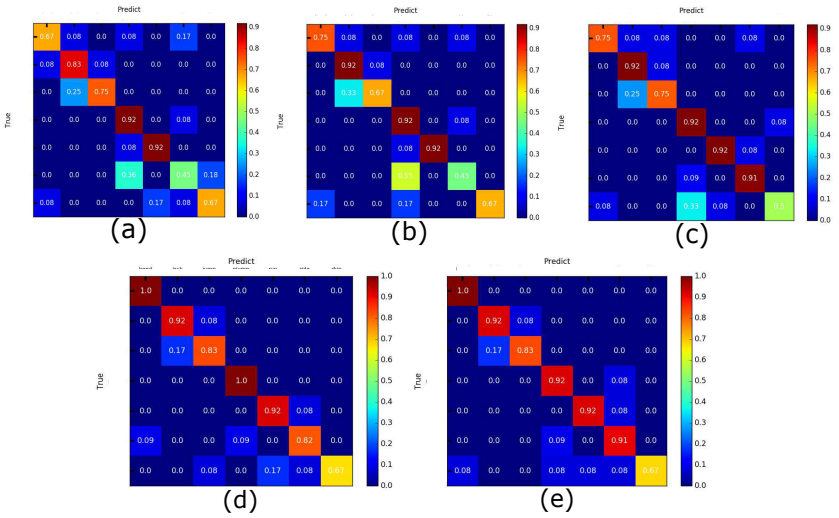


Figure 11 Confusion matrices for the JPL Interaction dataset using $L = 10$: (a) 1 seed; (b) 1 seed, 4 seeds; (c) 1 seed, 4 seeds, 16 seeds; (d) 1 seed, 4 seeds, 16 seeds, 64 seeds; (e) 1 seed, 4 seeds, 16 seeds, 64 seeds, 256 seeds. The mean accuracies are 67%, 79%, 81%, 84% and 86%, respectively. The vertical axes represent predictions, and the horizontal axes represent the truth; from top to bottom and left to right are shaking hands, hugging, petting, waving, pointing, punching, and throwing.

frame size of 720 x 480. Therefore, the initial number of seeds is set to 16 rather than one. The best confusion accuracy is 79% when using HOF descriptor and a single track. Figure 13 (b) shows that there is an improvement when spatial information is added (HOG) and bi-tracks (two-way paths) are used. This confirms that both motion and spatial information are important descriptors for activity recognition. Moreover, the use of bi-tracks enriches track information, which could be important for motion analysis.

Table 4 Comparison with other methods on the UCF Sports dataset.

Methods	Accuracy
STIP Sampling foreground only [32]	71.92 %
STIP Sampling background only [32]	73.97%
Dense sampling of STIP [?]	75.34 %
Spatial superpixel (HOF, HOG) [14]	86.7 %
Spatial superpixel (HOF) [14]	87.9 %
Dense trajectories (HOF, HOG, MBH) [21]	88.9 %
Temporal motion superpixel (HOF, HOG)	89.1 %

Table 4 shows a comparison of features sampled using non-geometric information such as cuboids. As opposed to superpixels, which give structural information about pixel boundaries, cuboid sampling is based on an arbitrary fixed size of the cuboid. We also compare with spatial superpixels [14] for superpixels via RGB images without tracking or dense trajectories [21] for pixel-level tracking. We achieve the best results using spatial superpixel (using HOF descriptors) and dense trajectories (using HOF, HOG, and motion boundary histograms (MBH)). The difficulty with our approach is that even though superpixels contain more meaningful information, motion superpixel tracking paths have many possibilities and there is a high probability of losing the optimal path, thus requiring enrichment from the multi-track approach.

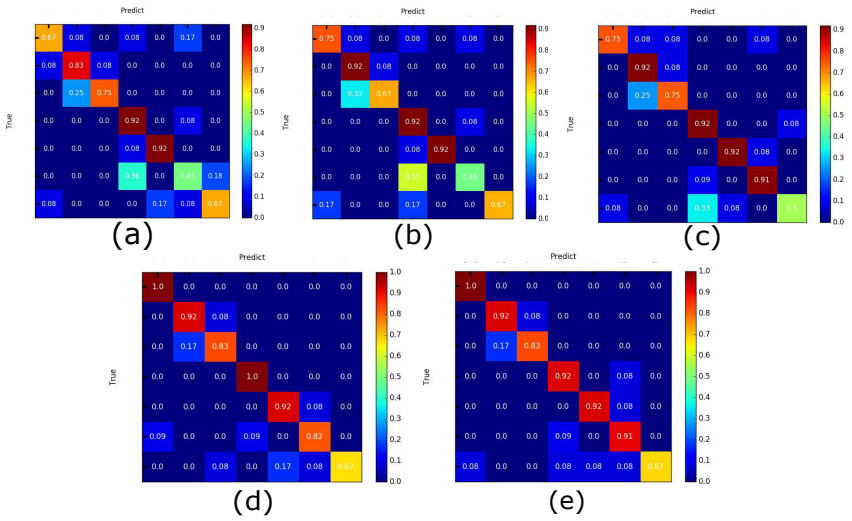


Figure 12 Confusion matrix for the JPL Interaction dataset using $L = 20$: (a) 1 seed; (b) 1 seed, 4 seeds; (c) 1 seed, 4 seeds, 16 seeds; (d) 1 seed, 4 seeds, 16 seeds, 64 seeds; (e) 1 seed, 4 seeds, 16 seeds, 64 seeds, 256 seeds. The mean accuracies are 74%, 76%, 81%, 88% and 89%, respectively. The vertical axes represent predictions, and the horizontal axes represent the truth; from top to bottom and left to right are shaking hands, hugging, petting, waving, pointing, punching, and throwing.

7 Conclusion

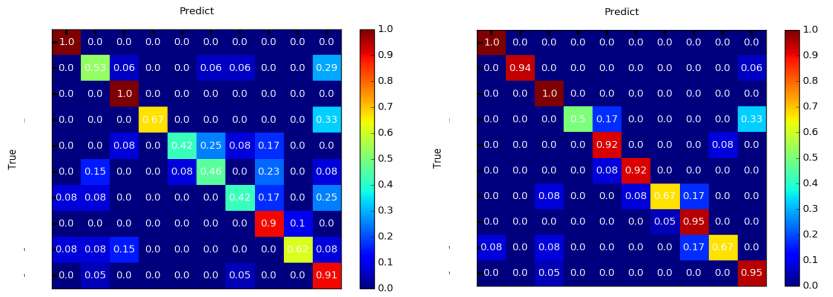
We have demonstrated local motion superpixel evolution over time using three principal local features, namely HOF, centers of mass, and HOG. By tracking the centers of mass of motion superpixels over time, feature vectors form time series data that can be used to analyze temporal dynamics. Moreover, superpixels capture locality evolution for motion that is important for achieving significant video classification performance. To enrich the temporal information, various sizes of superpixels, spatial and motion descriptors, and two-way tracking with separate BOFs can be applied. We have applied our approach to the UCF Sports dataset and the JPL First-Person Interaction dataset and found it to be comparable with existing methods. Future research will involve concatenation with global motion superpixels. There is also a possibility of merging these ideas with CNNs to better understand locality and its temporal evolution for various tasks, in particular video classification. Moreover, there is the opportunity to adapt motion superpixel in case of video segmentation with extra treatments.

Abbreviations
HOF=Histogram of Flows, HOG=Histogram of Gradients, MBH=Motion Boundary Histogram, RGB=Red Green Blue, HSV=Hue Saturation Value, BOF=Bag of Features, SVM=Support Vector Machine, SEEDS=Superpixels Extracted via Energy-driven Sampling, CNN=Convolutional Neural Networks, STIP=Spatio-temporal Interest Points, SIFT=Scale-Invariant Feature Transforms

Availability of data and materials
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Author's contributions
Novanto Yudistira contributes to designing the algorithms, coding, and writing papers. Takio Kurita contributes to supervision and several suggestions for improvement of proposed method.



(a) HOF (b) HOF, HOG

Figure 13 Confusion matrix results for leave-one-out classification on the UCF Sports dataset. The vertical axes represent predictions, and the horizontal axes represent the truth; from top to bottom and left to right are diving, golf swing, kicking, lifting, riding a horse, running, skateboarding, swing-bench, swing-side angle, and walking.

Funding

The authors declare that KAKENHI no. 16K00239 funds this research and publication.

Acknowledgements

The research is supported by KAKENHI no. 16K00239. We thank Peter Humphries, PhD, from Edanz Group (www.edanzediting.com/ac) for editing a draft of this manuscript.

Author's information

Novanto Yudistira is currently a PhD student at Hiroshima University. He received his BS in Informatics Engineering from the Institut Teknologi Sepuluh in November, 2007, and his MS in Computer Science from Univeristi Teknologi Malaysia in 2011. His current research interests include multimodal feature extraction, deep learning, computer vision and applications to machine vision.

Takio Kurita received the his B.Eng. degree from Nagoya Institute of Technology in 1981 and his Dr. Eng. degree from the University of Tsukuba in 1993. He joined the Electrotechnical Laboratory, AIST, MITI in 1981. From 1990 to 1991, he was a visiting research scientist at the Institute for Information Technology, National Research Council, Canada. From 2001 to 2009, he was a deputy director of the Neuroscience Research Institute, National Institute of Advanced Industrial Science and Technology (AIST). He was also a Professor at the Graduate School of Systems and Information Engineering, University of Tsukuba, from 2002 to 2009. He is currently a Professor at Hiroshima University. His current research interests include statistical pattern recognition and its applications to image recognition. He is a member of the IEEE, the IPSJ, the IEICE of Japan, the Japanese Neural Network Society, and The Japanese Society of Artificial Intelligence.

Author details

[1]Graduate School of Information Engineering, Hiroshima University, Japan. [2]Departement of Information Engineering, Hiroshima University, Japan.

References

1. Zhang, G., Jia, X., Kwok, N.M.: Superpixel Based Remote Sensing Image Classification with Histogram Descriptors on Spectral and Spatial Data. In: IEEE Geoscience and Remote Sensing Symposium (IGARSS) (2012)
2. Tao, W., Zhou, Y., Liu, L., Li, K., Sun, K., Zhang, Z.: Spatial adjacent bag of features with multiple superpixels for object segmentation and classification. Information Sciences 281 (2014)
3. Fang, L., Li, S., Kang, X., Benediktsson, J.A.: Spectral–spatial classification of hyperspectral images with a superpixel-based discriminative sparse model. IEEE Transactions on Geoscience and Remote Sensing (2015)
4. Cheng, J., Liu, J., Xu, Y., Yin, F., Wong, D.W.K., Tan, N.-M., Tao, D., Cheng, C.-Y., Aung, T., Wong, T.Y.: Superpixel classification based optic disc and optic cup segmentation for glaucoma screening. IEEE Transactions on Medical Imaging (32), 1019–1032 (2013)
5. Van den Bergh, M., Boix, X., Roig, G., de Capitani, B., Van Gool, L.: Seeds: Superpixels Extracted Via Energy-driven Sampling. In: ECCV (2012)
6. Vazquez-Reina, A., Avidan, S., Pfister, H., Miller, E.: Multiple Hypothesis Video Segmentation from Superpixel Flows. In: Computer Vision–ECCV (2010)
7. Fragkiadaki, K., Arbelaez, P., Felsen, P., Malik, J.: Learning to Segment Moving Objects in Videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2015)
8. Giordano, D., Murabito, F., Palazzo, S., Spampinato, C.: Superpixel-based Video Object Segmentation Using Perceptual Organization and Location Prior. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2015)

9. Li, F., Kim, T., Humayun, A., Tsai, D., Rehg, J.M.: Video Segmentation by Tracking Many Figure-ground Segments. In: Proceedings of the IEEE International Conference on Computer Vision (2013)
10. Liu, Z., Zhang, X., Luo, S., Le Meur, O.: Superpixel-based spatiotemporal saliency detection. IEEE transactions on circuits and systems for video technology (2014)
11. Taralova, E.H., De la Torre, F., Hebert, M.: Motion Words for Videos. In: ECCV (2014)
12. Trichet, R., Nevatia, R.: Video Segmentation and Feature Co-occurrences for Activity Classification. In: Applications of Computer Vision (WACV) (2014)
13. Chang, J., Wei, D., Fisher, J.W.: A Video Representation Using Temporal Superpixels. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2013)
14. Dong, X., Tsoi, A.-C., Lo, S.-L.: Superpixel Appearance and Motion Descriptors for Action Recognition. In: International Joint Conference on Neural Networks (IJCNN) (2014)
15. Van den Bergh, M., Roig, G., Boix, X., Manen, S., Van Gool, L.: Online Video Seeds for Temporal Window Objectness. In: Proceedings of the IEEE International Conference on Computer Vision (2013)
16. Shiraki, T., Saito, H., Kamoshida, Y., Ishiguro, K., Fukano, R., Shirai, T., Taura, K., Otake, M., Sato, T., Otsu, N.: Real-time Motion Recognition Using Chlac Features and Cluster Computing. In: Proceedings of the 3rd IFIP International Conference on Network and Parallel Computing (2006)
17. Kim, S., Nowozin, S., Kohli, P., Yoo, C.D.: Higher-order Correlation Clustering for Image Segmentation. In: Advances in Neural Information Processing Systems (2011)
18. Ke, Y., Sukthankar, R., Hebert, M.: Spatio-temporal Shape and Flow Correlation for Action Recognition. In: Computer Vision and Pattern Recognition (2007)
19. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229 (2013)
20. Laptev, I.: On space-time interest points. International Journal of Computer Vision, 107–123 (2005)
21. Wang, H., Kläser, A., Schmid, C., Liu, C.-L.: Dense trajectories and motion boundary descriptors for action recognition. International journal of computer vision (2013)
22. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision (2004)
23. Ryoo, M., Matthies, L.: First-person Activity Recognition: What Are They Doing to Me? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2013)
24. Iwashita, Y., Takamine, A., Kurazume, R., Ryoo, M.S.: First-person Animal Activity Recognition from Egocentric Videos. In: Pattern Recognition (ICPR), 2014 22nd International Conference On. IEEE (2014)
25. Wu, Z., Hu, Z., Fan, Q.: Superpixel-based Unsupervised Change Detection Using Multi-dimensional Change Vector Analysis and SVM-based Classification. In: ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci 7, pp. 257–262 (2012)
26. Farneback, G.: Two-frame motion estimation based on polynomial expansion. Image analysis of Springer Berlin Heidelberg (2003)
27. Bachem, O., Lucic, M., Hassani, H., Krause, A.: Fast and Provably Good Seedings for k-Means. In: Advances in Neural Information Processing Systems (2016)
28. Khan, F.S., Van De Weijer, J., Bagdanov, A.D., Felsberg, M.: Scale Coding Bag-of-words for Action Recognition. In: Pattern Recognition (ICPR), 2014 22nd International Conference On. IEEE (2014)
29. Boughorbel, S., Tarel, J.-P., Boujemaa, N.: Generalized Histogram Intersection Kernel for Image Recognition. In: Image Processing, IEEE International Conference On. Vol. 3. IEEE
30. Rodriguez, M.D., Ahmed, J., Shah, M.: Action Mach a Spatio-temporal Maximum Average Correlation Height Filter for Action Recognition. In: Computer Vision and Pattern Recognition (2008)
31. Soomro, K., Zamir, A.R.: Action recognition in realistic sports videos. Computer Vision in Sports, 181–208 (2014)
32. Sultani, W., Sulaemi, I.: Human action recognition across datasets by foreground-weighted histogram decomposition, In: Computer Vision and Pattern Recognition (2014)

List of Figures

List of Tables