Universität Karlsruhe (TH) at TRECVID 2008

Hazım Kemal Ekenel, Hua Gao, Rainer Stiefelhagen

Computer Science Department, Universität Karlsruhe (TH)
Am Fasanengarten 5, Karlsruhe 76131, Germany
{ekenel,hua.gao,stiefel}@ira.uka.de
Web page: http://isl.ira.uka.de/cvhci

***Abstract*—In this paper, we present the system developed by the Interactive Systems Labs at Universität Karlsruhe for the TRECVID 2008 evaluation. It is the second time that we participate in the TRECVID evaluation this year. Last year, the main goal of our first participation was to develop a common software framework for multimedia processing and to build baseline systems for the shot boundary detection and high level feature (HLF) extraction tasks. The evaluation task for shot boundary detection was dropped this year, hence we focused our work on HLF extraction only. The color, texture and shape features were used as low-level features as before, while textual feature was not used. We used support vector machine (SVM) classifier for low-level feature classification. The parameters for SVMs were optimized in a grid-search scheme. A simple weighted sum fusion was applied to extract the high-level features, where the weights were calculated for each category and each low-level feature. More training data was used and evaluation results were improved compared to the last year's evaluation results.**

I. Introduction

It was the second time that we participated in the TRECVID evaluation this year. Last year, the main goal of our first participation was to develop a common software framework for multimedia processing and to build baseline systems for the shot boundary detection and high level feature (HLF) extraction tasks. The evaluation task for shot boundary detection was dropped this year, hence we focused our work on HLF extraction only. Unfortunately, the evaluation results on our HLF extraction system from last year were not as good as we expected, they were below the average. So, this year, we focused our work on finding implementation problems of the old system and parameter optimization for training SVM-based classifiers.

The paper is organized as follows: The old HLF detection system from the last year is described in Section II. In Section III, we explain the modifications we have done for this year's evaluation. The experimental results on the TRECVID 2007 and 2008 data are presented in Section IV. The conclusions are given in Section V.

II. Overview

Our baseline system for HLF detection was composed of several successive processing steps. In the first step, the key-frames of the master shots in the videos were represented by low-level feature vectors, which describe the color, shape and textual content of the frames. In the second step, these features were normalized and fed into a set of SVM classifies in order to determine whether a category exists in a key-frame or not. We trained SVM classifier for each low-level feature descriptor and category. In the third step, the scores that are returned by the SVM classifiers were normalized and combined to generate the final decision and the shots were ranked according to the fused scores.

Both visual cues and textual cues were used as low-level features. For the visual cues, we employed the features similar to MPEG-7 visual descriptors, including color histogram [10], grid color moments [9], color correlogram [5], co-occurrence texture [2], wavelet texture [2] and edge histogram [4]. In addition to the color-based and texture-based features, the geometric blur [1] was also used as a shape-based visual cue. Relative term frequency-based textual features were investigated as an additional descriptor which were obtained from the machine translated automatic speech recognition output. Details about the design and implementation of these features are given in [4].

We split the development data of TRECVID 2007 into three portions. The first part was used for low-level feature learning, the second was for validation of the learned classifiers and learning the high-level fusion (for learning-based fusion methods), and finally the third part was used for the validation of high-level fusion. Through permutation, a 6-fold cross validation was performed on these three portions of data. The classifiers from the fold that achieved the highest mean average precision (MAP) were chosen as the final classifiers.

We considered several high-level fusion schemes last year to combine the outputs from low-level feature classifiers. Two of which are learning-based fusion approaches that are trained with the scores returned by the low-level feature classifiers. We used neural networks and SVMs, respectively, for this kind of high-level fusion. Simple fusion rules were utilized as well to combine the low-level classifiers, which include the sum rule, min-max rule, voting rule and the weighted version of the sum rules. Correlation among the categories was also considered as a simple semantical analysis between the categories.

Last year, we submitted totally 6 runs for the final evaluation. However, all the results were below the average. The best run was generated using the fusion scheme of sum rule and the achieved infAP was 2.3%. There were several reasons for the low performance compared to other systems using the similar low-level features and classification scheme, according to our analysis in [4] two of which were the fixed assignment of weights for a certain low-level feature extractor and parameter optimization for SVMs.

III. Revised System

We investigated the old system intensively to find the problems that cause the poor performance. We started from extracting the low-level features. Since the textual cues did not help in the old system, we dropped this feature. Only the sum rule was used since it outperformed the other fusion approaches and it was also widely applied in many other systems. The major revision steps of the old system is described in the following subsections.

A. Low-level features

Last year, the color feature vector of grid color moments was calculated in HSV space and the grid size was 3×3. We followed the implementation in [2] where the grid size is 5×5 to obtain finer local color features. The first order (mean), the second (variance) and the third order (skewness) color moments were calculated in each local block in the image and the Lab color space was used instead of HSV. The dimension of this feature was $9 \times 5 \times 5 = 225$.

B. Low-level feature classification

According to [3], comparing the histogram-based feature vectors with the $L1$ metric or the χ^2 metric outperforms the $L2$ metric, because the histograms are discrete densities. We chose the $L1$ distance for its simplicity and outstanding performance in [3]. The $L1$ distance is defined as:

$$d_{L1}(x, x') = \sum_i |x_i - x'_i|,$$

which gives the Laplacian RBF kernel:

$$K_{Laplacian}(x, x') = e^{-\frac{d_{L1}(x,x')}{\sigma^2}}.$$

We trained the SVMs for the histogram-based features such as the color histogram, edge histogram and the color correlogram with the Laplacian RBF kernel, while the remaining features were trained with the Gaussian RBF kernel. The Laplacian RBF kernel was implemented as a user defined kernel in the employed SVM-light [6] implementation.

The parameters for the SVMs were optimized by conducting a coarse grid search with 3-fold cross validation.

C. Fusion

The output of the SVM-based classifiers were actually the distance of the feature vector to the learned hyper-plane. To avoid the dominance of some outputs, we normalized the distance scores with a sigmoid function [8]:

$$S_{norm} = \frac{1}{1 + e^{-S_{raw}}}.$$

Where S_{norm} is the normalized score and S_{raw} is the raw score from the SVM classifier.

We only considered the weighted sum rule for combining the low-level classifiers. In the old system, a global weight was assigned to each low-level feature, which was not reasonable because a specific feature classifier may be more suitable for some categories but not for the others. Thus we assigned weights for each feature and each category according to the average precision (AP) score obtained by the experiments on the validation data. The fused score is then computed as follows:

$$S_{fusion} = \frac{\sum_{i=1}^{N} w_i S_i}{\sum_{i=1}^{N} w_i}.$$

Where S_{fusion} is the fusion score, S_i is the score to be fused, w_i is the corresponding weight of score S_i.

D. Training data

The positive and negative training samples in the development data were quite unbalanced. Some categories, such as the concept "Flag-US" in TRECVID 2007, only contains 12 positive samples while the number of the negative samples is over 20000. We set the SVM parameter for controlling the cost margin to the number of negative samples in the training set divided by the number of positive samples [7]. However, SVM classifiers with this parameter setting return too few positive returns which leads to low recall.

To avoid this problem, we down-sampled the negative samples in the training set, while all the positive samples were kept. The down-sampling followed the method in [11]. For a given category, the number of the negative samples was limited according the number of the positive samples. The training samples were then not heavily unbalanced for some categories and less support vectors were generated during the SVM training, which accelerated both training and evaluation.

To obtain the weights for each low-level classifier and the parameters for the SVMs, we conducted a 3-fold cross validation. The development data was split into two partitions, since the sum rule fusion is not learning-based fusion, we do not have to split the data to learn the scores. $2/3$ of the stratified training samples were selected for training and the remaining samples were used as a validation set. However, all the samples were trained again using the optimized SVM parameters to provide enough positive training samples.

E. Ranking

Since we got very few positive returns last year, some negative returns were also added in the ranking list. The length of the ranking list was set according to the priori ratio of the positive samples in the development data. This was the major reason which cause the poor results of the old system. With down-sampling of the negative samples we got reasonable number of positive returns in the revised system, and the ranking lists were limited to 2000 according to the requirement for evaluation without any other limitations.

IV. Experimental Results

After conducting grid search with 3-fold cross validation, the parameters for the SVMs were coarsely optimized (with a coarse step-size because of time limit). The parameter for trading-off between training error and margin was set to 1.0, and the parameter gamma (γ) for RBF kernel was set to 0.05. We evaluated the revised system on the 2007 TRECVID testing data with ground-truth annotation provided by NIST.

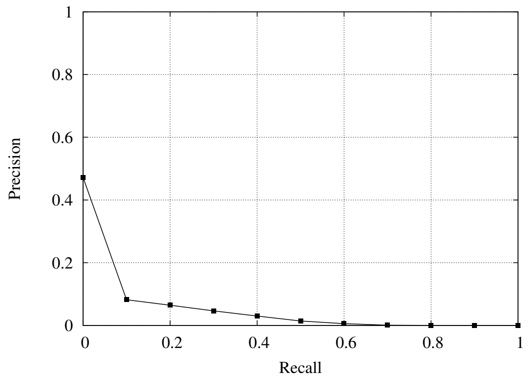

Fig. 1. Interpolated recall-precision. Estimated using 50% sample (e.g., estimated precision = 2 * actual from sample)

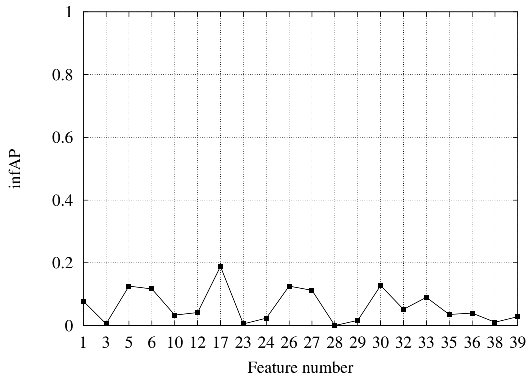

Fig. 2. The inferred average precision (infAP) scores for all evaluated categories. Estimated using 50% sample (e.g., estimated precision = 2 * actual from sample)

Run	Mean infAP	Description
UKA1	3.0%	Weighted sum fusion
UKA2	3.8%	Weighted sum fusion with unconfidence filtering,

TABLE I
HIGH-LEVEL FEATURE EXTRACTION RESULTS

Concept	UKA1	UKA2	Median	Max
Classroom	0.001	0.003	0.008	0.152
Bridge	0.006	0.010	0.004	0.117
Emergency_Vehicle	0.000	0.000	0.003	0.065
Dog	0.057	0.094	0.101	0.275
Kitchen	0.004	0.009	0.010	0.165
Airplane_flying	0.028	0.034	0.029	0.278
Two people	0.032	0.030	0.050	0.174
Bus	0.002	0.001	0.004	0.119
Driver	0.035	0.059	0.046	0.324
Cityscape	0.034	0.053	0.059	0.258
Harbor	0.001	0.005	0.008	0.182
Telephone	0.007	0.006	0.011	0.136
Street	0.055	0.080	0.113	0.413
Demonstration_Or_Protest	0.001	0.002	0.013	0.233
Hand	0.014	0.020	0.095	0.377
Mountain	0.032	0.037	0.041	0.246
Nighttime	0.129	0.130	0.102	0.323
Boat_Ship	0.100	0.113	0.093	0.394
Flower	0.048	0.062	0.058	0.161
Singing	0.025	0.014	0.014	0.258
Mean	0.030	0.038	0.043	0.232

TABLE II
RESULTS FOR EACH CONCEPT

The mean infAP score was 6.3%, which was above the average. The interpolated recall-precision diagram is shown in Figure 1 and the infAP scores for all evaluated categories are plotted in Figure 2. With the priori ratio limitation, the system got a mean infAP score of 3.9%, which means that the limitation discarded lots of hits and thus degraded the performance. The number of the positive samples also plays an important role for the final performance. For example, the group NII-ISM (National Institute of Informatics, The Institute of Statistical Mathematics) [8] used the similar low-level features and fusion scheme and got similar infAP score (6.6%) if only the development data of TRECVID 2007 is used. But the score improved to 10.1% when they trained their system using more data including TRECVID 2005, 2006 and 2007.

Last year, totally 36 concepts were used for HLF detection task, while only 20 categories were evaluated. In this year, however, 20 concepts were selected and all were evaluated with the same criteria as before. The task was more object detection oriented and 15 of the 20 features were annotated with bounding boxes in the key-frames for training. Due to scarcity of time, we did not build another system to utilize the object annotation for object detection. We evaluated our revised system on the data provided for this year and submitted two runs. However, they were quite similar runs. The second run filtered out some relatively less confident returns. The results are listed in Table I. The inferred average precisions for each concept are listed in Table II. The columns UKA1 and UKA2 list the infAP scores of our two submitted runs. The median and maximum infAP scores are listed in the fourth and fifth columns, respectively. Our results are close to median, but they are still far from the best system.

V. CONCLUSION

Several modifications were made based on the old system from last year. We evaluated the revised system on the 2007 TRECVID data with ground-truth annotation and the results were improved. From our experiments and the report from the other groups, we concluded that training with more data improves the performance. However, since the evaluation in this year is more object detection oriented, we should have utilized the provided object annotations to improve our results further.

REFERENCES

[1] A. C. Berg, T. L. Berg, and J. Malik, "Shape matching and object recognition using low distortion correspondences", *Proc. IEEE. Conf. Computer Vision and Pattern Recognition*, 2005.

[2] M. Campbell, A. Haubold, and S. Ebadollahi et al., "IBM Research TRECVID-2006 Video Retrieval System", *NIST TRECVID Workshop*, Gaithersburg, USA, Nov. 2006.

[3] O. Chapelle, P. Haffner, and V. Vapnik, "SVMs for Histogram-Based Image Classification", *IEEE Trans. on Neural Networks*, vol. 10, pp. 1055-1064, 1999.

[4] H. K. Ekenel, M. Fischer, H. Gao, K. Kilgour, J. S. Marcos, and R. Stiefelhagen, "Universität Karlsruhe (TH) at TRECVID 2007", *NIST TRECVID Workshop*, Gaithersburg, USA, Nov. 2007.

[5] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih, ”Image indexing using color correlograms”, *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 762-768, San Juan, 1997.

[6] T. Joachims, ”Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999.

[7] K. Morik, P. Brockhausen, T. Joachims, ”Combining statistical learning with a knowledge-based approach - A case study in intensive care monitoring”, *Proc.* 16^{th} *Int'l Conf. on Machine Learning*, 1999.

[8] D.-D. Le, S. Satoh, and T. Matsui, ”NII-ISM, Japan at TRECVID 2007: High level Feature Extraction”, *NIST TRECVID Workshop*, Gaithersburg, USA, Nov. 2007.

[9] M. Stricker and M. Orengo, ”Similarity of color images”, *Proc. SPIE Storage and Retrieval for Image and Video Databases*, vol. 2420, pp. 381-392, San Jose, USA, Feb, 1995.

[10] M. J. Swain and D. H. Ballard, ”Color Indexing”, *Int. Journal of Computer Vision*, vol. 7, no. 1, pp. 11-32, 1991.

[11] A. Yanagawa, S.-F. Chang, L. Kennedy, and W. Hsu, ”Columbia University's Baseline Detectors for 374 LSCOM Semantic Visual Concepts”, *Columbia University ADVENT Technical Report #222-2006-8*, March, 2007.