UEC at TRECVID 2009 High Level Feature Task

Zhiyuan Tang, Akitsugu Noguchi and Keiji Yanai
Department of Computer Science, The University of Electro-Communications, JAPAN
{tou-s,noguchi-a,yanai}@mm.cs.uec.ac.jp

Abstract

In this paper, we describe our approach and results for high-level feature extraction task (HLF) at TRECVID2009. This year, we focus on fusion of a number of features effectively. Color, local pattern, texture, face, motion, and text were extracted from the video data. After that, an AP-weighted fusion and Multiple Kernel Learning were applied as a fusion method to combine all these features.

Our submitted runs are as follows:

- *(Run1) UEC.APW*

  fusion of six kinds of features, color, texture, face, motion, text and Bag-of-Features (BoF) model of local pattern features, by using the AP-weighted fusion.

- *(Run2) UEC.mkl_10, (Run3) UEC.mkl_100, (Run4) UEC.mkl50_100, (Run5) UEC.mkl100_10*

  fusion of six kinds of the features which the same as Run1 by using Multiple Kernel Learning (MKL)

- *(Run6) UEC.uni_10*

  fusion of six kinds of the features with a standard Support Vector Machine (SVM) and a uniformly-combined kernel.

In all the runs, we used the same six kinds of the features, color, texture, faces, motion, local pattern features. Run1 used the AP-weighted fusion, and Run2∼Run5 used Multiple Kernel Learning with different parameters. Run6 combined the six kernels uniformly each of which corresponds to one of the six kinds of the features, and applied a standard SVM. Since MKL estimates weights to combine kernels, Run6 can be regarded as a baseline for Run2∼Run5. As a result, Run1 yielded the best performance (infAP=0.063) of these our 6 runs. MKL Runs achieved the results from 0.014 to 0.028 in terms of infAP, and outperformed the baseline (Run6). However, MKL was outperformed by the AP-weighted method which are much simpler than MKL, although we expected MKL achieved the better performance than the AP-weighted fusion.

1. Introduction

Since TRECVID [12] provides not only a large video date set but also a systematic protocol for evaluating video concept detection performance, it is appreciated by the researchers in the field of video/image recognition. Using this valuable date set, we have been testing our system in recent years.

For the HLF task in TRECVID2006, we extracted some single type visual features from the Jereme's (for example, color histogram, edge histogram, etc.), and classified test frames by the support vector machine (SVM). From the results, we realized that a certain feature cannot satisfy all the concepts. For TRECVID2007, we attempted to adopt a kind of fusion to combine some features to get a result that is effective for any kind of concept. What we did is to apply SVM to the extracted features respectively, and then to fuse these SVM classifiers by linear combination with weights selected by cross validation. This method is more effective, however it is intractable to implement when more than 3 kinds of features are extracted. For the TRECVID2008 HLF task, we still used the thought of developing a framework to fuse a number of features to get more effective performance. This time we added some new features. In addition, in-

spired by some papers [2, 16], we implemented a simple version of Adobe's [11] algorithm as a late fusion. This method can choose the suitable weights automatically no matter how many kinds of features there are.

For the TRECVID2009 HLF task, we explore the feature fusion strategy furthermore. This year we use the AP-weighted fusion [17] and Multiple Kernel Learning (MKL) [5, 14] both of which achieved the best performance in our preliminary experiments.

2. Overview

At the first stage, color, text, face, motion and local pattern features of the learning/test data are extracted from different granularity of global scale, local region and grid segmentation. Then three kinds of fusion methods are applied to model all the features, respectively.

The AP-weighted fusion is one of late fusion methods which fuse the output of SVMs for single features. MKL is a modification of SVM so as to integrate several kernels by weighted linear combination. Therefore, all the six runs can be regarded as using SVM as a classifier.

2.1. Features

Basically, we extract visual features from a keyframe of each shot and textual features from ASR texts associated with each shot.

2.1.1. Color

In the experiment, we use a normal color histogram as the color feature. The axises of RGB color space are divided in quarters and a 64-bin histogram is generated. For getting some location information, besides extracting from global scale of the image, we also tried to extract a 768 bins histogram by dividing the image to 4×3 grid segments.

2.1.2. Local pattern

We use SIFT [7] as the local pattern feature. The local patches are detected by three ways: (1) Doy (2) random sampling [10] (3) grid. The bag-of-outpoints [1] model is used to represent the whole image. The codebooks are obtained by performing the k-means clustering and the vector is generated by voting the SIFT descriptors of each image to the codebook pattern. In our experiment, the codebooks are computed for each concept respectively and every codebook size is 1000.

2.1.3. Texture

Gabor features are used as a texture feature. A Gabor texture feature represents texture patterns of local regions with several scales and orientations. In this paper, we use 24 Gabor filters with four kinds of scales and six kinds of orientations. Before applying the Gabor filters to an image, we divide an image into 3×3 or 4×4 blocks. We apply 24 Gabor filters to each block, then average filter responses within the block, and obtain a 24-dim Gabor feature vector for each block. Finally we simply concatenate all the extracted 24-dim vectors into one 216-dim or 384-dim vector for each image.

2.1.4. Motion

The Lucas-Kanade's optical flow [8] is used as our motion feature. We extract the frames 0.5 seconds before and after each keyframe and choose 500 interest points from them. The circle (360 degrees) is divided to 12 equal parts. And the motion feature is generated by voting the magnitude of the optical flow of each point to the corresponding region according to their angular degree.

2.1.5. Face

We perform a face detection by using Haar-like features [15]. The number of faces is expected to help handle with "Two People" concept.

2.1.6. Text

Automatic speech recognition (ASR) text data is provided by the sponsor every year. We use this ASR text to make a text feature. We choose 2000 representative words. Then we count their global frequency in the whole text data and the local frequency in every shot. At last a 2000 bins histogram is generated by using TF-IDF algorithm.

2.2. Fusion

This year we adopt two types of fusion methods to fuse various kinds of extracted features. One is Mul-

tiple Kernel Learning (MKL) [5], and the other is the AP-weighted fusion [17].

2.2.1. Multiple Kernel Learning

Multiple Kernel Learning (MKL) is an extension of a support vector machine (SVM). MKL treats with a combined kernel which is a weighted liner combination of several single kernels, while a normal SVM treats with only a single kernel. MKL can estimates weights for a linear combination of kernels as well as SVM parameters simultaneously in the train step. The training method of a SVM employing MKL is sometimes called as MKL-SVM. MKL-SVM is a relatively new method which was proposed in 2004 in the literature of machine learning [5], and recently MKL is applied to image recognition.

Since by assigning each image feature to one kernel MKL can estimate the weights to combine various kinds of image feature kernels into one combined kernel, we can use MKL as a feature fusion method. As mentioned before, Varma et al.[14] proposed using MKL to fuse various kinds of image features and made experiments with Caltech-101/256. Similarly, Nilsback et al.[9] applied a MKL-based feature fusion into flower image classification. On the other hand, Kumar et al.[3] used MKL to estimate combination weights of the spatial pyramid kernels (SPK)[6] with a single kind of image features. Lampert et al.[4] estimated the degree of contextual relations between objects in the setting of multiple object recognition employing MKL. In this paper we propose food image recognition employing the MKL-based feature fusion method.

In this paper, we use the multiple kernel learning (MKL) to fuse various kinds of image features. With MKL, we can train a SVM with a adaptively-weighted combined kernel which fuses different kinds of image features. The combined kernel is as follows:

$$K_{comb}(\mathbf{x},\mathbf{y}) = \sum_{j=1}^{K} \beta_j K_j(\mathbf{x},\mathbf{y}) \quad \text{with } \beta_j \geq 0,\ \sum_{j=1}^{K} \beta_j = 1. \tag{1}$$

where β_j is weights to combine sub-kernels $K_j(\mathbf{x},\mathbf{y})$. MKL can estimate optimal weights from training data.

By preparing one sub-kernel for each image features and estimating weights by the MKL method, we can obtain an optimal combined kernel. We can train a SVM with the estimated optimal combined kernel from different kinds of image features efficiently.

Sonnenburg et al.[13] proposed an efficient algorithm of MKL to estimate optimal weights and SVM parameters simultaneously by iterating training steps of a normal SVM. This implementation is available as the SHOGUN machine learning toolbox at the Web site of the first author of [13]. In the experiment, we use the MKL library included in the SHOGUN toolbox as the implementation of MKL.

2.2.2. AP-weighted fusion

The AP-weighted fusion [17] is a relatively simple fusion method. All the features are weighted in proportion to the average precision of classification results using each single feature. We train a standard SVM with a single feature and classify validation data. The average precision for each feature is estimated by cross-validation using only training data.

This is a simple method, but in our preliminary experiments the AP-weighted fusion achieved the best result among several kinds of fusion methods including MKL and Adaboost-based methods.

3. Experiments

We made 6 runs as shown in Table 1. The difference among them are only fusion methods. The features used in the experiments were the same over all the runs.

Among the six runs, one run used the AP-weighted fusion [17], four runs used Multiple Kernel Learning [5, 14], and the rest used a standard Support Vector Machine.

Run1 : Fuse features by the AP-weighted fusion. This run has achieved the best performance among the six runs.

Run2 : Fuse features by MKL. Set C as 10. C represents a soft margin parameter.

Run3 : Set C as 100.

Run4 : Set MKL_C as 50 and C as 100. MKL_C represents a parameter which adjusts sparseness of estimated weights. In Run2 and Run3, MKL_C was set as 0.

Run5 : Set MKL_C as 100 and C as 10.

Table 1. 6 runs for HLF in TRECVID2008.

Runs	Description	infAP
Run1 UEC.APW	Combine color, face, motion, text and BOF model of local pattern features AP-weighted fusion	0.063
Run 2 UEC.mkl_10	Combine color, face, motion, text and BOF model of local pattern features Multiple Kernel Learning (MKL)	0.019
Run3 UEC.mkl_100	Combine color, face, motion, text and BOF model of local pattern features Multiple Kernel Learning (MKL)	0.028
Run4 UEC.mkl50_100	Combine color, face, motion,text and BOF model of local pattern features Multiple Kernel Learning (MKL)	0.015
Run5 UEC.mkl100_10	Combine color, face and BOF. Multiple Kernel Learning (MKL)	0.014
Run6 UEC.uni_10	Combine color, face and BOF. using a standard SVM and uniformly-combined kernel	0.010

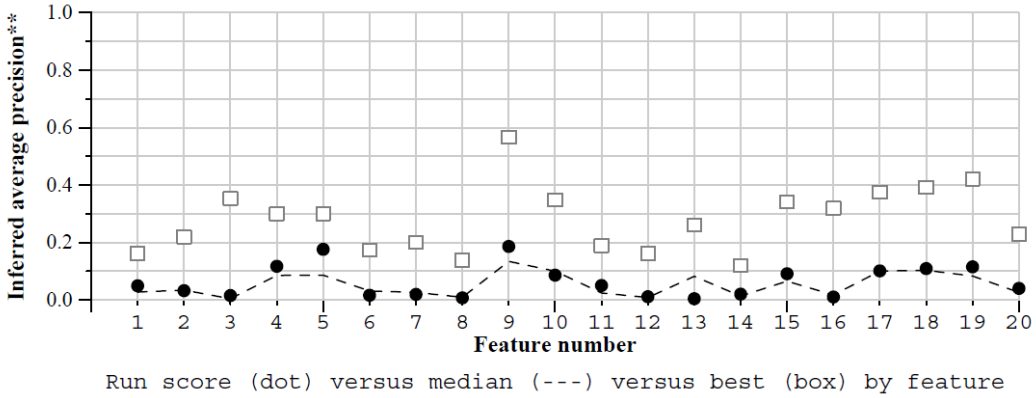

Figure 1. The comparison with the median and best results in TRECVID 2009.

Run6 : Fuse features by SVM and a uniformly-combined kernel. Set C as 10.

Our best result (run1) are compared with the best and mean result of participators as Figure 1. This results completely differed from our expectation that MKL was the best method for multi-modal fusion

4. Conclusions

In the high-level feature extraction task of TRECVID2009, we used the AP-weighted fusion and Multiple Kernel Learning to combine color, text, face, motion and local pattern features. These method can choose the suitable weight for every automatically no matter how many kinds of features there are. The results differed from our expectation that MKL was the best method to fuse many kinds of feature vectors. In the future work, we plan to explore feature fusion by MKL and other methods.

References

[1] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual Categorization with Bags of Keypoints. In *Proc. of ECCV Workshop on Statistical Learning in Computer Vision*, pages 59–74, 2004.

[2] W. Jiang, S. Chang, and A. Loui. Kernel sharing with joint boosting for multi-class concept detection. In *Proc. of CVPR Workshop on Semantic Learning Applications in Multimedia*, 2007.

[3] A. Kumar and C. Sminchisescu. Support kernel machines for object recognition. In *Proc. of IEEE International Conference on Computer Vision*, 2007.

[4] C. H. Lampert and M. B. Blaschko. A multiple kernel learning approach to joint multi-class object detection. In *Proc. of the German Association for Pattern Recognition Conference*, 2008.

[5] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72, 2004.

[6] 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.

[7] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[8] 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*, volume 3, 1981.

[9] M. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *Proc. of Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, 2008.

[10] E. Nowak, F. Jurie, and B. Triggs. Sampling Strategies for Bag-of-Features Image Classification. In *Proc. of European Conference on Computer Vision*, 2006.

[11] R. Schapire, Y. Freund, and R. Schapire. Experiments with a New Boosting Algorithm. In *Proc. of International Conference on Machine Learning*, pages 148–156, 1996.

[12] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and trecvid. In *Proc. of ACMMM WS on Multimedia Information Retrieval*, pages 321–330, 2006.

[13] S. Sonnenburg, G. Rätsch, C. Schäfer, and B. Schölkopf. Large scale multiple kernel learning. *Journal of Machine Learning Research*, 7:1531–1565, 2006.

[14] M. Varma and D. Ray. Learning the discriminative power-invariance trade-off. In *Proc. of IEEE International Conference on Computer Vision*, pages 1150–1157, 2007.

[15] P. Viola and M. Jones. Rapid Object Detection Using a Boosted Cascade of Simple Features. In *Proc. of IEEE Computer Vision and Pattern Recognition*, volume 1, 2001.

[16] D. Wang, X. Liu, L. Luo, J. Li, and B. Zhang. Video diver: generic video indexing with diverse features. In *Proc. of ACM SIGMM International Workshop on Multimedia Information Retrieval*, pages 61–70, 2007.

[17] M. Wang and X. S. Hua. Study on the combination of video concept detectors. In *Proc. of the 16th ACM international conference on Multimedia*, pages 647–650, 2008.