WHU-NERCMS at TRECVID2013:Instance Search Task

Yimin Wang, Mang Ye, Qingming Leng, Bingyue Huang,
Zheng Wang, Yuanyuan Nan, Wenhua Fang, Chao Liang
National Engineering Research Center for Multimedia Software,
School of Computer ,Wuhan Univesity,Wuhan 430079,P.R.China.
cliang@ whu.edu.cn

Abstract

This paper introduces our NERCMS team work at the automatic instance search task of TRECVID 2013. Our work is divided into three parts: feature extraction, distance measure and results combination. In feature extraction, only SIFT feature is employed, and further used to generate histogram descriptions through general BoW and BoW based on vocabulary tree methods. In the distance measure, L1 and L2 distances are adopted. Considering the availability of contextual region, a stare model is utilized to weight query images. At the final stage we use several result combination strategies to generate the final ranking lists. The details are shown as following.

1 Introduction

In TRECVID 2013 [1], we participate in the automatic instance search task, four kinds of results are submitted as shown in Table 1, and the evaluation index is mAP [2]. In the table, BoW is the method that uses general BoW description, while Voc_Tree utilizes BoW description based on vocabulary tree. "min" and "avg" refer to computing the minimum and average distances between probe shot and gallery shot representatively.

The overall process of this work can be summarized as the follows: First, we extract histogram description through general BoW and and vocabulary tree based on BoW methods; Second, L2 and L1 distance are used to compare the probe and gallery shots for the above two feature representations, and the initial sorting result is obtained. At last, several combination strategies are conducted to improve the initial result. The frame-work of ourteam is shown in Fig 1.

Table 1: INS results and descriptions for each run.

Method	MAP	Description
F_NO_NERCMS_1	0.006	BoW with 1000 visual words
F_NO_NERCMS_2	0.007	BoW_min+ Voc_Tree_min
F_NO_NERCMS_3	0.007	BoW_min+ Voc_Tree_min+Voc_Tree_avg
F_NO_NERCMS_4	0.008	BoW_min+BoW_avg+Voc_Tree_min+Voc_Tree_avg

2 Feature Extraction

This section presents our feature extraction method. A method based on sampling keyframes extraction is firstly adopted as the pretreatment of video representation, where a keyframe is taken in every five frames. In feature extraction, while the SIFT feature is the most commonly used in TRCEVID works [3], we employed dense SIFT feature to express the picture, the parameters are chosen as follows, 10 is the size of a SIFT spatial bin and the step-length is 24, finally we get a 713×128 vector to represent a keyframe.

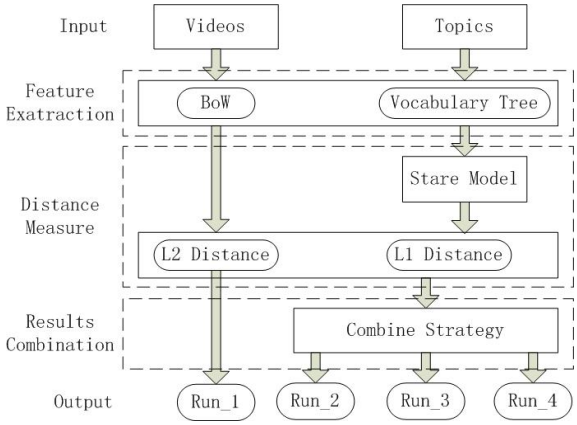

Figure 1: The frame-work of our team.

2.1 K-mean clustering based BoW representation

A general BoW method is firstly used based on SIFT feature, we divided it into three steps, the detail procedure is listed as follows:

- **Sampling** [4]: Randomly sampling 500 points in each shot; and then continue to sample to get 30000 points for one video, therefore 5 million points are extracted in the whole corpus.
- **Clustering**: We employ cluto [5] tools for clustering the points to 1000 visual words.
- **Histogram generation** : A kdtree is built for calculating the histogram based on the 1000 visual words of each keyframe.

2.2 Vocabulary tree based BoW representation

This subsection describes feature extraction details of vocabulary tree based BoW representation. Vocabulary tree [6] is efficient for reducing the computational complexity especially in massive dataset, which also can be used to extract BoW feature as follows:

- **Sampling**: 500 points are sampled randomly in each shot, and then 30000 points are sampled out for one video, finally we will get 5 million points in the whole dataset.
- **Voc_Tree_2×100**: All the selected SIFT points are divided into 100 parts in every layer of the tree, and the depth of the tree is two, so we build a 2×100 tree.
- **Voc_Tree_4×10**: All the selected SIFT points are divided into 10 parts in every layer of the tree, and the depth of the tree is four, so we build a 4×10 tree.

3 Distance Measure

Because the contextual region may provide effective information, especially for the small-scale object. For example, the sun in the sky, the sky is useful for searching the sun. The "Stare Model" [7] is used to weight every keyframe as follows:

$$w(x) = \begin{cases} 1 & if\, x \in mask \\ \frac{2}{e^{kx/diag}+1} & otherwise \end{cases} \quad (1)$$

where $w(x)$ is the weight of a pixel, $diag$ indicates the length of diagonal axis of the query image, x is the minumn distance between the point and the mask region, k is a parameter of weight adjustment, in our experiments we choose $k = 15$. If x is belong to mask region, its weight is 1. Otherwise is damping according to the rule indicated by Eq.1. With the

"stare" model, we are able to emphasis the context when the instance is small to improve the searching results.

As for the multiframes matching problem, the distance between the topics and the keyframes we have two different choices, the "min" and "avg", the minimum distance are chosen to stand for the shot distance to the topics. In distance measure, we chose L2 distance in general BoW feature, chose L1 distance in Vocabulary Tree method [8].

4 Results Combination

Considering the fact that not every method is superior to other methods, so we combined the results to improve our work. In above sections, we can get six reuslts: BoW(min,avg), Voc_Tree_2×100(min,avg), Voc_Tree_4×10(min,avg). By comparing the six kinds of results, through some simple expriments, we found that the "min" is better than the "avg", the BoW method is better than Voc_tree, and there is not much difference between the Voc_Tree_2×100and Voc_Tree_4×10, so we combine the two Voc_Tree method with a ratio of 0.5:0.5 first, so at last, we determined our integration schemes to form the final the final four runs as follows:

- **F_NO_NERCMS_1**: BoW_min, because we found the effcet of this method is not bad, we directly chose the BoW as the run 1.
- **F_NO_NERCMS_2**: BoW_min+Voc_Tree_min, because both "min" are better than "avg", in this run, we combined the distance of the BoW_min and Voc_Tree_min with a ratio of 0.5:0.5, then output the combined results as the run 2.
- **F_NO_NERCMS_3**: BoW_min+Voc_Tree_min+Voc_Tree_avg, In order to make full use of these results, we combined their distances with a ratio of 0.5:0.3:0.2 as the final distance, then output the combined results as the run 3.
- **F_NO_NERCMS_4**: BoW_min+Voc_Tree_min+Voc_Tree_avg, In this run, we didn't combine the distance, we directly conduct the four results as follows, the final result grew out of the four results interpolated in order.

5 Results and Analysis

Our final results are shown in Fig 2. By participating in the instance search task in TRECVID 2013, we have the following conclusions: (1) Only use SIFT feature can not get a good performance; (2) the Vocabulary tree can improve the searching speed; (3) Combine strategy can improve the perfomance.

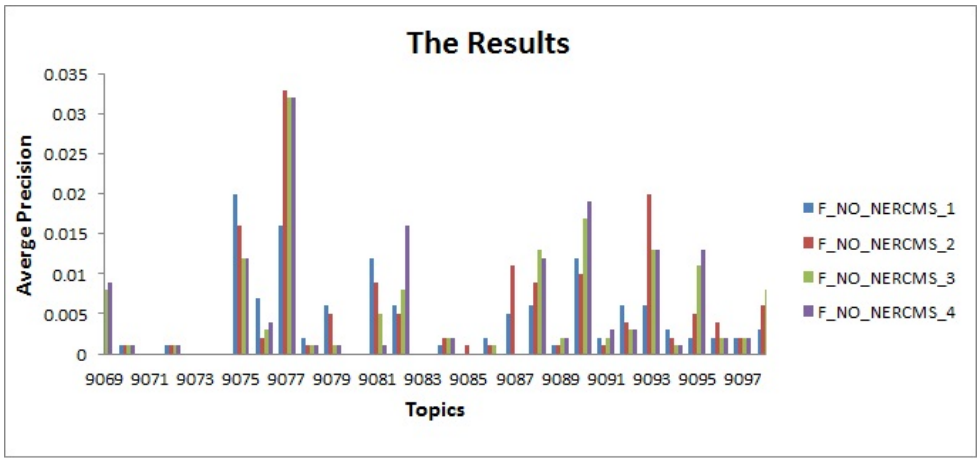

Figure 2: Our NERCMS's results

References

[1] Paul Over, George Awad, Martial Michel, Jonathan Fiscus, Greg Sanders, Wessel Kraaij, Alan F. Smeaton, and Georges Quenot, "Trecvid 2013 – an overview of the goals, tasks, data, evaluation mechanisms and metrics," in *Proceedings of TRECVID 2013*. NIST, USA, 2013.

[2] Alan F. Smeaton, Paul Over, and Wessel Kraaij, "Evaluation campaigns and trecvid," in *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, New York, NY, USA, 2006, pp. 321–330, ACM Press.

[3] Stephane Ayache and Georges Quenot, "Video Corpus Annotation using Active Learning," in *European Conference on Information Retrieval (ECIR)*, Glasgow, Scotland, mar 2008, pp. 187–198.

[4] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," *http://www.vlfeat.org*, 2008.

[5] Karypis Lab, "Cluto-software for clustering high-dimensional datasets," *http://glaros.dtc.umn.edu/gkhome/cluto*.

[6] David Nister and Henrik Stewenius, "Scalable recognition with a vocabulary tree," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*. IEEE, 2006, vol. 2, pp. 2161–2168.

[7] Wei Zhang, Chun-Chet Tan, Shi-Ai Zhu, Ting Yao, Lei Pang, and Chong-Wah Ngo, "Vireo@ trecvid 2012," .

[8] Josef Sivic and Andrew Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 1470–1477.