Extracting Spatio-temporal Local Features Considering Consecutiveness of Motions

Akitsugu Noguchi and Keiji Yanai

Department of Computer Science,
The University of Electro-Communications,
1-5-1 Chofugaoka, Chofu-shi, Tokyo, 182-8585, Japan
noguchi-a@mm.cs.uec.ac.jp, yanai@cs.uec.ac.jp

Abstract. Recently spatio-temporal local features have been proposed as image features to recognize events or human actions in videos. In this paper, we propose yet another local spatio-temporal feature based on the SURF detector, which is a lightweight local feature. Our method consists of two parts: extracting visual features and extracting motion features. First, we select candidate points based on the SURF detector. Next, we calculate motion features at each point with local temporal units divided in order to consider consecutiveness of motions. Since our proposed feature is intended to be robust to rotation, we rotate optical flow vectors to the main direction of extracted SURF features. In the experiments, we evaluate the proposed spatio-temporal local feature with the common dataset containing six kinds of simple human actions. As the result, the accuracy achieves 86%, which is almost equivalent to state-of-the-art. In addition, we make experiments to classify large amounts of Web video clips downloaded from Youtube.

1 Introduction

Recently the number of videos people have and on the Web is increasing rapidly, and content-based video analysis becomes more important. For example, video summarization and content-based video retrieval help users to find videos which they want to watch efficiently.

As one of the methods for that, recently spatio-temporal local features have been proposed as image features to recognize events or human actions in videos. Local features are commonly used for object recognition because of its robustness about noise, rotation and occlusion. Recently this idea has been imported to event and action recognition for video. Video analysis with spatio-temporal features is new, and has not been explored much yet. Then, in this paper, we propose yet another spatio-temporal feature based on the SURF local feature. The existing methods of extraction features from videos are classified into two types. The first one is extracting global features from a whole video. The second one is extracting many local spatio-temporal features from a video. In this paper, we focus on the second type of methods based on spatio-temporal features.

To extract spatio-temporal feature, local cuboid is one of the common methods. However, it is difficult to decide cuboid size and features extracted from

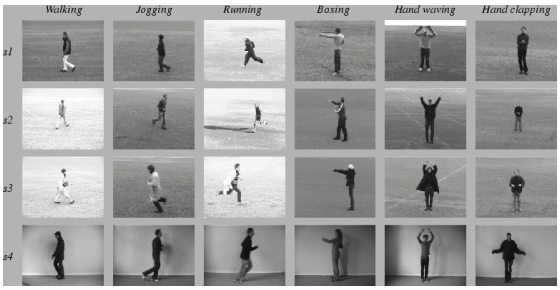

Fig. 1. KTH dataset.

cuboid. Dollar et al.[1] and Laptev et al.[2] proposed extracting Histogram of Gradient (HoG) and Histogram of Flow (HoF) from a cuboid, respectively. Extracting such features from a whole cuboid is costly in terms of computation and is not robust to noise generally.

In this paper, we detect spatio-temporally interest points and extract local pattern around them as features by extending the SURF method. This proposed method is more simple, fast and efficient method to extract spatio-temporal features than the existing ones.

In the experiment, we classify simple human motion. We use KTH dataset (Figure 1), which is a standard dataset for evaluation of human action recognition methods. This dataset contains six kinds of simple human primitive actions: "walking", "running", "jogging", "boxing", "hand waving" and "hand clapping". This dataset assumes that "each video contains only single human and action", and "no camera motion". As the result of classification experiments, we obtain the 86% classification rate. As an additional experiment, we classified shots of Web videos which are 100 soccer videos downloaded from Youtube.

In the rest of this paper, we describe related work in Section 2. Then we explain the proposed method in Section 3. Section 4 describes the experimental results. Finally we conclude this paper in Section 5.

2 Related Work

The existing methods of extraction features from videos can be classified into two types. The first one is tracking major parts of human bodies and extracting features from their regions. However, this method assumes that tracking and detection of body parts are almost successful. This assumption is sometimes difficult.

The other one is sampling many local cubic spatio-temporal regions, which is called "cuboid", from a video, and extracting features from cuboids. In this paper, we focus on this second type of methods based on spatio-temporal features.

Dollar et al. proposed the method to detect local cuboids to apply 2-D Gaussian kernels in the spatial space and 1-D Gabor filters for the temporal direc-

Fig. 2. Detected interest points by the SURF.

tion [1], and they generated video visual words by vector-quantizing local cuboids in the same way as bag-of-visual-words for object recognition [3].

Laptev et al proposed STIP (Spatio-Time Interest Points) [2] as a method to detect cuboids. This method can be regarded as an extension of Harris detector. They extracted Histogram of Gradient (HoG) and Histogram of Flow (HoF) from detected cuboids as features.

Alireza et al. proposed to extract low-level optical flows from cuboids and select good features from them with boosting to improve accuracy of classification [4].

However, computational cost of extracting features from cuboids by the methods described above is relatively high. In addition, it is difficult to decide the proper size of cuboid. To overcome these problem, in this paper, we propose to detect interest points using SURF [5] and Lucas-Kanade optical flow detection methods [6] both of which are very fast detectors and extract features by tracking interest points instead of cuboids. Since we do not use cuboids, the proposed method is more simple, fast and efficient method to extract spatio-temporal features than the existing ones.

3 Proposed Method

Our proposed method consists of four steps. In the first step, we detect interest points and extract SURF features for the detected points employing the SURF (Speeded-Up Robust Feature) [5] from the frame images which are extracted from a given video every N frames. Extracted SURF descriptors represent local appearances around interest points. Figure 2 shows that extracted interest points by the SURF, which are candidate points for tracking. In the second step, we estimate the degree of motion for each candidate points based on optical flows computed by the Lucas-Kanade [6], and select points having motion from the candidate points. This is because interest points without motion are not suitable for the points from which spatio-temporal features are extracted. In the third step, we track each tracking point locally in the temporal direction and extract motion features. In the forth step, we generate spatio-temporal features by combining SURF features and motion features for the points in the third step.

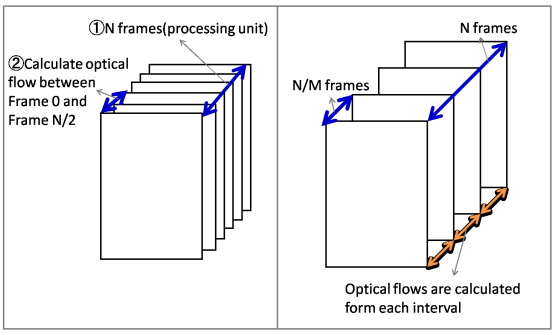

Fig. 3. Selecting frames from which optical flows are extracted (left). Extracting optical flows from the selected frames (right).

3.1 Extraction of Appearance Features

In the proposed method, we extract both local appearance features and local motion feature, and combine them into local spatio-temporal features. As local appearance features, we use the SURF descriptor [5].

The SURF is a method to extract and describe local features from one still image. Although its function is the same as SIFT [7], its processing is much lighter and faster than SIFT. The SURF method consists of two steps: detector and descriptor. In the part of the SURF detector, it selects interest points based on the Hessian matrix. In the part of the SURF descriptor, it describes local patterns around detected points with 64-d vectors per point based on the Haar-like wavelet. Refer to [5] for the detail. We obtain SURF vectors the number of which is the same as the number of the interest points. However, the SURF vectors used as actual descriptor of a video are selected in the next step.

3.2 Selection of Motion Points

In this step, we select in-motion points from all the points detected by the SURF detector by optical flow analysis.

As mentioned before, we apply the SURF detector every N frames. Then, we calculate optical flows between the first frame and the $N/2$-th frame by Lukas-Kanade optical flow detector [6] as shown in the left side of Figure 3, and select the points where optical flows are detected among the points extracted by the SURF detector. We call such points as "motion points". In the proposed method, we extract both spatially local appearance features and temporally local motion features for each motion point.

3.3 Extraction of Motion Features

In the third step, we extract optical flows to generate motion features from $M-1$ intervals among the N frames which is a unit of motion processing, after picking

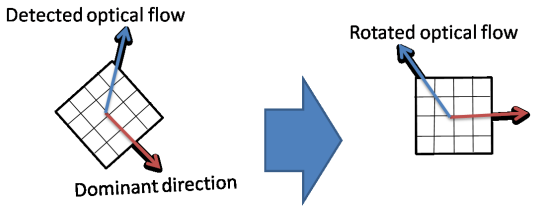

Fig. 4. Normalizing the direction of an optical flow by rotating it based on the dominant direction detected by the SURF detector.

up M frames out of N frames (M should be a factor of N). As shown in the right side of Figure 3, we calculate optical flows from $M-1$ consecutive intervals at each motion point in order to consider consecutiveness of motions. In case that M is 1, we can extract detailed motions. On the other hand, In case that M equals to N, motion information becomes condensed. In the experiment, we set both N and M as 5.

As representation of motion features, we generate a 5-d vector for each interval of each motion point from the motion matrix estimated by the Lucas-Kanade method [6]. The 5-d vector consists of x^+, x^-, y^+, y^- and no optical flow x^0, where x^+ means the degree of the positive elements along x-axis and x^- means the degree of the negative elements along x-axis. The motion feature for each interval is normalized so that the summation of all the elements equals to 1. We combine M 5-d vectors extracted from $M-1$ intervals into one motion vectors for each motion points, and totally the dimension of motion feature becomes $(M-1) \times 5$.

We hope that this feature is robust about rotation. The same feature should be extracted from "walk to right" and "walk to left", since our objective is proposing spatio-temporal features to categorize actions ignoring the directions of actions. To this end, in this paper, we propose to rotate optical flows along the dominant direction of visual features to normalize their direction. Figure 4 shows the rotation of an optical flow.

The rotated optical flow vector (x, y) are represented as follows:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \tag{1}$$

where (x_0, y_0) is the original optical flow vector for the motion point, and θ is the dominant direction of the SURF descriptor at the motion point.

3.4 Generation of Local Spatio-Temporal Features

In the final step, in the proposed method, we extract both local appearance features and local motion features, and we combine local appearance features extracted in the first step and local motion features extracted in the third step into local spatio-temporal features.

The SURF-based appearance feature is represented by a 64-d vector, and the motion feature is represented by a $(M-1) \times 5$-d vector. After weighting the motion vector with w, we concatenate both vectors into in one $(64+(M-1)\times 5)$-d vector.

In the experiment, we set 5 to both M and N, and totally the dimension of the final feature vector becomes 84. In the experiment, we explored the optimal weight. As the result, we found that 2.5 is optimal for w.

4 Experimental Results

We made experiments to evaluate the proposed feature by classifying Web videos as well as simple human actions. In this section, we describe classification methods, datasets and results.

4.1 Action Recognition

Dollar et al.[1] classified human action employing bag-of-video-words. Bag-of-video-words (BoVW) is an extension of bag-of-feature (BoF) for action recognition. Following this, we generate bag-of-video-words from the proposed local spatio-temporal features, and classify human action by a support vector machine (SVM) with a RBF kernel.

First, we extract local spatio-temporal features proposed in this paper from training video data and generate a codebook by k-means clustering from all of the extracted features. Then, a BoVW vector is generated based on the codebook for each training video, and we train a SVM with the generated BoVW vectors. Next, each test video is also converted into a BoVW vector based on the precomputed codebook, and we classify test videos with the trained SVM.

As data set, we use the KTH dataset which is commonly used for benchmark test of spatio-temporal features. This dataset contains six kinds of primitive motions such as "walking", "running", "jogging", "boxing", "hand waving" and "hand clapping". This dataset assumes that "there is no camera motion" and "each video contain only one human and motion". At each motion, 25 individuals engaged 4 times, wearing different clothing. So each motion contains 100 videos. In the experiment, we did a multi-class classification with 5-fold cross validation employing the 1-vs-rest strategy. Note that the average length of videos in the KTH dataset is about 20 second long, and we extracted about 4000 features from each video.

First, we explored optimal parameters of the motion weight w and the codebook size k. Figure 5 shows that classification rates of the six actions and their average in case of changing the motion weight w with 1, 2, 2.5 and 3. Figure 6 shows results in case of changing the codebook size k with 700, 1500 and 2500. These results indicate that the case of $w = 2.5$ and $k = 1500$ performed well. We used this setting for all the rest of the experiments,

In the next experiments, we evaluate the following four combinations of the extracted features.

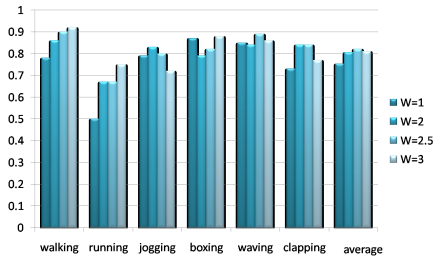

Fig. 5. Results in case of changing the motion weight w.

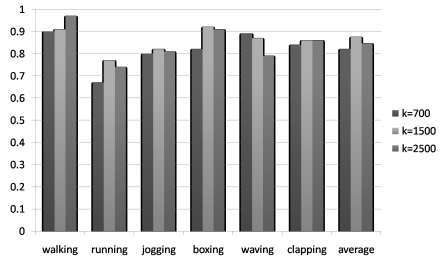

Fig. 6. Results in case of changing the codebook size k.

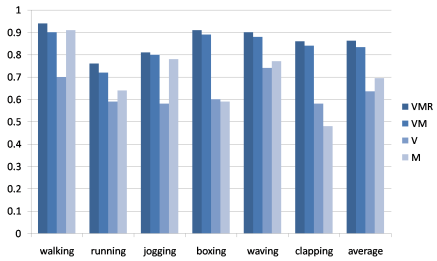

Fig. 7. Results by four types of combinations of features.

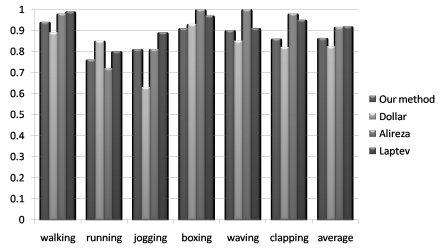

Fig. 8. Comparison with other results by the state-of-the-art methods.

1. visual appearance + motion + rotation (VMR)
2. visual appearance + motion (VM)
3. visual appearance (V)
4. motion (M)

Figure 7 shows the results of the classification rates for the six motions and their average. The average accuracy of VMR and VM both of which combine visual appearance and motion features are better than V and R both of which utilize only a single feature. VMR is better than VM, which indicates that considering rotation improved the results.

The single motion feature (M) performed well for "walking", "running" and "hand waving", while for "boxing" and "hand clapping" the results are very bad. This is because both actions of "boxing" and "hand clapping" include only horizontal motion as shown in Figure 1. Since "hand waving" contains not only horizontal motion but also small vertical motion, we can classify this action with only motion features relatively well.

On the other hand, the single visual feature (V) did not performed well for all the actions, and especially did not for "walking", "running" and "jogging" since appearances of these actions are very similar to each other.

Table 1-4 shows the confusion matrix of the classification results by four types of combinations of the features. Regarding all the combinations, the results for "walking " was good. On the other hand, it is difficult to classify "running" and

Table 1. Confusion matrix for VMR

	walking	running	jogging	boxing	waving	clapping
walking	0.94	0.02	0.03	0.01	0	0
running	0.02	0.76	0.22	0	0	0
jogging	0.04	0.15	0.81	0	0	0
boxing	0.01	0	0	0.91	0.02	0.07
waving	0	0	0	0.04	0.9	0.06
clapping	0	0	0	0.1	0.03	0.86

Table 2. Confusion matrix for V

	walking	running	jogging	boxing	waving	clapping
walking	0.7	0.13	0.16	0.01	0	0
running	0.1	0.59	0.21	0	0	0
jogging	0.12	0.29	0.58	0	0	0.01
boxing	0.13	0.13	0.1	0.6	0.03	0.01
waving	0.03	0.09	0.01	0.05	0.74	0.08
clapping	0.04	0.05	0.02	0.06	0.25	0.58

Table 3. Confusion matrix for M

	walking	running	jogging	boxing	waving	clapping
walking	0.91	0	0.06	0.03	0	0
running	0	0.64	0.3	0	0.02	0.04
jogging	0.04	0.13	0.78	0.02	0.03	0
boxing	0.01	0	0	0.59	0.32	0.08
waving	0	0	0.01	0.17	0.77	0.05
clapping	0	0	0	0.18	0.33	0.48

Table 4. Confusion matrix for VM

	walking	running	jogging	boxing	waving	clapping
walking	0.9	0.01	0.07	0.01	0	0
running	0.01	0.72	0.27	0	0	0
jogging	0.01	0.18	0.8	0.01	0	0
boxing	0	0	0	0.89	0	0.11
waving	0	0	0	0.06	0.88	0.06
clapping	0	0	0	0.13	0.02	0.84

"jogging" for all the combinations, because these two actions are so similar to each other that sometimes it is difficult for even human to classify.

Table 2 and Table 3 show the confusion matrices in case of only the visual appearance feature (V) and only the motion feature (M), respectively. From these tables, we found that it is difficult to classify "walking", "running" and "jogging" with only the visual appearance feature, while "'boxing', "hand waving" and "hand clapping" tend to be confused with only the motion feature.

Table 4 shows the confusion matrix in case of the visual appearance and motion feature without rotation. Compared to Table 1 (VMR), the accuracy of classification for all the action are worse. This means considering rotation contributes to improve the classification results.

Finally we compared our results to the other results by the state-of-the-art methods such as Dollar et al. [1], Alireza et al. [4] and Laptev et al. [2] as shown in Figure 8. The average classification rate by our method was 86%, one by the Dollar's method is 82.3%, one by the Alireza's method is 91.5% and one by the Laptev's method is 91.8%. Therefore, the proposed method is almost equivalent to the state-of-the-art methods.

4.2 Web Video Shot Classification

We classify Web video shots by k-means clustering to confirm efficiency of our features. Classifying Web video shots helps search video.

This experiment consists of four steps: (1) collect Web video, and divide them into shots by comparing HSV color histograms of consecutive frames, (2) extract the proposed feature from each shot, (3) build BoVW vectors and (4) cluster shots extracted from a single video with $k = 8$ or all the video with $k = 50$. In the experiment, we used 100 soccer videos collected from the Youtube.

Figure 9 shows the result of Web video shot clustering for a single video. This figure shows only 3 clusters out of 8 clusters. The cluster in the top row includes only shots taken from far places, the shots in the cluster in the middle row are taken near the field relatively, and the shots in the bottom are close-up of players.

Figure 10 shows 3 clusters out of 50 clusters as clustering results for all the video shots. Most of the shots in the cluster in the top row are taken from far

Fig. 9. Result of web video shot clustering per single video: cluster of far angle(top), near angle(middle) and closed-up person(bottom)

Fig. 10. Result of all web video shot clustering: cluster of far angle(top), near angle(middle) and noisy(bottom)

places, and the shots in the middle are taken mainly for players. On the other hand, the bottom cluster contains many noisy shots. Overall, shot clustering performed well, and it shows that the proposed feature is also effective to classify Web video.

However, in this experiment, we extracted ten thousands of features on average and 200 thousand features at most from one shot. This is because of camera motion. For shots with camera motion, all extracted interest points are detected as motion points, so that processing time becomes larger. To solve this, we need to detect the direction and speed of camera motion and compensate it for motion features. This is one of our future work.

5 Conclusion

In this paper, we proposed a yet-another spatio-temporal feature. Proposed method consists of two parts: extracting visual appearance features and extracting motion features. First, we select candidate points based on the SURF detector. Next, we calculate several motion features at each point with local temporal units divided in order to consider consecutiveness of motions. Since our proposed feature is intended to be robust to rotation, we rotate optical flow vectors to the dominant direction of extracted SURF features.

In the experiments, we evaluate the proposed spatio-temporal local feature with KTH. As the result, the accuracy achieves 86%, which is almost equivalent to state-of-the-art. In addition, we make experiments to classify large amounts of Web video clips downloaded from Youtube, and indicate the efficiency of our feature.

In future work, we can consider two ways. The first one is to improve the proposed feature to add more features, to improve feature descriptors, and to consider camera motions. The second one is to apply the proposed feature and build applications, such as content-based video retrieval, video summarization, and video surveillance system.

References

1. 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.
2. I. Laptev and T. Lindeberg. Local descriptors for spatio-temporal recognition. In *Proc.of IEEE International Conference on Computer Vision*, 2003.
3. G. Csurka, C. Bray, C. Dance, and L. Fan. Visual categorization with bags of keypoints. In *Proc. of ECCV Workshop on Statistical Learning in Computer Vision*, pages 59–74, 2004.
4. F. Alireza and M. Greg. Action recognition by learning mid-level feature. In *Proc.of IEEE Computer Vision and Pattern Recognition*, 2008.
5. B. Herbert, E. Andreas, T. Tinne, and G. Luc. Surf: Speeded up robust features. *Computer Vision and Image Understanding*, pages 346–359, 2008.
6. B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. of International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.
7. D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, pages 91–110, 2004.
8. C. Fanti and P. Perona. Hybrid models for human motion recognition. In *Proc.of IEEE Computer Vision and Pattern Recognition*, 2005.
9. C. Rao, A. Yilmaz, and M. Shah. View-invariant representation and recognition of actions. *International Journal of Computer Vision*, 50(2):203–226, 2002.
10. Y. Yacoob and M.J. Black. Parameterized modeling and recognition of activities. *Computer Vision and Image Understanding*, 72(2):203–226, 2002.
11. S.Konrad and G.Luc. Action snippets: How many frames does human action recognition require? In *Proc.of IEEE Computer Vision and Pattern Recognition*, 2008.