Motion Boundary Trajectory for Human Action Recognition

Sio-Long Lo and Ah-Chung Tsoi

Faculty of Information Technology,
Macau University of Science and Technology

Abstract. In this paper, we propose a novel approach to extract local descriptors of a video, based on two ideas, one using motion boundary between objects, and, second, the resulting motion boundary trajectories extracted from videos, together with other local descriptors in the neighbourhood of the extracted motion boundary trajectories, histogram of oriented gradients, histogram of optical flow, motion boundary histogram, can be used as local descriptors for video representations. The motion boundary approach captures more information between moving objects which might be caused by camera movements. We compare the performance of the proposed motion boundary trajectory approach with other state-of-the-art approaches, e.g., trajectory based approach, on a number of human action benchmark datasets (YouTube, UCF sports, Olympic Sports, HMDB51, Hollywood2 and UCF50), and found that the proposed approach gives improved recognition results.

1 Introduction

Recognizing human action in a video is a commonly studied topic in computer vision and machine learning [1–4]. Broadly speaking, a popular approach is to first extract a set of local descriptors, and then use a bag-of-features model for matching those local descriptors obtained in the set of labeled training video clips, to those as yet unlabelled in the testing dataset [5–7].

Laptev [8] introduced space-time interest points (STIPs) using an extension of the Harris corner detection method [9] from image to video. Other detectors are also used to detect interest points in videos, e.g., Willems et al. [10] proposed using the determinant of the spatiotemporal Hessian matrix for interest point detection, Dollar et al. [11] proposed a 1D Gabor filter in the time dimension with a 2D Gaussian in the spatial dimensions to detect the underlying periodic frequency components for interest point detection.

Based on the detected interest points in a video, a descriptor is proposed to describe the information of sub-regions of the video as local features. Several descriptors have been proposed for describing these spatiotemporal local features, e.g., higher order derivatives (local jets) [8], histogram of oriented gradient (HOG) [12] for capturing object shape, these are called the appearance descriptors; histogram of optical flow (HOF) [12] for capturing object motion

information, a spatiotemporal version of HOG, called HOG3D [13] which extends the idea of HOG to the 3D case, histogram of oriented flows (HOF), a way of representing movements across time [12], and motion boundary histograms (MBH) [14] to cope with the camera motion. This detector/descriptor approach can be considered as a kind of bag-of-features video representation.

In contrast from detecting interest points in a 3D volume data, another approach to obtaining local features from a video is the trajectory approach, so called dense trajectory approach, as the patch is represented by a large number of interest points, [4, 15]. In this approach, a set of local interest points is first detected using the 2D Harris condition [9] from video frames and an optical flow field is then used to track these interest points temporally to form the patch trajectories in the video [4]. The trajectory descriptor, together with the local descriptors, can be used to represent the video under a bag-of-features framework.

However, it is difficult to detect the actual moving objects in a complex background scene with severe camera motion using the 2D Harris corner condition [9] as the local patch detector. In this paper, we wish to show that the motion patterns of objects are important and will help detect informative patch trajectories for action recognition. In [16], the authors also introduced a motion boundary based sampling for action recognition, though it is different from the one which we proposed in this paper. The fact that motion provides important cue for grouping objects is well known [17]. On the other hand, to cope with camera motion, Dalal et al. introduced the motion boundary histogram (MBH) [14] as an effective local descriptor. MBH encodes the gradients of optical flow, which are helpful for canceling constant camera motion. Despite the importance of MBH as clearly shown in Dalal et al [14], it appears that no one has yet explored the idea of a motion boundary in the dense trajectory approach [4]. It is expected that if we can embed the motion boundary concept in the dense trajectory approach [4], then it can handle issues related to camera motion, and thus would result in improved recognition rate, for datasets which may have taken while the camera might be moving. In this paper, we propose to use the motion boundary between objects for detecting local patches within the dense trajectory approach [4]. The motion boundary can capture more informative information between moving objects which might be caused because the camera was moving. With the motion boundary defined, the motion boundary trajectory can be extracted and can be used for the video representation. We compare the performances of various approaches on a number of standard benchmark datasets [18–22] and achieve better results using the proposed approach.

The rest of this paper is organized as follows. Section 2 discusses related work; Section 3.3 briefly introduces the concept of local descriptor extractions from videos, which include motion boundary trajectories (in Section 3.2), appearance based descriptors and motion descriptors; Section 4 provides approaches to classification; experimental results are shown in Section 5. Finally, some conclusions are drawn in Section 6.

Contribution: This paper establishes the deployment of motion boundary determination in the dense-trajectory approach for action recognition. The motion boundary between objects is determined and then those points in this motion boundary are tracked to form the motion boundary trajectories for video representation. Experimental results show that this idea can improve the performance of recognition significantly.

2 Relative Works

The most popular approach for action recognition is the well known bag-of-feature model [23, 19, 24]. In this model, the selection of local features of a video is important for the video representation. There are two broad approaches within this tradition: the detector/descriptor approach [18] and the trajectory approach [4]. In the detector/descriptor approach [18], the detector is used to detect interesting sub-regions of a video, contained within such sub-regions are typically the intensity values that have significant local variations in both space and time. For these sub-regions, the descriptors are applied to describe the spatial-temporal local features of the video [18]. The dense trajectory approach [4] tracks the detected local patches in the video frames through time. Then patch trajectories can be extracted from these sub-regions of the video. In the dense trajectory approach, the extracted spatial-temporal local features are significant [4]. It can be explained that the detected/extracted features are specifically based on object appearances and, to some extent, on motions (as the motion boundary histogram is used to represent the motion).

Some related work can be found in motion segmentation and video co-segmentation [25]. Motion segmentation is the problem of decomposing a video and to detect moving objects and background based on the idea of coherent regions with respect to motion and appearance properties [25]. Motion information provides an important cue for identifying the surfaces in a scene and for differentiating image texture from physical structures. In [17], long term point trajectories based on dense optical flow are used to spatial–temporal cluster the feature points into temporally consistent segmentations of moving objects. The quality of motion segmentation depends significantly on the pair of frames with a clear motion difference between the objects [26]. The advantage of motion segmentation derives from the fact that it combines motion estimation with segmentation. For segmenting multiple objects in the scene, the layered model for motion segmentation is proposed [27]. Typically, the scene consists of a number of moving objects and representing each moving object by a layer that allows the motion of each layer to be described [27]. Such a representation can model the occlusion relationships among layers making the detection of occlusion boundaries possible [28, 29]. Typically, the background/foreground segmentation is a special case of binary object segmentation in this layered model [30].

In [25], multiple objects and multi-class video co-segmentation task is proposed to segment objects in videos. Object co-segmentation [25] is to segment a prominent object based on an image pair in which it appears in both images.

With this idea, video co-segmentation segments the objects that are shared between videos, therefore co-segmentation can be encouraged. With this approach, object boundaries can be detected [28, 29].

Based on the idea of motion segmentation, objects may be segmented from the background in the action recognition. Inspired by the idea of motion boundary histogram descriptor in the bag-of-feature framework, in this paper, we propose to use the boundary between objects as a descriptor in the dense-trajectory approach. The motion boundary can then be tracked frame by frame and then deployed as a descriptor, very much in the same manner as the patch trajectories in the dense trajectory approach [4] and then used for action recognition. This has the advantage of not requiring to perform the segmentation or co-segmentation task which are very time consuming tasks, where there is no significant occlusion of the objects involved.

3 Motion Boundary Trajectories

In this section, we will describe the proposed motion boundary dense trajectory approach. We will first describe the dense trajectory approach [4] briefly, and then we will show how motion boundary trajectories can be extracted from the video.

3.1 Dense Trajectories

The idea of a trajectory is based on interest points tracking [4]; the interest points are tracked frame by frame and then the corresponding trajectory can be extracted based on the tracked points [4]. For the motion boundary trajectories, we first detect the motion boundary on video frames and then track the detected motion boundary through time to form the motion boundary trajectories of a video.

Consider a video which consists of $I^{(t)}, t = 1, 2, \ldots, T$ and $I^{(t)}$ is a 2D pixel intensity array with dimensions $W \times H$. The optical flow field is computed over a two-frame sequence $I^{(t)}$ and $I^{(t+1)}$, $\omega^{(t)} = (u^{(t)}, v^{(t)})$, where, $u^{(t)}$, $v^{(t)}$ are respectively the optical flow in the horizontal and vertical directions. We apply a median filtering on the optical flow field $\omega^{(t)} = (u^{(t)}, v^{(t)})$ within a 3×3 patch. The resulting optical flow field is denoted by $\bar{\omega}^{(t)} = (\bar{u}^{(t)}, \bar{v}^{(t)}) = \omega^{(t)} \star M_{3\times3}$, where $M_{3\times3}$ is the median filter kernel and $\bar{\omega}^{(t)}$ is the filtered result of the optical flow field and $\star$ is the convolution operator.

In the dense trajectory approach [4], the Harris corner condition [9]. With this selection, a set of interest points, determined using a 2D Harris corner condition [9] on the object appearance, is then tracked frame by frame to form the dense trajectories.

In other to cope with the camera motion, a matching of feature points using SURF descriptors and dense optical flow is applied to eastimate a homography between two subsequent frames by RANSAC algorithm as in [31]. Based on the reason of human action is ingeneral different from camera motion. A human

detector is employed to remove matches from human regions to improve the camera motion eastimation. Finally, the trajectories consistent with the camera motion are then removed no longer to for the tracking process [31].

3.2 Motion Boundary Trajectories

Different from using object appearances, motion boundary trajectory approach is based on the motion boundary between objects. To detect the motion boundary, we extract its location using the optical flow. Assume each object will have different flow directions and velocities, we detect their boundaries using the derivative of the optical flow field which captures the discontinuity, e.g., edges, of the optical flow field. For the point $P_i^{(t)} \in I^{(t)}$, the measurement of its boundary is given by

$$H_{P_i^{(t)}} = ||\nabla \bar{u}_{P_i^{(t)}}||^2 + ||\nabla \bar{v}_{P_i^{(t)}}||^2$$

where, $(\bar{u}_{P_i^{(t)}}, \bar{v}_{P_i^{(t)}})$ is the flow vector of point $P_i^{(t)}$.

The determination of the motion boundary trajectories is very similar to that proposed in [4] in the dense trajectory approach. Given a dense grid of frame $I^{(t)}$, we can densely sample points on a grid spaced by w pixels. In our case, the dense grid is set to 5×5. Sampling is carried out on each spatial scale separately. Different scales can be obtained by simply re-sizing the video to different resolutions, with a scaling factor of $\frac{1}{\sqrt{2}}$. In our setting, there are at most 8 spatial scales in total [4]. To obtain the motion boundary trajectories, we first select the points based on Harris corner condition

$$T_{corner}^{(t)} = C_1 \times \max_{P_i^{(t)} \in I^{(t)}} \min(\lambda^1_{P_i^{(t)}}, \lambda^2_{P_i^{(t)}})$$

where, $(\lambda^1_{P_i^{(t)}}, \lambda^2_{P_i^{(t)}})$ are the eigenvalues of the auto-correlation matrix of point $P_i^{(t)}$ in frame $I^{(t)}$. We then thresholds the motion boundary based on the threshold $T_{corner}^{(t)}$ as

$$\tilde{H}_{P_i^{(t)}} = \begin{cases} H_{P_i^{(t)}} & \min(\lambda^1_{P_i^{(t)}}, \lambda^2_{P_i^{(t)}}) \geq T_{corner}^{(t)} \\ 0 & otherwise \end{cases}$$

We then use another threshold condition for which a point is of interest (i.e., significant enough for further consideration):

$$T_{motion}^{(t)} = C_2 \times \max_{P_i^{(t)} \in I^{(t)}} \tilde{H}_{P_i^{(t)}} + C_3$$

The point $P_i^{(t)}$ will be selected, if its magnitude is greater than the threshold, i.e., $\tilde{H}_{P_i^{(t)}} > T_{motion}^{(t)}$, while those points which do not satisfy this condition will not be considered further. In our setting, we set $C_1 = 0.0001$, $C_2 = 0.01$

and $C_3 = 0.002$. From the above process, we will know which sub-sampled point $P_i^{(t)}$ will need to be considered for the trajectory tracking. We then track the selected points using optical flow field $\bar{\omega}^{(t)} = (\bar{u}^{(t)}, \bar{v}^{(t)})$. Consider a point $P_i^{(t)} = (x_i^{(t)}, y_i^{(t)})$ in frame $I^{(t)}$, the tracked point $P_i^{(t+1)} = (x_i^{(t+1)}, y_i^{(t+1)})$ of $P_i^{(t)}$ in the next frame $I^{(t+1)}$ is computed by:

$$\begin{aligned} P_i^{(t+1)} &= P_i^{(t)} + \bar{\omega}_{t,P_i^{(t)}} \\ &= (x_i^{(t)}, y_i^{(t)}) + (\bar{u}_t, \bar{v}_t) \mid_{(x_i^{(t)}, y_i^{(t)})} \end{aligned}$$

The tracked points of subsequent frames are then concatenated temporally to form a trajectory, $\text{Traj}_i = (P_i^{(t)}, P_i^{(t+1)}, P_i^{(t+2)}, \ldots)$. For each frame, if no tracked point is found in the neighborhood, a new point $P_{i^*}^{(t)}$ is sampled and added to the tracking process. If the length of a trajectory has reached a maximum length $L = 15$, a post-processing stage is then performed to remove the static trajectories [4].

In other to obtain a better motion boundary, we follow [31] and estimate the homography of two subsequent frames, and then warp the second frame with the estimated homography. Based on the warped frame, the Harris cornerness is computed by the warped second frame and the optical flow is computed between the first and the warped second frame. To obtain more interest points arounding the moving obejcts, we apply a Gaussian filter and then a median filter on the motion boundary map, i.e., $\tilde{H}$. We then select and track the points for extracting the motion boundary trajectoies. For the optical flow, we use the Farneback optical flow algorithm [32], which employs a polynomial expansion to approximate the pixel intensities in the neighborhood to obtain a good quality flow field as well as capturing some fine details [4]. Figure 1 shows the results of the motion boundary as well as the motion boundary trajectory obtained from some selected videos.

It is observed that the motion boundary trajectories can capture the motion quite well.

3.3 Motion Boundary Descriptors

Local descriptors are features which describe the spatial temporal behaviours of humans in the video. There are a number of such descriptors proposed by various researchers: [4]. The essential idea is to find good descriptors which will describe the spatial temporal behaviours of pixel values in a small neighborhood of a volume consisting of two dimensional space and time [4]. Some of these methods were extended from image processing techniques, while others were constructed explicitly for spatial temporal behaviours [4].

Several descriptors can be obtained to encode either the shape of a trajectory or the local motion [4] and appearance within a space-time volume [14] around the trajectory. The trajectory shape descriptor encodes local motion patterns by using the displacement vectors of a trajectory [4]. HOG (Histogram of oriented gradient) along a trajectory focuses on the static part of the appearance of a

Fig. 1. The first row shows the original images; the second row shows the detected motion boundaries; and the third row shows the corresponding motion boundary trajectories

local patch of the video. For encoding the motion information, HOF (Histograms of optical flow) captures the local motion information based on the optical flow field; MBH (Motion boundary histogram) uses the gradient of the optical flow to cancel out most of the effects of camera motion [14]. These descriptors give a state-of-the-art performance for representing local information.

In this paper, we will add the motion boundary trajectories as the descriptors for the motion in the time axis. The motion trajectory descriptor can be formed by considering the shape of the trajectories, in a manner very similar to that proposed in [4]. Given a trajectory of length L, a sequence $(\triangle P_i^{(t)}, \ldots, \triangle P_i^{(t+L-1)})$ of the displacement vectors $\triangle P_i^{(t)} = P_i^{(t+1)} - P_i^{(t)} = (x_i^{(t+1)} - x_i^{(t)}, y_i^{(t+1)} - y_i^{(t)})$ is used for describing the trajectory shape. The normalized concatenation of the displacement vectors will become the feature vector of the trajectory shape:

$$\text{Shape}_i = \frac{(\triangle P_i^{(t)}, \ldots, \triangle P_i^{(t+L-1)})}{\sum_{k=t}^{t+L-1} \parallel \triangle P_i^{(k)} \parallel}$$

With the motion boundary trajectory, $\text{Traj}_i = (P_i^{(t)}, P_i^{(t+1)}, P_i^{(t+2)}, \ldots)$, the corresponding HOG, HOF and MBH descriptors can also be extracted based on the motion boundary trajectory as the trajectory based HOG, HOF and MBH descriptors (please see Figure 2 for an illustration of these concepts). We follow [31], motion descriptors (HOF and MBH) are computed on the warped optical flow. The trajectory shape descriptor and HOG descriptor remains unchanged.

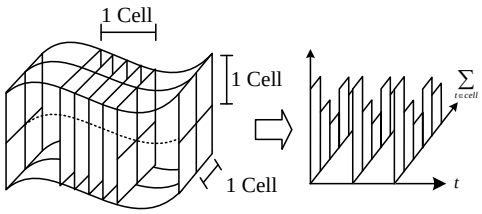

Fig. 2. Illustration of Motion Boundary Trajectory Descriptor. The motion boundary trajectory is represented by relative point coordinates, $\text{Traj}_i = (P_i^{(t)}, P_i^{(t+1)}, P_i^{(t+2)}, \ldots)$; based on the motion boundary trajectories, the HOG, HOF and MBH descriptors are computed along the trajectories.

4 Classification

We apply the standard bag-of-features approach to convert the local descriptors from a video into a fixed-dimensional vector. We first construct a codebook for the trajectory descriptor (Section 3.3) using the k-mean clustering algorithm, and then the clusters will serve as visual words. We fix the number of visual words to $V = 4,000$. To limit the complexity of the problem, we cluster a subset of 100,000 randomly selected from the training features in the k-mean clustering algorithm. Descriptors are then assigned to their closest vocabulary word using an Euclidean norm. The resulting histograms of visual word occurrences are used as video representations.

We apply the linear and non-linear SVM for action recognition. For the linear SVM [33], we first scale the value of each visual word feature to $[0, 1]$, and then the feature vector of a video is normailzied by a norm-2 normalization. For the nonlinear SVM [12], we normalize the histogram using the RootSIFT approach [34], i.e ., square root each dimension after L1 normalization, and then apply the standard RBF (radial basis function)-χ^2 kernel [4] as the baseline algorithm in our experiments.

$$K_{\chi^2}(H_i, H_j) = \exp\left(-\frac{1}{2A}\sum_{k=1}^{V}\frac{(h_{ik} - h_{jk})^2}{h_{ik} + h_{jk}}\right)$$

where $H_i = \{h_{ik}\}_{k=1}^{V}$ and $H_j = \{h_{jk}\}_{k=1}^{V}$ are the frequency histograms of word occurrences and V is the vocabulary size. A is the mean value of distances between all training samples [18]. In the case of multi-class classification, the one-against-all approach is applied, we select the class with the highest score. Typically, the approach for integrating the contribution of different descriptors is the multiple channel SVM [12, 7], which is a special case of multiple kernel learning [35]. We simply average the kernels computed from different representations to combine different channels using the idea of multiple channel SVM.

We also apply the Fisher vector [36] encoding for video representation. Fisher vector encode both first and second order statistics between the video descriptors

and a Gaussian Mixture Model (GMM). We follow [31], first reduce the descriptor dimensionality by Principal Component Analysis (PCA), as in [31]. We set the number of Gaussians to $K = 256$ and randomly sample a subset of 256,000 features from the training set to estimate the GMM [31]. As a result, for each type of descritpor, each video is represented by a $2DK$ dimensional Fisher vector, where D is the dimension of the descriptor after performing PCA. Finally, we apply power and the RootSIFT approach normalization to the Fisher vector. For integrating different descriptor types, we concatenate their normalized Fisher vectors, and a linear SVM is used for classification.

5 Experiments

This section evaluates the proposed motion boundary trajectories as a descriptor. We run the experiments at least 3 times for descriptor-classifier pairs. We will report the average accuracy of those experiments.

5.1 Datasets

We evaluate our proposed motion boundary descriptor on five standard benchmark datasets, viz., UCF-Sports [20], YouTube dataset [19], Olympic Sports dataset [21], the HMDB51 dataset [22], the Hollywood2 datasets, and the UCF50 datasets.

The UCF-Sports dataset contains 150 videos from ten action classes, diving, golf swinging, kicking, lifting, horse riding, walking, running, skating, swinging (on the pommel horse and on the floor), and swinging (at the high bar). These videos are taken from real sports broadcasts and the bounding boxes around the subjects are provided for each frame. We follow the protocol proposed in [37, 38] using the same training/testing samples for our experiments; by taking one third of the videos from each action category to form the test set, and the rest of the videos are used for training. Average accuracy over all classes is reported as the performance measure.

The YouTube dataset contains 11 action categories: basketball shooting, biking/cycling, diving, golf swinging, horse back riding, soccer juggling, swinging, tennis swinging, trampoline jumping, volleyball spiking, and walking with a dog. For each category, the videos are grouped into 25 groups with more than 4 action clips in it. The dataset contains a total of 1,168 sequences. We follow the original setup [19], using leave-one-out cross-validation for a pre-defined set of 25 groups. Average accuracy over all classes is reported as the performance measure.

The Olympic Sports dataset [21] consists of athletes practising different sports, which are collected from YouTube and annotated using the Amazon Mechanical Turk technique. There are 16 sports actions: high jump, long jump, triple jump, pole vault, discuss throw, hammer throw, javelin throw, shot put, basketball layup, bowling, tennis serve, platform (diving), springboard (diving), snatch (weight lifting), clean and jerk (weight lifting) and vault (gymnastics), represented by a total of 783 video sequences. We adopt the train/test split from

[21]. The mean average precision (mAP) over all classes [12, 39] is reported as the performance measure.

The HMDB51 contains 51 distinct action categories, each containing at least 101 clips for a total of 6,766 video clips extracted from a wide range of sources. We follow the original evaluation protocol using three train-test splits [22]. For every class and split, there are 70 videos for training and 30 videos for testing. We report the average accuracy over three-splits as performance measure.

The Hollywood2 dataset [40] has been collected from 69 different Hollywood movies and includes 12 action classes. It contains 1,707 videos split into a training set (823 videos) and a test set (884 videos). Training and test videos come from different movies. The performance is measured by mean average precision (mAP) over all classes, as in [40].

The UCF50 dataset [41] has 50 action categories, consisting of real-world videos taken from YouTube. There are 50 categories in UCF50 dataset, the videos are split into 25 groups. For each group, there are at least 4 action clips. In total, there are 6,618 video clips. We apply the leave-one-group-out cross-validation as recommended by the authors and report average accuracy over all classes.

5.2 Experimental Results

The experimental results using bag-of-feature histogram are shown in Table 1. We also list the results of improved dense trajectory approach [4] in our experiments, under the name Dense Trajectory in Table 1. For the dense trajectory approach, the 2D interest points are detected based on corner condition [4], and then track the detected points frame by frame to form the dense trajectories. From the results listed in Table 1, we note that the best performance is achieved using our motion boundary trajectory descriptor.

	UCF Sport				YouTube			
	Dense Trajectory		Motion Boundary		Dense Trajectory		Motion Boundary	
	Linear	χ^2 SVM	Linear	χ^2 SVM	Linear	χ^2 SVM	Linear	χ^2 SVM
Traj. Shape	73.1	79.4	70.6	**83.8**	66.0	76.4	71.2	**78.6**
HOG	71.6	74.5	72.8	**80.0**	69.0	74.4	69.5	**74.6**
HOF	75.9	82.3	85.1	**91.5**	76.8	80.9	78.0	**82.2**
MBH	78.0	80.9	80.4	**84.2**	77.4	**85.1**	78.1	84.2
Combined	82.3	85.1	90.2	**90.6**	86.6	87.1	**87.9**	87.4
	Olympic Sports				HMDB51			
	Dense Trajectory		Motion Boundary		Dense Trajectory		Motion Boundary	
	Linear	χ^2 SVM	Linear	χ^2 SVM	Linear	χ^2 SVM	Linear	χ^2 SVM
Traj. Shape	65.8	73.3	67.7	**76.7**	19.2	34.8	23.4	**39.1**
HOG	66.0	70.8	68.1	**73.4**	22.8	**33.5**	20.9	32.9
HOF	73.9	78.2	78.9	**80.6**	26.8	42.2	30.3	**45.3**
MBH	80.1	81.6	**83.9**	83.2	28.9	46.6	30.8	**50.0**
Combined	85.4	84.0	**86.5**	84.7	49.7	53.6	52.5	**56.7**

Table 1. Experimental results of Motion Boundary Trajectory on different datasets.

We found that on the UCF Sports dataset, the motion boundary trajectory descriptor together with HOF as well as MBH obtain very good results. The UCF Sports dataset contains videos which are typically featured on broadcast television channels, e.g., BBC and ESPN; these videos are recorded by professional cameramen and camera movement is relatively smooth. As a result, the detected motion boundary is much more meaningful, which is shown in Figure 3. This observation is also true with the Olympic Sports dataset, in which the motion boundary trajectory with MBH descriptor obtain good results.

The videos of YouTube dataset are collected from YouTube and are personal videos. This dataset is very challenging due to large variations in camera motion. In this case, the motion boundary trajectories are not very accurate. As a result, the performance of motion boundary trajectory only improve slightly that compare with dense trajectory.

Fig. 3. Comparison between the dense trajectories and motion boundary trajectories (the first row shows dense trajectory; the second shows motion boundary trajectory)

We also evaluated the performance of combining representations named Combined as listed in Table 1. We evaluated two different classifiers, viz., the linear SVM and the χ^2 SVM. We simply average the kernel matrices computed from different representations to obtain the aggregated results. The motion boundary trajectory also improves the performance at least 1% on the UCF Sports and HMDB51 datasets and slightly improves on YouTube and Olympic Sports datasets.

Figures 3 show the motion boundary trajectories and the dense trajectories. In Figure 3, we note that the motion boundary detected in some videos is significant, the motion boundary can capture the trajectories around the moving objects when compare with those obtained from the dense trajectory approach.

Comparison to the state of the art. In [31], Wang introducted improved dense trajectory feature for action recognition. Together with the fisher vector encoding for video representation, Wang obtained state-of-the-art results. We use the same setting as in [31] but instead of extracting dense trajectory, we extract the motion boundary trajectory. We also use the human boundary boxes

provided by authors [31] for better eastimation of homography between two subsequent frames. The experimental result in Table 2, we also listed the result from [31], named as IDT (improved dense trajectory). In Table 2, we noted that the Olympic Sports dataset, the motion boundary trajectory (MBT) approach obtains at least 2% improvement. We obtain 93.5% mAP. For the HMDB51 dataset, we obtain at least 5% improvement and obtain 63.8 accuracy. For the Hollywood2 dataset, the improvement is not too much, only 0.1% improvement. For the UCF50 dataset, we get 1% improvement and obtain 92.2% accuracy. Those results show that the motion boundary is useful for describing the motion information and significantly improve the recognition accuracy in action recognition.

	Olympic Sports		HMDB51		Hollywood2		UCF50	
	IDT[31]	MBT	IDT[31]	MBT	IDT[31]	MBT	IDT[31]	MBT
Traj. Shape	77.2	**81.5**	32.4	**35.9**	**48.5**	45.8	**75.2**	74.5
HOG	78.8	**82.1**	40.2	**43.2**	**47.1**	44.3	82.6	**83.9**
HOF	**87.6**	87.5	48.9	**53.2**	**58.8**	58.1	85.1	**87.1**
MBH	89.1	**92.2**	52.1	**58.2**	60.5	**60.7**	88.9	**90.5**
Combined	91.1	**93.5**	57.2	**63.8**	64.3	**64.4**	91.2	**92.2**

Table 2. Experimental results of Motion Boundary Trajectory on different datasets using Fisher vector video representation; IDT means Improved Dense Trajectory, and MBT means Motion Boundary Trajectory; The results listed in IDT here are from [31].

6 Conclusion

In this paper, we propose a novel approach based on two ideas, one using motion boundary between objects, and, second, the resulting motion boundary trajectories extracted from videos as the local descriptors. These resulted in a new descriptor, the motion boundary descriptor. We compare the performance of the proposed approach with other state-of-the-art approaches, e.g., trajectory based approach, on six human action recognition benchmark datasets, and found that the proposed approach gives better recognition results.

Acknowledgment

This work was financially supported by Fundo para o Desenvolvimento das Ciencia das e da Tecnologia, Macau SAR Grant Number 034/2011/A2. The authors would like to thank Associate Prof. Markus Hagenbuchner, University of Wollongong and Prof. Franco Scarselli, University of Siena, for many helpful comments on the proposed approach.

References

1. Brendel, W., Todorovic, S.: Learning spatiotemporal graphs of human activities. In: ICCV. (2011) 778–785
2. Niebles, J.C., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. IJCV **79** (2008) 299–318
3. Guo, K., Ishwar, P., Konrad, J.: Action recognition in video by covariance matching of silhouette tunnels. In: Brazilian Symposium on Computer Graphics and Image Processing. (2009) 299–306
4. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Dense trajectories and motion boundary descriptors for action recognition. IJCV (2013)
5. Wallraven, C., Caputo, B., Graf, A.: Recognition with local features: the kernel recipe. In: ICCV. (2003) 257–264
6. Willamowski, J., Arregui, D., Csurka, G., Dance, C.R., Fan, L.: Categorizing nine visual classes using local appearance descriptors. In: ICPR Workshop on Learning for Adaptable Visual Systems. (2004)
7. Zhang, J., Lazebnik, S., Schmid, C.: Local features and kernels for classification of texture and object categories: a comprehensive study. IJCV **73** (2007)
8. Laptev, I.: On space-time interest points. IJCV **64** (2005) 107–123
9. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proceedings of the Alvey Vision Conference. (1988) 147–151
10. Willems, G., Tuytelaars, T., Gool, L.: An efficient dense and scale-invariant spatio-temporal interest point detector. In: ECCV. (2008) 650–663
11. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR. (2005) 886–893
12. Laptev, I., Marszałek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: CVPR. (2008) 1–8
13. Kläser, A., Marszałek, M., Schmid, C.: A spatio-temporal descriptor based on 3d-gradients. In: BMVC. (2008) 995–1004
14. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: ECCV. Volume 3952. (2006) 428–441
15. Matikainen, P., Hebert, M., Sukthankar, R.: Trajectons: Action recognition through the motion analysis of tracked features. In: ICCV workshop on Video-oriented Object and Event Classification. (2009)
16. Peng, X., Qiao, Y., Peng, Q., Qi, X.: Exploring motion boundary based sampling and spatial-temporal context descriptors for action recognition. In: BMVC. (2013)
17. T.Brox, J.Malik: Object segmentation by long term analysis of point trajectories. In: ECCV. (2010)
18. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: A local svm approach. In: ICPR. Volume 3. (2004) 32–36
19. Liu, J., Luo, J., Shah, M.: Recognizing realistic actions from videos in the wild. In: CVPR. (2009)
20. Rodriguez, M., Ahmed, J., Shah, M.: Action mach a spatio-temporal maximum average correlation height filter for action recognition. In: CVPR. (2008) 1–8
21. Niebles, J.C., Chen, C.W., Fei-Fei, L.: Modeling temporal structure of decomposable motion segments for activity classification. In: ECCV. (2010) 392–405
22. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: ICCV. (2011)
23. Sivic, J., Zisserman, A.: Video Google: A text retrieval approach to object matching in videos. In: ICCV. Volume 2. (2003) 1470–1477

24. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: In. (2006) 2169–2178
25. Chiu, W.C., Fritz, M.: Multi-class video co-segmentation with a generative multi-video model. In: CVPR. (2013)
26. Wang, J.Y., Adelson, E.H.: Representing moving images with layers (1994)
27. Sun, D., Sudderth, E.B., Black, M.J.: Layered segmentation and optical flow estimation over time. In: CVPR. (2012) 1768–1775
28. Black, M.J., Fleet, D.J.: Probabilistic detection and tracking of motion boundaries. IJCV **38** (2000) 231–245
29. Feghali, R., Mitiche, A.: Spatiotemporal motion boundary detection and motion boundary velocity estimation for tracking moving objects with a moving camera: a level sets pdes approach with concurrent camera motion compensation. IEEE Transactions on Image Processing **13** (2004) 1473–1490
30. Sun, D., Wulff, J., Sudderth, E., Pfister, H., Black, M.: A fully-connected layered model of foreground and background flow. In: CVPR. (2013)
31. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: ICCV. (2013)
32. Farnebäck, G.: Two-frame motion estimation based on polynomial expansion. In: Scandinavian Conference on Image Analysis. Volume 2749. (2003) 363–370
33. Chang, C.C., Lin, C.J.: Libsvm: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology **2** (2011) 1–27
34. Arandjelović, R., Zisserman, A.: Three things everyone should know to improve object retrieval. In: CVPR. (2012)
35. Gönen, M., Alpaydin, E.: Multiple kernel learning algorithms. JMLR **12** (2011) 2211–2268
36. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: ECCV. (2010) 143–156
37. Shapovalova, N., Vahdat, A., Cannons, K., Lan, T., Mori, G.: Similarity constrained latent support vector machine: an application to weakly supervised action classification. In: ECCV. (2012) 55–68
38. Lan, T., Wang, Y., Yang, W., Robinovitch, S., Mori, G.: Discriminative latent models for recognizing contextual group activities. PAMI (2011)
39. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: (The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results)
40. Marszałek, M., Laptev, I., Schmid, C.: Actions in context. In: CVPR. (2009)
41. Reddy, K.K., Shah, M.: Recognizing 50 human action categories of web videos. Machine Vision and Applications **24** (2013) 971–981