UEC at TRECVID 2007 High Level Feature Task

Ounan Liu, Zhiyuan Tang and Keiji Yanai

Department of Computer Science, The University of Electro-Communications, JAPAN
E-Mail: {liu-o,tou-s,yanai}@mm.cs.uec.ac.jp

ABSTRACT

In this paper, we describe our approach and results for high-level feature extraction task at TRECVID 2007. This year, we adopted late fusion of several types of features. As a first step, we extract several types of visual features and ASR texts from the given movies, and apply SVM to them independently. As the next step, we fuse these results by linear combination with weights chosen by cross validation.

1. INTRODUCTION

In TRECVID 2006, we extracted single type of feature vectors from each test images, and classified test keyframes by the machine learning method such as support vector machines (SVM). From the results, we noticed that if concepts (e.g. mountain, animal) are different, generally, the type of the suitable feature are different, and there is not a almighty feature that is effective for any kind of concepts.

For TRECVID 2007, we decided to adopt late fusion of several types of features. As a first step, we extract several types of visual features and ASR texts from the given movies, and apply SVM to them independently. As the next step, we fuse these results by linear combination with weights chosen by cross validation.

2. METHODS

We extract features from the keyframes of each shot. We use two types of visual features, color and texture, and the automatic speech recognition (ASR) texts provided by NIST. For each type of features, we make use of the SVM as a classifier. To fuse the results by different features, we compute the weighted sum of each output value of the SVM as a final result. The weights are obtained by cross validation within training data. Figure 1 shows the experiment process.

Due to time limitation, we have made only one run which utilizes fusion of SVM outputs of different kinds of features. Table 1 shows the six runs we have submitted to TRECVID 2007.

As pre-processing, we extract representative keyframes from the center of shots based on the shot boundary data provided by NIST.

2.1. Visual Features

As visual features, we use a color histogram and SIFT-based bag-of-features.

2.1.1. Color Histogram

We use a normal color histogram. We divide an RGB color space into 64 bins, and make a color distribution histogram by counting relative frequency of each bin.

2.1.2. Bag-of-Keypoints

As a method representing local texture, we use the bag-of-keypoints representation [1, 2]. By using SIFT [3] we extract hundreds of keypoints from each keyframe, and obtain 128-dimension SIFT vectors which code the local pattern of the neighborhood of the keypoints. In the training phase, we sample all the SIFT vectors extracted from all the keyframes of training data randomly, and perform the k-means clustering to obtain a codebook for vector quantization. In the experiments, we set the size of the codebook as 1500. Codewords in the codebook correspond to representative local patterns over all the keyframes.

Table 1: Overview of our approaches.

run ID		training data	feature, fusion	classifier	best	MAP
1	A_UEC_Combine_1		fusion of Run2 and Run4		0.015	0.005
2	A_UEC_bag06+07_2	2006, 2007	SIFT/Bag-of-keypoints	SVM	0.036	0.006
3	A_UEC_bag07_3	2007	SIFT/Bag-of-keypoints	SVM	0.036	0.006
4	A_UEC_Bag_4	2006	ASR text	SVM	0.030	0.004
5	A_UEC_Color_5	2007	color histogram	SVM	0.015	0.005
6	A_UEC_SPK_6	TV 2006,2007	SIFT/Bag-of-keypoints	SVM + SP kernel	0.146	0.046

"Best" means the best average precision of each run over all the features.

After a codebook is obtained, we vector-quantize SIFT vectors extracted from each keyframe into one vector. We assign all the SIFT vectors extracted from each keyframe to the nearest codewords, and build a histogram regarding the codewords as a bag-of-keypoints vector associated with a keyframe.

2.2. ASR Text

We use the automatic speech recognition (ASR) text provided by the TRECVID 2007 sponsor, NIST. We extract sentences associated with each frame from ASR test data, and build a bag-of-words vector for each frame as a textual feature vector. The bag-of-words vector is a histogram of the appearance frequency of words regarding the top 1500 frequent words over all the training ASR text data.

2.3. Classifier and Fusion

For each high-level concept, we train the SVM with the training data individually by each feature, and classify each frame image of test data. Then, we fuse the output values of SVM of the different feature by weighted linear combination. The cross validation is performed to select the best choice of the weight. We use RBF as the kernel of the SVM except RUN_6. In the RUN_6, we use the spatial pyramid kernel as the SVM kernel [4].

3. EXPERIMENTS

We made six run experiments including one fusion and five single-feature runs as shown in Table 1. Due to time limitation, we have made only one run, Run_1, which utilizes fusion of SVM outputs of the bag-of-keypoints and the ASR tests

We used both TRECVID 2006 and TRECVID 2007 training data as training data for Run_2 and Run_6, only TRECVID 2006 training data for Run_4, and only TRECVID 2007 training data for the others.

Table 2 and Figure 2 show the average precisions of our submitted six runs for each concept. In the figure, the sky blue sticks with the values show the median of the average precision of all the TRECVID 2007 runs.

As shown in Table 2, Run_6 has achieved the best mean average precision 0.046 among the 6 submitted runs. The best runs used only visual features with the spatial pyramid method [4].

Run_2 that used both TRECVID2007 and TRECVID 2006 training data was slightly better than Run_3 that used only TRECVID2007 training data. Therefore, for Run_1 we fused the result of Run_2 and the result of ASR text data (Run_4).

By comparing the results of Run_1, Run_2 and Run_4, we discuss the effect of fusion. For Sports and Mountain, the result of Run_1 is better than either of Run_2 and Run_4. So that we can conclude that fusion was effective for these two kinds. However, for the other kinds, the results of fusion were not improved. For example, the the fusion (Run_1) for Desert, Airplane, People-Marching, the result of fusion is the same as the result of Run_2. For Weather, Office, Meeting, Police, Animal, Computer_TV_Screen, Maps and Charts, the

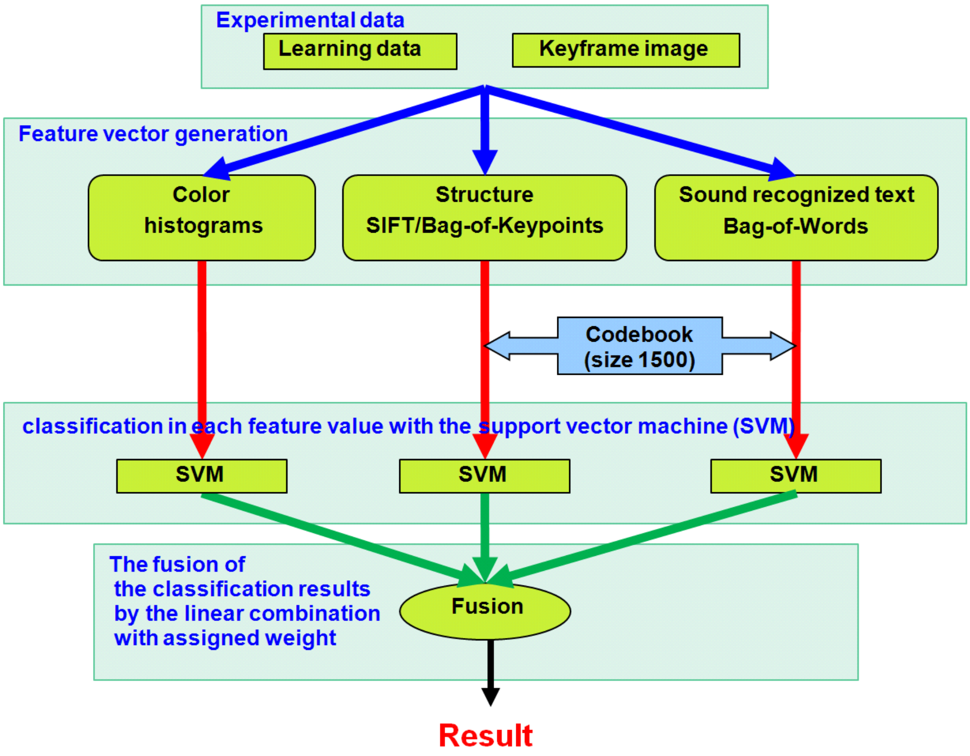
Experimental data
Learning data
Keyframe image
Feature vector generation
Color histograms
Structure SIFT/Bag-of-Keypoints
Sound recognized text Bag-of-Words
Codebook (size 1500)
classification in each feature value with the support vector machine (SVM)
SVM
SVM
SVM
The fusion of the classification results by the linear combination with assigned weight
Fusion
Result

Figure 1: Overview of our experiment processes.

average precision of Run_1 were degraded from the better results of either Run_2 or Run_4. Regarding the mean average precision (MAP), Run_1 was inferior to Run_2.

Generally we did not achieve the higher precision with fusion according to the result of Run_1. The result of all of our runs except Run_6 did not reach the median of all the TRECVID2007 runs. We should have fused Run_6 and ASR text data, but unfortunately we had no time to do that.

4. CONCLUSIONS

In the high-level feature extraction task of TRECVID 2007, we focused on late fusion of SVM outputs of different type of features. However, we were not able to improve the mean average precision by fusion. For a future work, we will estimate the optimal weights again and devise to fuse much more kinds of features.

5. REFERENCES

[1] J. Sivic and A Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proc. of IEEE International Conference on Computer Vision*, pages 1470–1477, 2003.

[2] 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 1–22, 2004.

[3] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[4] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proc. of IEEE Computer Vision and Pattern Recognition*, pages 2169–2178, 2006.

Table 2: The results of our submitted 6 runs for each concept.

no.	concepts	Run1	Run2	Run3	Run4	Run5	Run6	*median*	*best*
1	Sports	0.003	0.001	0.001	0.002	0.003	0.003	0.028	0.144
3	Weather	0.001	0.002	0.000	0.001	0.001	0.000	0.002	0.062
5	Office	0.004	0.013	0.012	0.003	0.004	0.052	0.061	0.222
6	Meeting	0.015	0.036	0.025	0.006	0.015	0.063	0.053	0.279
10	Desert	0.004	0.004	0.001	0.002	0.004	0.019	0.008	0.155
12	Mountain	0.009	0.004	0.001	0.001	0.009	0.051	0.030	0.120
17	Water	0.007	0.007	0.000	0.010	0.007	0.146	0.167	0.374
23	Police	0.004	0.009	0.002	0.001	0.004	0.008	0.003	0.046
24	Military	0.002	0.000	0.000	0.000	0.002	0.001	0.006	0.081
26	Animal	0.010	0.019	0.002	0.003	0.010	0.091	0.074	0.249
27	Computer	0.003	0.007	0.010	0.004	0.003	0.043	0.051	0.209
28	Flag-Us	0.000	0.000	0.000	0.000	0.000	0.004	0.000	0.410
29	Airplane	0.004	0.004	0.005	0.000	0.004	0.028	0.022	0.226
30	Car	0.004	0.004	0.004	0.030	0.004	0.033	0.082	0.265
32	Truck	0.002	0.002	0.002	0.002	0.002	0.023	0.026	0.101
33	Boat_Ship	0.007	0.002	0.001	0.010	0.007	0.103	0.083	0.212
35	People-M	0.002	0.002	0.000	0.000	0.002	0.041	0.028	0.104
36	Explosion	0.001	0.000	0.001	0.000	0.001	0.018	0.006	0.069
38	Maps	0.001	0.003	0.000	0.003	0.001	0.112	0.035	0.236
39	Charts	0.007	0.008	0.022	0.001	0.007	0.073	0.017	0.225
	mean	0.005	0.006	0.004	0.004	0.005	0.046	0.039	0.189

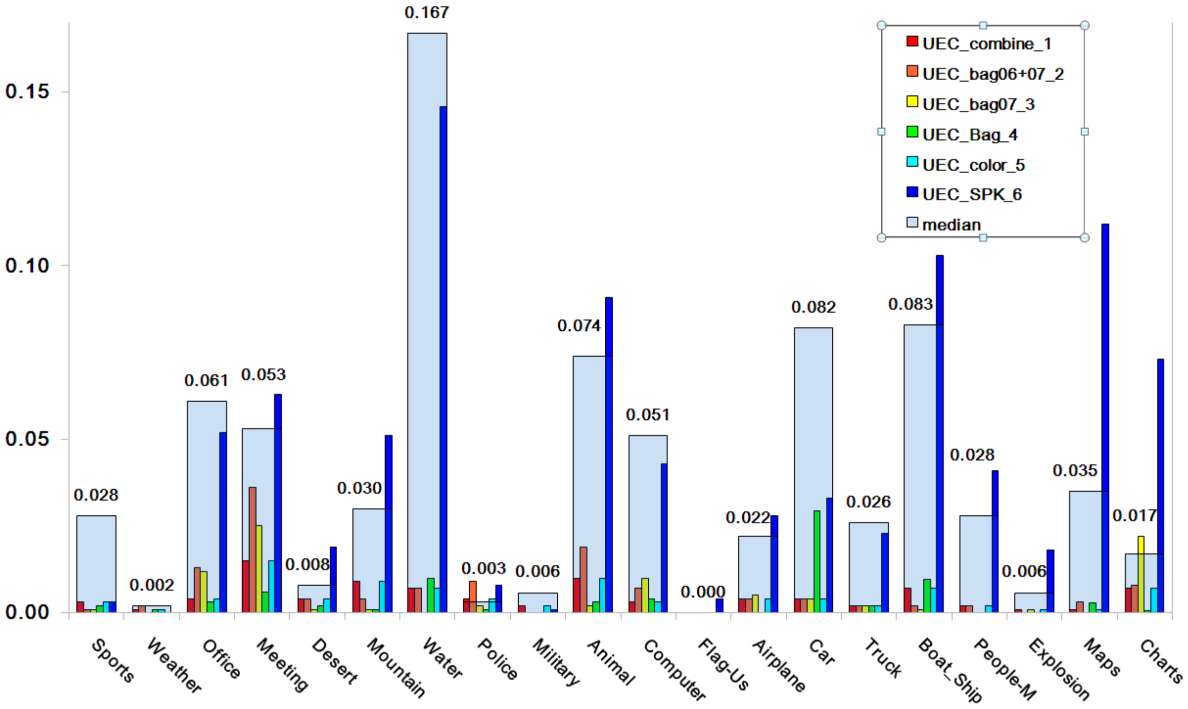

Figure 2: The results of our submitted 6 runs for each concept.