A Spatio-Temporal Feature based on Triangulation of Dense SURF

Do Hang Nga Keiji Yanai
Department of Informatics, The University of Electro-Communications, Tokyo
1-5-1 Chofu, Tokyo 182-0021 JAPAN
dohang@mm.cs.uec.ac.jp, yanai@cs.uec.ac.jp

Abstract

In this paper, we propose a spatio-temporal feature which is based on the appearance and movement of interest SURF keypoints. Given a video, we extract its spatio-temporal features according to every small set of frames. For each frame set, we first extract dense SURF keypoints from its first frame and estimate their optical flows at each frame. We then detect camera motion and compensate flow vectors in case camera motion exists. Next, we select interest points based on their movement based relationship through the frame set. We then apply Delaunay triangulation to form triangles of selected points. From each triangle we extract its shape feature along with trajectory based visual features of its points. We show that concatenating these features with SURF feature can form a spatio-temporal feature which is comparable to the state of the art. Our proposed spatio-temporal feature is supposed to be robust and informative since it is not based on characteristics of individual points but groups of related interest points. We apply Fisher Vector encoding to represent videos using the proposed feature. We conduct various experiments on UCF-101, the largest action dataset of realistic videos up to date, and show the effectiveness of our proposed method.

1. Introduction

Researches on action recognition can be divided according to their scope: recognition of action in still images or in videos. In the former case, only static cues are exploited. Our work belongs to the latter case, where motion cues are additionally employed in order to model the actions. There exist multiple well-known methods to represent the movements of actors performing the actions such as Histogram of Oriented Optical Flow (HOOF) [1] or trajectory of interest points [2]. In addition to exploiting spatial or temporal features separately, using spatio-temporal (ST) features which integrate both visual and motion characteristics of actions has also been preferred among approaches of action recognition in videos.

To represent videos, in recent years some high-level features such as human pose or human-object interaction has been investigated and obtained promising results [3, 4, 5, 6]. However, they need to setup many hypothesis and face problems mostly caused by the diversity of human actions. Thus, low-level features have still been being explored in order to overcome these problems. To extract local spatio-temporal features, one of the most popular methods is based on cuboids [7, 8]. However, to decide the cuboid size is a tough task. Instead of detecting local cuboids before extracting features from them, some recent methods tracked interest points in video sequence then leveraged the motion information from their trajectories [2, 9]. These approaches obtained good results for action recognition. To track interest points, either tracker based technique or point matching based technique has been employed. Our method is also based on trajectories of interest points. We apply LDOF [10] to estimate optical flows of all video frames.

We aim to recognize human actions in realistic videos where the background is complex. Empirical results have shown that dense features perform better for complex videos [2, 11, 12, 13]. In this paper, we propose a spatio-temporal feature based on dense SURF points that is comparable to state-of-the-arts. Our idea is inspired by the work of Noguchi *et al*. [14]. This method is the baseline we refer in this paper. They proposed to extract spatio-temporal features based on moving SURF keypoints. We address some problems of their method such as their failure in feature extraction of videos containing camera motion or holistic decision of motion threshold in the selection of interest points. We propose simple yet effective solutions to solve these problems. That means, similar to their method, our method is also based on SURF interest points with robust movements, nevertheless how we determine those points is different. Moreover, we propose to improve their feature by exploring more aspects of selected points and introducing several novel spatio-temporal descriptors. The experimental results show significant improvements of our method over

the baseline as precision boosted approximately 22% by using our method.

According to the success of the BoV (Bag of Visual words) model, the most popular framework for pattern representation basically consists of the following steps: (1) extracting local features of patterns, (2) k-means clustering extracted features to obtain a codebook ("visual vocabulary"), (3) encoding each pattern as a BoV by assigning each descriptor to its closest visual word in the codebook. Noguchi *et al*. also applied this methodology to encode videos. In this paper, we use Fisher Vector to encode videos. So far, Fisher kernel has been demonstrated as a powerful framework which integrates the strengths of generative and discriminative approaches for pattern representation. Fisher vector encoding technique was first applied to image classification task several years ago, shown to extend the BOV representation [15]. The advantage of this technique has been demonstrated that it is not limited to the number of occurrences of each visual word but it also encodes additional information about the distribution of the descriptors. The methodology is that, (1) extracting local features from images, (2) modeling the distribution of those features as mixtures of Gaussian (GMM), training a soft codebook, (3) applying Fisher kernels on obtained codebook to encode each image as a Fisher Vector. Recently, some works on action recognition have also employed this approach to encode videos and showed the effectiveness of Fisher Vector encoding over traditional BoV [16]. In this paper, we apply Fisher encoding technique as described in [15].

Our contribution is three-fold: first, a method of extracting spatio-temporal features which is comparable to the state of the arts, second, a simple yet efficient selection of interest points, and finally, novel descriptorization of spatio-temporal features. We conduct our experiments on UCF-101 [1] which is the largest-scale human action dataset up to date with 101 action categories to validate our proposed method. This dataset is not only large-scale but also a comprehensive benchmark for human action recognition in realistic under challenging settings such as large variations in camera motion, object appearance and pose, viewpoint and complicated background, etc. The experimental results demonstrate the effectiveness of our proposed feature and spatio-temporal feature based Fisher representation for action recognition.

The reminder of this paper is organized as follows: Section 2 discusses more about some related works. In section 3 the proposed feature is described in detail. Section 4 explains about conducted experiments and presents the results. Conclusions are presented in section 5.

[1] http://crcv.ucf.edu/data/UCF-101.php

2. Related work

So far, local spatio-temporal features have become popular features for representing videos in action recognition since they can capture both shape and motion characteristics of videos and provide relatively independent representation of actions. Many methods of extraction and descriptorization of spatio-temporal features [7, 17, 13, 8] have been proposed over the past few years. To determine space-time regions (called as cuboids) where features are extracted, Dollar *et al*. [7] proposed to apply 2-D Gaussian kernels to the spatial space and 1-D Gabor filters to the temporal direction. Laptev *et al*. [17] proposed an extended Harris detector to extract cuboids. Feature descriptors range from higher order derivatives (local jets), gradient information, optical flow (laptev), and brightness information to spatio-temporal extensions of image descriptors, such as 3D-SIFT [18], HOG3D [19], and Local Trinary Patterns [20]. As another method other than using cuboids, extracting local features based on trajectories of interest points also showed good results for action recognition [2, 9, 16]. Matikainen *et al*. [21] proposed to extract trajectories using a standard KLT tracker, cluster the trajectories, and compute an affine transformation matrix for each cluster center.

In this paper, we propose to improve method of extracting ST feature proposed by Noguchi *et al*. [14]. Following [14], we also extract features based on moving SURF points and use Delaunay triangulation to model the spatial relationships between interest points. We address some problems of their method such as the inability to handle camera motion or holistic decision of motion thresholds for selecting points which cause failure in extracting features of some videos. We propose to solve these problems by simple yet efficient methods of motion compensation and point selection.

As treatment for camera motion, Cinbis *et al*. [6] applied video stabilization using homography-based motion compensation approach. They estimated camera flow by calculating the homography between consecutive frames and compensate optical flow of points by removing estimated camera flow. Similarly, Jain *et al*. [9] also removed camera motion from original optical flow, nevertheless they consider affine motion as camera motion. Wu *et al*. [22] decomposed Lagrangian particle trajectories into camera-induced and object-induced components for videos acquired by a moving camera. In [2], Heng Wang *et al*. did not compensate camera motion in advance but employed motion boundary histograms which already have constant motion removed. We also reduce the influence of any existed camera motion by cancelling the constant motion. Our proposed method of motion compensation improves significantly performance of feature extraction over the baseline since it

helps not only extract features in case that camera motion exists but also detect more robust interest points.

3. Proposed spatio-temporal feature

3.1. Overview of proposed feature

Our spatio-temporal feature is based on SURF keypoints with dominant and reliable movements. SURF keypoints are extracted densely using Dense SURF [23]. Here dominant and reliable points are supposed to belong to the human who is performing the action. Our feature extraction method investigates the triangles of informative SURF points based on the features of their shape and their movements (flow vectors of their points). These triangles are produced by applying Delaunay triangulation on selected SURF points. We show that concatenating these features with SURF features of interest points can form a powerful spatio-temporal feature.

We extract features with temporal step size of N frames. We set the value of N to be small so that the extracted features are temporally dense. We operate tracking of interest points through N frames using LDOF [10]. As trajectories may drift from their precise locations during the tracking process, limiting the tracking process within short duration like this is supposed to be able to overcome this problem. In our experiments, we fix N as 5. See Fig.1 for the illustration of our method. The process of extracting our proposed spatio-temporal feature from each frame set is summarized as follows:

1. Extract SURF keypoints of the first frame using Dense SURF [23].

2. Compute optical flows from k^{th} frame ($k = 1, 2, ..., N-1$) to the next frame ($k+1^{th}$ frame) using LDOF [10].

3. Estimate camera motion in each frame and compensate motion if camera motion detected (Section 3.2).

4. Select points which are expected being more informative than the others (Section 3.3) and form triangles of selected points using Delaunay triangulation.

5. Extract a ST feature from each triangle based on its shape along with motion features of its points through the frame set (Section 3.4).

The main improvements of our method over the baseline can be summarized as follows: (1) treatment of camera motion, (2) selection of interest points and, (3) enhancement on descriptorization of ST features. We explain in details these improvements in following subsections.

3.2. Detection and compensation of camera motion

In Noguchi *et al.*'s work [14], once camera motion has been detected in a frame set, obtained information would be considered as noise, thus no points would be selected. Consequently, no features are extracted if the whole video contains camera motion. We propose to solve this problem by following a simple 2-step technique:

1. Step 1: Confirm the existence of camera motion based on optical flows of SURF keypoints. If detecting camera motion, determine the direction and magnitude of camera motion before going to the next step.

2. Step 2: Compensate motion by cancelling camera motion from original flows of SURF keypoints.

Detection of camera motion: At the first step, we aim to find out at each frame how the camera move in both horizontal direction (forward or backward) and vertical direction (up or down). This step is based on our assumption that if most points move toward the same direction, camera motion exists. Let denote P^{x^+} and P^{x^-} as number of points with positive and negative optical flows, $P_m^{x^+}$ and $P_m^{x^-}$ as number of moving points which shift forward and backward respectively, so that we suppose that camera is moving forward if Eq.1 and Eq.2 are satisfied or backward if Eq.3 and Eq.4 are satisfied:

$$P_m^{x^+} \geq k^{x^+} \tag{1}$$

$$P_m^{x^+} > P_m^{x^-} \tag{2}$$

$$P_m^{x^-} \geq k^{x^-} \tag{3}$$

$$P_m^{x^-} > P_m^{x^+} \tag{4}$$

Here, k is a fraction threshold representing minimal required proportion of moving points over all points with the same direction. In our experiments, we set k as $\frac{2}{3}$. A point is considered as moving points if its absolute optical flow is larger than or equal to 1. The camera is supposed as horizontally stable if none of above condition is satisfied. If the camera is detected as being moved, camera motion is calculated as the average of absolute optical flows of points which moved to the same direction as the camera movement. Camera motion for vertical direction is estimated in the similar manner.

Compensation of camera motion: Flow of each SURF keypoint is compensated simply as follows:

$$f_i = f_i - df_{camera} \tag{5}$$

Here, f_i refers to flow of point i, f_{camera} refers to camera flow. d equals 1 if camera moved to positive direction or -1

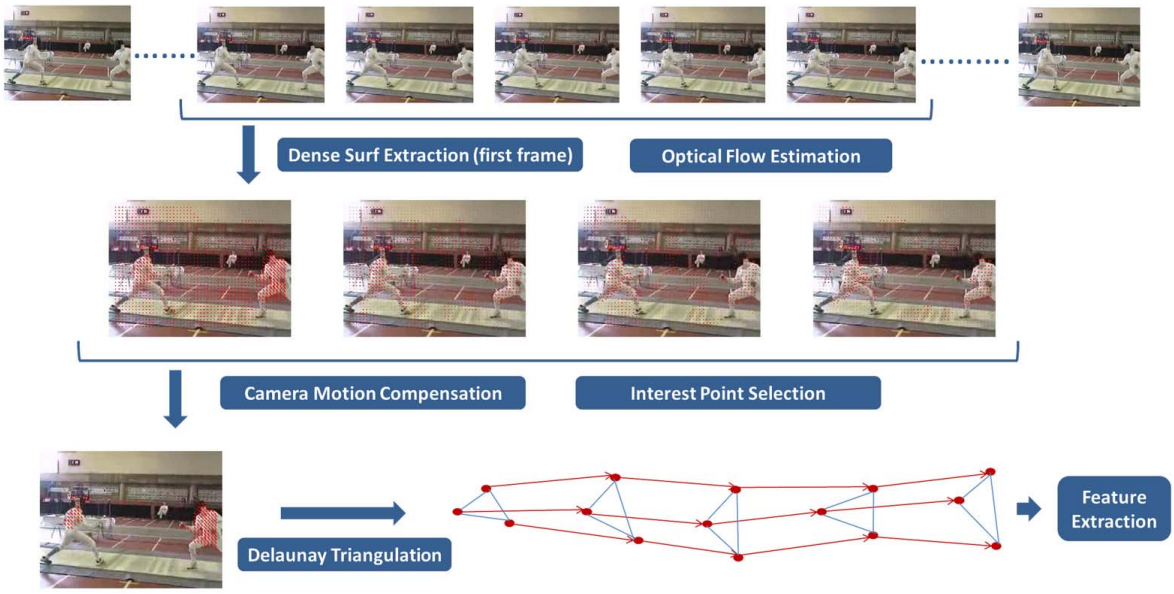

Figure 1. Overview of our method. Here we show an example of extracting spatio-temporal features from a frame set of 5 consecutive frames. The shot is from UCF-101 dataset for "Fencing" category. First, dense SURF points of the first frame are extracted and optical flow vectors of these points through the frame set are estimated (figures in the second row). Next, based on flows of SURF points, camera motion is compensated and interest points are selected (selected points are shown in the left figure of the third row). Then Delaunay Triangulation is applied to model the relation between selected points and spatio-temporal features are extracted from each obtained triangle.

if camera moved to negative direction. f_{camera} is measured separately for all considered directions (forward, backward, up and down) and compensation is operated in each of those directions. By our manner, camera motion can be compensated in most cases except for zooming. Handling this case of camera motion is one of our future works. See Fig.2 for an example result of our motion compensation method.

3.3. Selection of interest points

According to the baseline, selection of interest points is based on their optical flows between the first frame and the middle frame of the frame set. A point is believed as an interest point if its flow is larger than the pre-defined motion threshold. As a result, in case of no significant movement than the threshold from the first to the middle frame, no feature can be extracted. Moreover, their motion threshold is determined in holistic manner and fixed for every video of every type of actions. However, due to camera motion, video resolution, movements of background objects, and especially the diversity of actions as well as actors, points selected based on a constant motion threshold may not always be representative. For example, even though that significant movements are expected to be caused mainly by the actor, in the background there may be objects which move dominantly at several frames. Hence, the points belong to these objects may be mistaken as interest points. In addition, magnitude of movement may vary largely from action to action. For instance, sport activities such as jumping trampoline or swimming are supposed to cause large displacements. On the other hand, daily activities such as

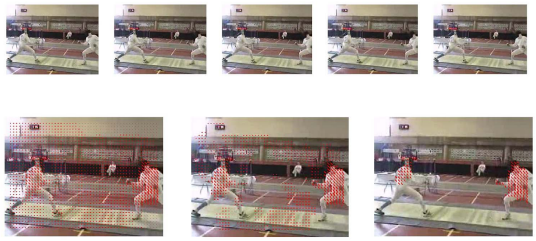

Figure 2. An example that shows efficiency of proposed method of reducing camera motion and selecting interest points. The first row presents a frame set of consecutive frames which contains camera motion. In this case, interest points are not detected according to the baseline. The second row shows optical flows of SURF points at the first frame of the frame set. The most left figure presents all SURF keypoints before the camera motion compensated. The middle figure and the right figure respectively present points determined as interest points by the baseline (with fixed motion threshold) and our method (with flexible threshold) after the camera motion compensated. This example shows that our method is not only able to reduce the effect of camera motion but also to select more representative points than the baseline.

drinking or talking in general generate small movements. We demonstrate that in order to overcome these problems, motion threshold should be flexible.

We propose to determine motion threshold flexibly and select as reliable moving points as possible. The idea is that robustness of a point should be evaluated based on comparison of its motion to motion of its fellows at the same time rather than to a fixed motion threshold. In our method, motion threshold is estimated for every frame in all directions based on flows of its SURF points. The following equation represents how we calculate motion threshold for a frame in forward direction (x^+). Thresholds for the remaining directions are similarly calculated.

$$thresh_{f_{x^+}} = aver_{f_{x^+}} + \alpha(\max_{f_{x^+}} - aver_{f_{x^+}}) \quad (6)$$

Here, $thresh_{f_{x^+}}$ means the motion threshold for frame f in x^+ direction. $aver_{f_{x^+}}$ and $max_{f_{x^+}}$ respectively refer to the average and the maximal flow magnitude at frame f in x^+ direction. The qualification that a point should satisfy to be considered as a moving point is that in at least one of four considered directions, its flow magnitude is somewhat greater than the average flow of that direction. The constant α controls that qualification. In our experiments, we set α as 0.5. Thus, the motion threshold is near to the median of the average and the max flows. However, in some case, at some frames, all objects including actor stay still, thus it is not necessary that there always must exist moving points. We suppose that nothing in a frame moved if all its thresholds are smaller than 1.

After determining which points are moving points, instead of simply taking all of them like Noguchi *et al*., we aim to select as many representative points as possible. A moving point is a point that ever moved at any frame in the frame set. We postulate a hypothesis that points with more movements are more reliable and informative. For example, through the whole frame set, points moved two times are expected to be more reliable as well as representative than points moved only once. Based on this hypothesis, we propose to select points greedily based on number of times they moved through the frame set. Our algorithm of point selection is described in Algorithm 1.

Algorithm 1 Algorithm for selecting interest points

M = maximal number of movements ($M \leq N-1$)
T = total number of moving points
GS = group of selected points (initialized as empty)
for $i = M$ to 1 **do**
  GS = $|GS$, points moved i times $|$
  if $|GS| \geq \beta T$ **then**
    break;
  end if
end for
end

Following Algorithm 1, the group of selected points is only a proportion of moving points but expected to consist of most representative points. In our experiments, we set β as $\frac{1}{2}$. Fig.2 shows the effectiveness of our method of selecting interest points over the baseline.

3.4. Descriptorization of ST features

After selecting interest points, following the baseline, we apply Delaunay triangulation to form triples of them. One ST feature can be obtained from each triple. Our proposed feature extracted from a triple is constructed based on following descriptors. We classify them to *spatial descriptor* which represents static visual features of points, *temporal descriptor* which presents movements of points through the frame set and *spatio-temporal descriptor* which characterizes trajectory-based visual features of points or group of points. Below we describe in detail each descriptor.

Spatial Descriptors. To form spatial descriptor, we combine SURF descriptors of three points of the triple at the first frame. SURF points are extracted with subregions of 3 by 3 pixels, Haar filters of 4 by 4 pixels and 4 subregions. Thus we obtain a 64-dimension SURF descriptor for each point [23]. However, concatenating SURF descriptors of three points forms a high-dimensional descriptor ($3 \times 64 = 192$ dimension) which may consist of repeated information. Thus, we apply PCA on this descriptor to acquire a lower dimensional but more representative one, and also to reduce computational cost. We denote this dimension reduced descriptor as PSURF. In our experiments, PSURF is a 96-dimension vector.

Temporal Descriptors. We propose to extract following two temporal features:

(1) **A histogram of Optical Flow (HOOF)**. $3(N-1)$ flow vectors of three points are binned to a B_o-bin histogram. Following [1], each flow vector is binned according to its primary angle from the horizontal axis and weighted according to its magnitude. That means, a flow vector $v = [x, y]$ with its angle $\theta = tan^{-1}(y)$ in the range:

$$-\frac{\pi}{2} + \pi\frac{b-1}{B_o} \leq \theta < -\frac{\pi}{2} + \pi\frac{b}{B_o} \quad (7)$$

will contribute by $\sqrt{x^2+y^2}$ to the sum in bin b. Finally, the histogram is normalized to sum up to 1.

(2) **A Histogram of Direction of Flows (HDF)**. Following Noguchi *et al*., we binned flow vectors of three points within the triple according to their direction. However, in [14], one histogram is calculated for each point. More specifically, $(N-1)$ flow vectors of a point are binned to a 5-bin histogram which represents 5 states: moving forward, backward, up, down and staying still (every flow equals to 0). Since the value of N is set to be small ($N = 5$ in their experiments), many bins become zeros. This makes

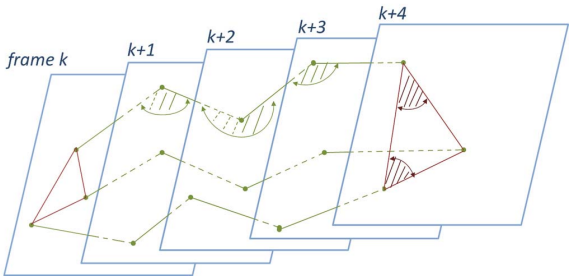

Figure 3. Illustration of proposed spatio-temporal features. We additionally explore characteristics of interest points by exploiting angles of triangles formed by them (red ones) and angles shaped by consecutive trajectories of them (green ones). We show here an example of trajectories of grouped interest points in a frame set of 5 frames. 2×5 smallest angles of triangles are binned to obtain a HAT and 3×3 trajectory based angles are binned to obtain a HAF following proposed method described in Section 3.4.

the feature be not so informative and have small effectiveness on action discrimination. Here we propose to bin all flow vectors of three points ($3(N-1)$ flows) into 4 bins: moving forward or horizontally still ($|f_{x^+}| \geq 0$), backward ($|f_{x^-}| \geq 0$), up or vertically still ($|f_{y^+}| \geq 0$), and down ($|f_{y^-}| \geq 0$). Similarly to HOOF, this histogram is also weighted by flow magnitude and normalized to sum up to 1.

Spatio-temporal Descriptors. We propose to generate the following three descriptors. The first two represent visual characteristics of triangles through the frame set. The last one descriptorizes the shape of trajectories. The last two are newly introduced by us. Refer to Fig.3 for illustration of these proposed two features.

(1) **Areas of Triangle (AT)**: Following Noguchi *et al.*' work, the area of the triangles at all frames is calculated then concatenated and normalized to form a N-dimension descriptor.

(2) **A Histogram of Angles of Triangle (HAT)**. To better explore the shape characteristics of obtained triangles, we propose to investigate their angles by binning them based on their magnitude. Here, we consider only two angles since given the degrees of any two out of three angles, it is sufficient to characterize the shape of a triangle. Using two optional angles is not preferred here since they may be not representative for their triangle. Thus, one can consider use two largest or two smallest angles. However, two largest angles can range from 0^o to 180^o while two smallest angles range only from 0^o to 90^o. Hence, selecting two smallest angles makes it easier to define histogram bin. Moreover, it can not happen that both of two smallest angles are larger than 60^o. Based on these observations, we propose to set up histogram bin as follows: for $\theta > 60^o$, the histogram bin is of size 30, otherwise, the histogram bin is of size 15. In this manner, 2 smallest angles are binned to 5 bins: [0-15], [15-30], [30-45], [45-60], [60-90]. Each angle is weighted by sum of magnitude of its two edges. The histogram is also normalized to sum up to 1.

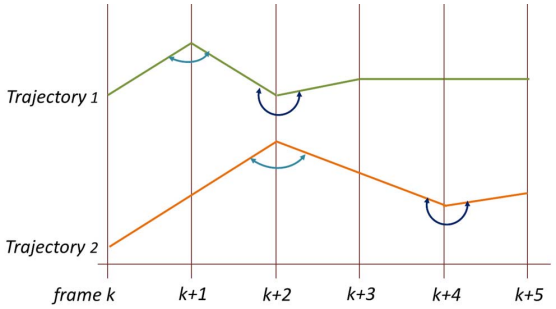

Figure 4. An example that illustrates the effect of variety in velocity on action recognition and the efficiency of our proposed method. We show trajectories of points which belong to two actors performing the same action in 6 consecutive frames. We assume that the actors move in similar way but at different speed. As shown here, Trajectory 1 which corresponds to faster actor and Trajectory 2 which belongs to lower actor only match at the first interval (from frame k to $k+1$), thus trajectory based descriptors become nearly totally different. On the other hand, according to our method, exploiting angles shaped by trajectories help to find out more the similarity between these two trajectories. The similar angles (marked by same color) can be binned to the same bin, hence this angle based descriptor can be expected to reduce the effect of diversity in velocity of different actors.

A Histogram of Angles of Flows (HAF). To exploit trajectories of interest points for modelling the action, some work straightly employ them as descriptors [2]. However, this approach suffers from the problem that trajectories may vary largely due to the velocity of the actor. To reduce the effect of the variety in velocity, we propose to extract features based on angles shaped by trajectories. These angles are supposed to be more informative than trajectories themselves (See Fig.4). The angles are binned by the same method as shown in Eq. 7. Number of histogram bin for HAF is denoted as B_a.

Finally all above descriptors are concatenated to form our ST feature which has 96 (PSURF) + B_o (HOOF) + 4 (HDF) + N (AT) + 5 (HAT) + B_a (HAF) = 105+B_o+N+B_a dimension. In our experiment, we set $N = 5$, $B_o = B_a = 4$, thus we obtain a 118-dimension ST descriptor.

4. Experiments and Results

We conduct various experiments on UCF-101 dataset [24] which is an action recognition data set of realistic action videos, collected from YouTube, having 101

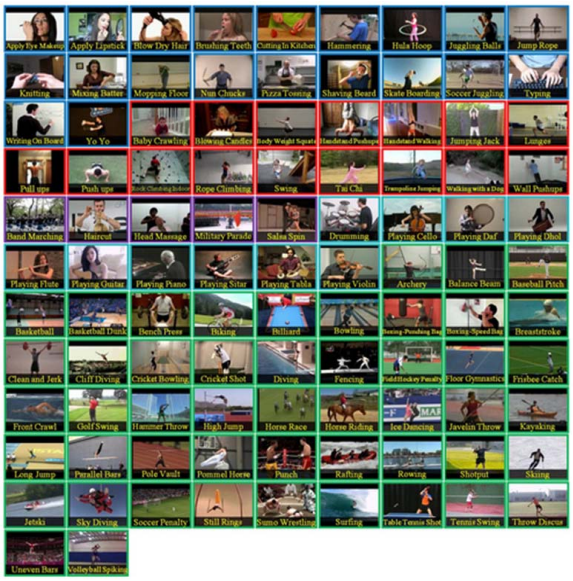

Figure 5. Thumbnails of UCF-101. This dataset consists of various action categories including sport activities such as "Basketball Shooting" or "Biking" and daily activies such as "Blow Dry Hair" or "Brush Teeth". The action categories are divided into five types: 1)Human-Object Interaction 2) Body-Motion Only 3) Human-Human Interaction 4) Playing Musical Instruments 5) Sports.

action categories. This dataset contains 13320 videos from 101 action categories which are grouped into 25 groups, where each group can consist of 4-7 videos of an action. The videos from the same group may share some common features, such as similar background, similar viewpoint. Fig.5 shows thumbnails of all action categories in this dataset.

We follow competition evaluation set up as suggested in the workshop page [2]. We adopt the provided three standard train/test splits to evaluate our results. In each split, clips from 7 of the 25 groups are used as test samples, and the rest for training. The result of each experiment reported here is calculated as the mean of average accuracies over the three provided test splits. We train multiclass SVMs [25] to perform action recognition. The results of our experiments are shown in Table 1.

To validate the enhancement of proposed feature over the baseline [14], we conduct experiments with ST feature proposed by the baseline and our proposed feature with Fisher encoding method. Since following Noguchi *et al*. [14], no points are selected if the whole video contains camera motion, feature extraction was failed for nearly one tenth of the dataset. We propose to improve their method of point selection as well as feature extraction and significantly boost the overall precision as shown in Table 1. The results demonstrate that, our method could select more representative points and also explore better the visual characteristics of them.

We also compare our results to the result reported in [24]. According to [24], their result is provided by the state of the arts for action recognition [26] using standard BoV. Applying BoV encoding on our proposed feature, we obtain 40.1% precision. Note that [24] follows the old experimental set up, that is "Leave One Group Out Cross Validation" which will lead to 25 cross-validations. Our results show the powerfulness of our proposed feature since we obtained comparable results to the state of the art although our evaluation set up is supposed to be more challenging. Our results also prove the efficiency of Fisher encoding technique on action recognition over traditional BoV technique as the precision is boosted 20% by using Fisher encoding.

Method	Average Precision (AP)
[24] (BoV)	44.5%
Our (BoV)	40.1%
[14] (Fisher)	38.2%
Our (Fisher)	**60.1%**

Table 1. Experimental results on UCF-101.

5. Conclusions

In this paper, we propose a method of extracting spatio-temporal features from videos which is able to efficently select interest points and descriptorize their features. The experimental results show significant improvement of our method over the baseline for action recognition task. In future work, we want to introduce recent approach of compensating camera motion in order to handle more complicated cases such as zooming.

References

[1] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In *Proc. of IEEE Computer Vision and Pattern Recognition*, pages 1932–1939, 2009. 1, 5

[2] H. Wang, A. Klaser, C. Schmid, and C-L. Liu. Dense trajectories and motion boundary descriptors for action recognition. *International Journal of Computer Vision*, 103(1):60–79, 2013. 1, 2, 6

[3] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *Proc. of*

[2] http://crcv.ucf.edu/ICCV13-Action-Workshop/

IEEE International Conference on Computer Vision, 2009. 1

[4] S. Maji, L. Bourdev, and J. Malik. Action recognition from a distributed representation of pose and appearance. In *Proc. of IEEE Computer Vision and Pattern Recognition*, 2011. 1

[5] Y. Bangpeng and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *Proc. of IEEE Computer Vision and Pattern Recognition*, pages 17–24, 2010. 1

[6] N. I. Cinbis and S. Sclaroff. Object, scene and actions: Combining multiple features for human action recognition. In *Proc. of European Conference on Computer Vision*, pages 494–507, 2010. 1, 2

[7] P. Dollar, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *Proc. of Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 65–72, 2005. 1, 2

[8] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *Proc. of IEEE Computer Vision and Pattern Recognition*, 2008. 1, 2

[9] M. Jain, H. Jegou, and P. Bouthemy. Better exploiting motion for better action recognition. In *Proc. of IEEE Computer Vision and Pattern Recognition*, 2013. 1, 2

[10] T. Brox, C. Bregler, and J. Malik. Large displacement optical flow. In *Proc. of IEEE Computer Vision and Pattern Recognition*, pages 41–48, 2009. 1, 3

[11] F. V. Jensen, H. I. Christensen, and J. Nielsen. Bayesian methods for interpretation and control in multi-agent vision systems. In *Proc. of SPIE 1708, Applications of Artificial Intelligence X: Machine Vision and Robotics*, pages 536–548, 1994. 1

[12] E. Nowak, F. Jurie, W. Triggs, and M. Vision. Sampling strategies for bag-of-features image classification. In *Proc. of European Conference on Computer Vision*, pages IV:490–503, 2006. 1

[13] G. Willems, T. Tuytelaars, and L.V. Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. In *Proc. of European Conference on Computer Vision*, pages 650–663, 2008. 1, 2

[14] A. Noguchi and K. Yanai. A surf-based spatio-temporal feature for feature-fusion-based action recognition. In *ECCV WS on Human Motion: Understanding, Modeling, Capture and Animation*, 2010. 1, 2, 3, 5, 7

[15] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *Proc. of IEEE Computer Vision and Pattern Recognition*, pages 1–8, 2007. 2

[16] I. Atmosukarto, B. Ghanem, and N. Ahuja. Trajectory-based fisher kernel representation for action recognition in videos. In *Proc. of IAPR International Conference on Pattern Recognition*, pages 3333–3336, 2012. 2

[17] I. Laptev and T. Lindeberg. Local descriptors for spatio-temporal recognition. In *Proc. of IEEE International Conference on Computer Vision*, 2003. 2

[18] P. Scovanner, S. Ali, and M. Shah. A 3-dimensional sift descriptor and its application to action recognition. In *Proc. of ACM International Conference Multimedia*, pages 357–360, 2007. 2

[19] A. Kläser, M. Marszałek, and C. Schmid. A spatio-temporal descriptor based on 3d-gradients. In *Proc. of British Machine Vision Conference*, pages 995–1004, 2008. 2

[20] L. Yeffet and L. Wolf. Local trinary patterns for human action recognition. In *Proc. of IEEE International Conference on Computer Vision*, pages 492–497, 2009. 2

[21] P. Matikainen, M. Hebert, and R. Sukthankar. Trajectons: Action recognition through the motion analysis of tracked features. In *ICCV Workshop on Video-Oriented Object and Event Classification*, 2009. 2

[22] W. Shandong, O. Omar, and S. Mubarak. Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories. In *Proc. of IEEE International Conference on Computer Vision*, pages 1419–1426, 2011. 2

[23] J.R.R. Uijlings, A.W.M. Smeulders, and R.J.H. Scha. Real-time visual concept classification. *IEEE Transactions on Multimedia*, 2010. 3, 5

[24] S. Khurram, R.Z. Amir, and S. Mubarak. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012. 6, 7

[25] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *The Journal of Machine Learning Research*, 6:1453–1484, 2005. 7

[26] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *Proc. of IEEE International Conference on Computer Vision*, 2009. 7