A Fast Table-Based Approach of Bag-of-Features for Large-Scale Image Classification

邱 盟竣[†‡]　山崎 俊彦[†]　相澤 清晴[†]
Meng-Jiun Chiou[†‡]　Toshihiko Yamasaki[†]　and　Kiyoharu Aizawa[†]

†東京大学　‡台湾国立交通大学
† The University of Tokyo　‡ National Chiao Tung University

Abstract　For large-scale image classification problems, feature extraction has been one of the most time-consuming parts. This paper introduces a table-based method of finding bag-of-features-based indexes of query pictures without feature extraction. In our experiment, we compared the proposed method with conventional feature extraction method.

1. Introduction

This paper deals with image classification and retrieval problems, especially for large-scale image categorization. For these problems, efficient processing is an important factor with respect to practicality. Feature extraction was shown to be one the most time-consuming parts in the object recognition problems [1]. To deal with this problem, this paper introduces a fast look-up table based method of finding bag-of-features-based indexes of query pictures without conducting feature extraction [2], as a starting point. The proposed look-up table consists of local patches (stored in a pixel form) and their corresponding visual words (assigned from the visual words codebook) of images. By this, the visual word is assigned to the query picture by retrieving the most similar patch in the database. In our experiment, we evaluated the proposed method by comparing it with common feature extraction methods and implementing exhaustive research to check the true performance of it.

2. Background

2.1. SIFT

In order to describe an image in a simple way, feature extraction is used in most common solutions. SIFT (Scale-Invariant Feature Transform) is used for extracting scale-invariant keypoints and computing its descriptor to form distinctive image features. In our experiment, SIFT is mainly computed by two steps: 1) keypoints detection: In our experiment, SIFT from 16×16 pixel patches are densely sampled on a grid with the step size of 8 pixels. 2) feature description: The direction and magnitude of gradient are computed for every pixel in a neighboring region of the detected keypoint. Finally, the neighboring region are divided into 16 blocks of 4×4 pixel, each with 8-bin orientation vector, which form an SIFT of 128-dimension vector.

2.2. Bag-of-Features

Bag-of-features (BoF) is a popular representation model, usually used to represent a single picture. First, feature extraction (e.g. SIFT) is implemented. Next, extracted features are used to train a visual words codebook through K-means clustering, and then the nearest visual word is assigned to the extracted features from the visual words codebook by vector quantization. All assigned visual words of a single picture form a high dimensional (which depends on number of visual words) histogram vector.

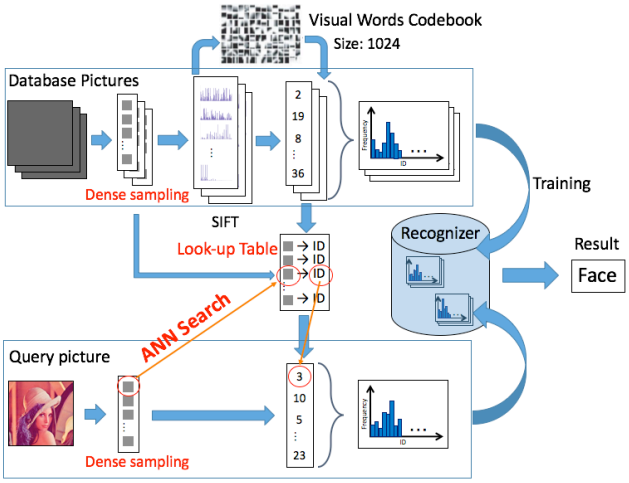


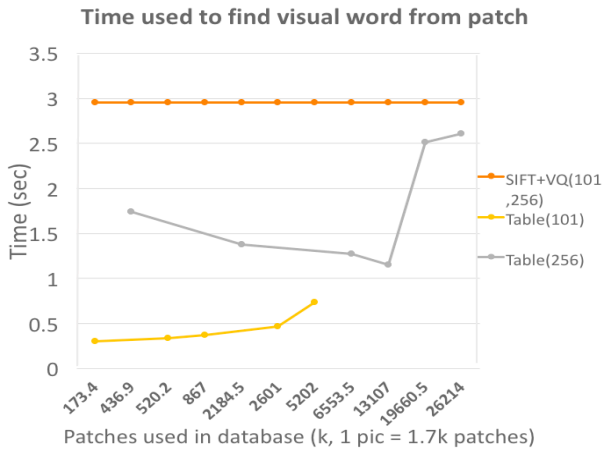
Fig.1 The overall process of proposed method

2.3. Classification

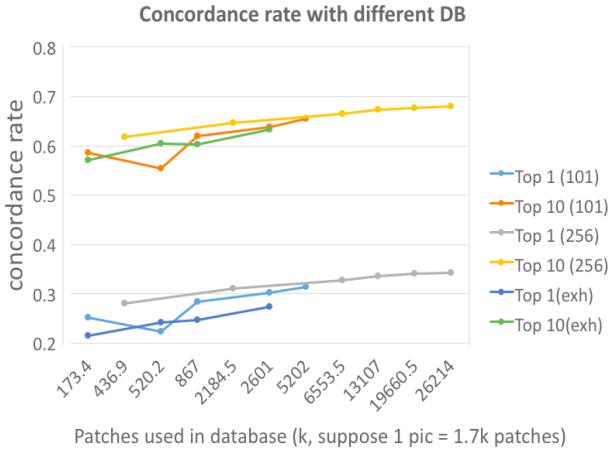
Such vectors described above can be trained through classifiers (e.g. Support Vector Machine) into a database, which can be used to predict the class of query pictures. Also, query pictures are in the same representation as database pictures (i.e. histogram vectors here). In this step, the accuracy of prediction will be evaluated across different sizes of training and table-forming database.

3. Bag-of-Features with Proposed Method

To deal with the problem of a large number of query tables, we applied look-up table to BoF. Look-up table is composed of: 1) 16×16 pixel patches of database pictures densely sampled on a grid (stored in pixel form) at feature extraction step, and 2) their corresponding visual words assigned from the visual words codebook. As shown in Fig. 1, by look-up table, instead of extracting SIFT, query images are split into 16×16 pixel patches. Then, these patches are used to search the nearest patches in database so that its



(a) Average searching time for a picture under different size of database of table



(b) Concordance rate (top-1 and top-10) under different size of database of table, with FLANN & exhaustive search (exh)

Fig. 2: Experiment result on Caltech101/Caltech256

Table 1: Accuracy of proposed method compared to SIFT way. M, N is the number of pictures used to train database and form table, respectively.

SVM Classification	Dataset: Caltech101				
M/N	1530/510	1530/1530	3060/510	3060/1530	3060/3060
Accuracy (Conventional Method,%)	**43.98**		**53.66**		
Accuracy (Proposed Method,%)	**29.17**	**30.92**	**36.91**	**39.00**	**38.62**

corresponding visual words are obtained.

4. Experiment

In our experiment, we adopted FLANN [4] as our indexing and searching method of finding the nearest patch, and used dataset of Caltech101 and Caltech256. In addition, the images were all preprocessed into gray scale and normalized (mean is zero). We extended the look-up table from only one nearest visual word stored, into the top-10 nearest so that we can calculate concordance of them. Also, we implemented a conventional method of feature extraction and visual words assignment on the query pictures in order to calculate the concordance between the visual words found by look-up table and the visual words assigned by the codebook.

As shown in Fig. 2 (a), the proposed method is faster compared to the conventional method under various size of database for both dataset. Time of assigning visual word to pictures is reduced by up to 85% in condition of small database. Fig. 2 (b) showed that the top-1 concordance of patches is about 25-35% and the top-10 concordance is about 55-70%. To know if the lower concordance is caused by improper policy of finding the nearest neighbor, we also implemented the exhaustive search on Caltech101 dataset. However, the result is even worse than that with FLANN, which shows that we should not pay attention to finding the nearest neighbor but revising the proposed method.

To evaluate the performance of the proposed method, classification based on histogram vectors, which is summation of visual words formed by top-1 visual word, was implemented by LIBSVM [5] (with RBF kernel). As shown in Table 2, though the proposed method is faster, lower accuracy is observed compared to the conventional method.

5. Conclusion

This study presented a fast BoF method to deal with large-scale image classification. The proposed look-up table created by local patches and their corresponding visual words is used to find BoF-based indexes of query images. With the proposed method, time for assigning visual word can be reduced by up to 85%, while lower accuracy was observed compared to common method. The proposed method may need to be revised, for example, patches can be added by some processing in SIFT (e.g. orientation) to enhance the distinctness when finding the nearest neighbor.

Reference

[1] Wu, Y., Lu, S., Mei, T., Zhang, J., and Li, S. : "Local visual words coding for low bit rate mobile visual search. " ACMMM, pp. 989-992 (2012)

[2] K. I., H. Y. and K. A. : "Fast Bag-of-Feature Generation Featuring Approximate Nearest Image-Patch Search, " MVE, pp. 125-126 (2014)

[3] Yang, J., et al. : "Linear spatial pyramid matching using sparse coding for image classification," CVPR, pp. 1794–1801 (2009)

[4] M. Muja and D. G. Lowe: "Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration," VISAPP, pp. 331-340 (2009)

[5] C.-C. C. and C.-J. L. : "libsvm: a library for support vector machines".ACM TIST, Vol. 2, pp. 1-27 (2011)

† 東京大学　工学部電子情報工学科

〒113-8656 東京都文京区本郷 7-3-1

‡ Department of Electrical and Computer Engineering, National Chiao Tung University. 1001 University Road, Hsinchu 30010, Taiwan.