Research Article
Research on Vocabulary Sizes and Codebook Universality

Wei-Xue Liu,[1] Jian Hou,[1] and Hamid Reza Karimi[2]

[1] *School of Information Science and Technology, Bohai Univiersity, Jinzhou 121013, China*
[2] *Department of Engineering, Faculty of Engineering and Science, University of Agder, 4898 Grimstad, Norway*

Correspondence should be addressed to Wei-Xue Liu; bhulwx@gmail.com

Codebook is an effective image representation method. By clustering in local image descriptors, a codebook is shown to be a distinctive image feature and widely applied in object classification. In almost all existing works on codebooks, the building of the visual vocabulary follows a basic routine, that is, extracting local image descriptors and clustering with a user-designated number of clusters. The problem with this routine lies in that building a codebook for each single dataset is not efficient. In order to deal with this problem, we investigate the influence of vocabulary sizes on classification performance and vocabulary universality with the kNN classifier. Experimental results indicate that, under the condition that the vocabulary size is large enough, the vocabularies built from different datasets are exchangeable and universal.

1. Introduction

Codebook is a feature representation which is originally used in text processing. This representation extracts the keywords from the text and then uses the frequency of these keywords to analyze the meaning of text. Therefore, we can get a compact and effective description of the text, which is easy for content-based text retrieval. In recent years, this approach is successfully applied to many image processing applications, including image retrieval, scene recognition, and classification. Codebook has become a very popular and effective method to represent the image characteristics.

In the codebook representation, one feature detector is used to extract the keypoints from given images in the first step. Generally, these keypoints are the most distinctive and stable areas in images, which are robust to the variation of light and perspective and could be detected reliably. Then, one descriptor is utilized to represent these keypoints. In general, the gray scale and color information in the neighbor of the key points will be expressed as a vector. The descriptor of a keypoint contains the most important information in the neighborhood of the keypoint and abandons the useless information. Compared with original gray scale or color values, the keypoint descriptor is more appropriate for feature matching and image representation. In the next step, the

descriptors are clustered and the centers of clusters are used to represent the descriptors in the clusters. This step helps to reduce the size of codebook and save memory space. It is also beneficial to increase the computation efficiency and improve the robustness to outliers and noise. The collection of cluster centers is the so-called codebook. The cluster centers could be the mean of all descriptors in the clusters, or the descriptor most close to the mean one. One image can then be expressed as a histogram of the cluster centers in the codebook [1, 2], that is, the frequency of each cluster center occurring in the image. In one codebook, each code corresponds to a descriptor and one type of image pattern. That is the reason why the codebook is also called visual word in image processing.

2. Related Works

The main idea of codebook is to calculate the distribution of feature detection operator vectors in the whole image. The advantage of this method is that it is invariant to image rotation and scale. Since its introduction, codebook has become a very popular feature descriptor in image classification [3–7]. However, the classical codebook methods fail to consider the relative position information of these feature points. In some images, the feature points are similar

but the relative positions of these feature points are different; the corresponding images are totally different. Thus, the relative positions play an important role in the scene image classification [8, 9]. In order to describe the relative spatial information and apply it in the codebook, the literature [10] proposes a pyramid based codebook method. This method divides the whole image into several grids. The pixels in each grid are counted as a codebook histogram. Then all the histograms are merged as a vector to represent the whole image. Therefore, the pyramid based codebook method can describe the distribution of feature detection operator vectors in order. If there is no rotation in the image matching, the recognition precision will be improved. In the real application, this strategy has achieved good performance. It deserves mentioning that the literatures [11, 12] develop another codebook method to make full use of relative spatial information, which also achieve good performance. Given an image database, codes in a codebook play different roles. Some of them are very important in identifying images while others may have negative effect in image classification and retrieval. Thus, the literatures [2, 13, 14] proposed a weighted codebook algorithm. Different codes are assigned different weight. Thus, the weighted codebook can have a better performance in scenery classification and retrieval. Another strategy is that some useless code is removed from codebook and the size of codebook is shrinking [15, 16]. The advantage of this strategy is that it can reduce the storage space and improve the computational efficiency when the performance is affected a little. Considering the bombing development of the digital images on the internet, the real tasks of scenery classification and retrieval rely on super image database, which would contain millions or billions images. However, the traditional codebook algorithms are only desired for small image database, which only consists of hundreds or thousands of images. When facing super databases, the traditional codebook algorithms would have computational efficiency problems. To solve this problem, the literature [14] introduces vocabulary tree to improve the efficiency of codebook training. In this method, a hierarchical clustering strategy is applied to speed up the process of codebook vectors constructing and matching calculating. Thus, this method would have a good performance when applied to super databases. At the same time, researchers are exploring more applications for codebook, such as new image feature extractions [17] and video processing [18]. Since codebook is widely used in classification, the idea can be potentially used in some other domains, for example, fault diagnosis and others [19–22].

Clustering is an important step in codebook construction. In the real applications, some images are selected as training samples. Then, the feature points are extracted and described for each image. These obtained feature description operator vectors are used for clustering. The most representable descriptor in each cluster is built as a code. In codebook algorithm, the size of the codebook or the number of clusters will have some influence on the codebook performance. If the size is very small, numerous descriptors are represented by a little code. Some dissimilar images will be recognized as

the same class because they represent the same code. Thus, the performance of scenery classification and retrieval will be very bad. With the increasing of codebook size, the code can distinguish the difference between different images more accurately. Thus, the identifying ability will be strengthened. However, literatures [9, 23] point out that the large size codebook will also reduce the performance. An optimized size for codebook is very important.

Although the influence of codebook size on performance is obvious, the selection of codebook size is based on experience. And the existing codebook size ranged from hundreds to ten thousands [1, 10, 24, 25]. On one hand, the performance requires a large codebook size. On the other hand, considering the computational efficiency and storage space, the size of codebook should not be too large. The literatures [9, 23] illustrate that there should exist an optimized codebook size for a given image database. However, they do not give any theoretic explanation for codebook size section.

Another important problem is the codebook building method. The codebook comes from the text processing areas. In text processing, there exists a vocabulary with limited size. This vocabulary can be used for any text retrieval and classifications. However, in image process, most researchers do not think there is a universal codebook for all image databases. They build a codebook for a given database, and the codebook is not used in other image databases. The literatures [26–29] have explored the possibility of building the universal codebook, which can be used in various image databases.

3. Vocabulary Sizes and Universality

In order to solve the above problem, in this paper we investigate the relationships among vocabulary sizes, classification precisions, and universality. In [28], the work has been done with SVM classifier. However, it is not clear if the conclusions from [28] are also applicable to kNN classifier.

In this paper, experiments are conducted on three image databases, that is, Event-8, Scene-15, and Caltech-101. These three databases contain various scenery images, which can ensure the reliability of experimental results. In the training process, we use SIFT to extract the feature detection operator firstly. Then the k-means algorithm is used for clustering. And the describing vector is obtained in the clustering process. The experiments are conducted in two cases; that is, the value of k in kNN classifiers is optimized via cross validation and fixed as 3.

The first set of experiments are to test the influence of codebook size to the classification performance. The size of codebook is selected as 100, 500, 1000, 5000, 10000, and 50000. The classification results with k optimized and $k = 3$ on the mentioned three databases are shown in Figure 1.

In [9] the authors use SVM classifier to evaluate the discriminative power of codebooks of different sizes, and they conclude that with the increase of sizes, the discriminative power firstly rises, then peaks, and finally drops. However, from Figure 1, we see that with kNN classifier, the discriminative power firstly rises, then peaks, then drops, and finally
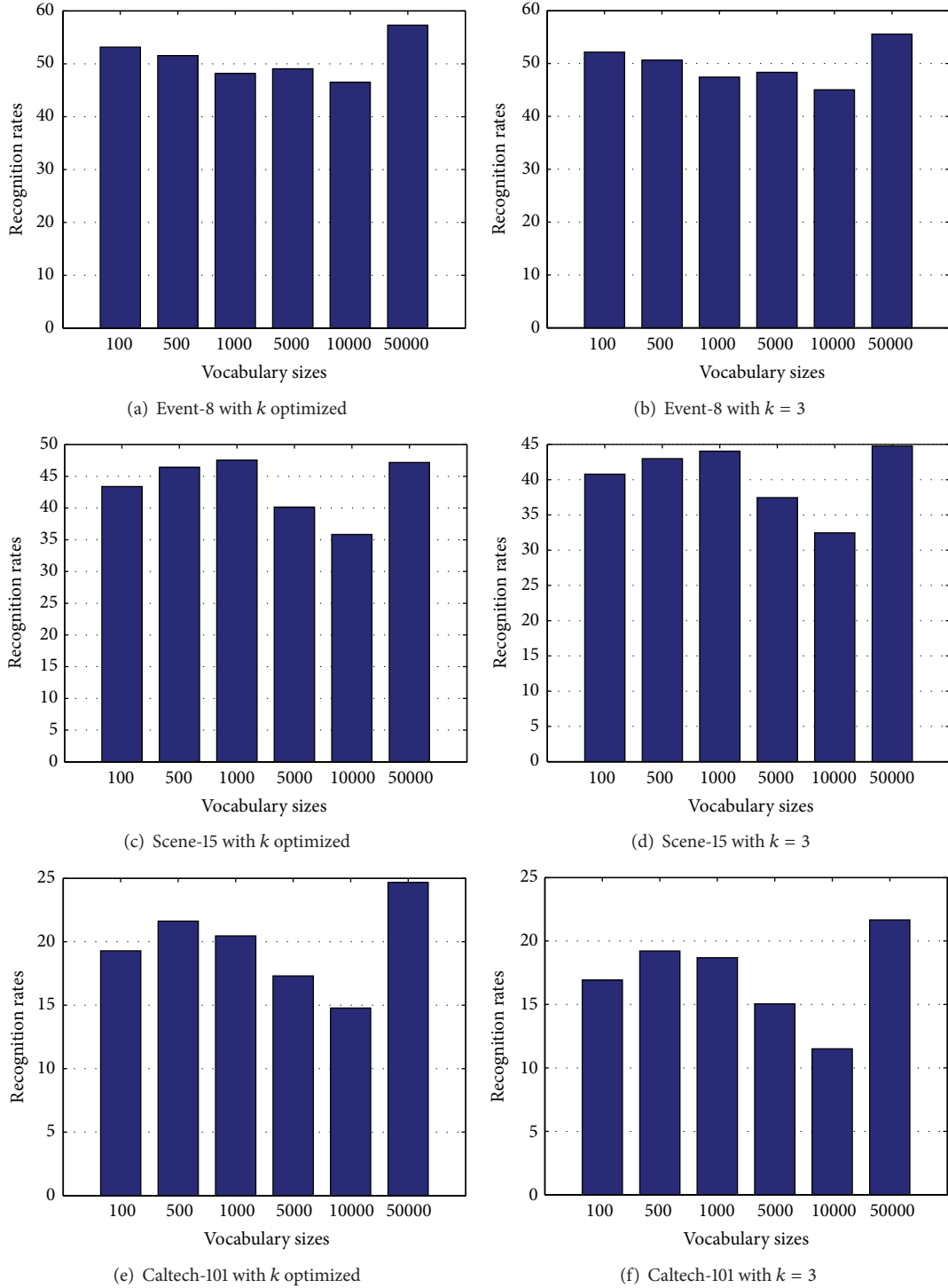
(a) Event-8 with k optimized

(b) Event-8 with $k = 3$

(c) Scene-15 with k optimized

(d) Scene-15 with $k = 3$

(e) Caltech-101 with k optimized

(f) Caltech-101 with $k = 3$

FIGURE 1: The relationship between classification rates and vocabulary sizes, with k optimized and $k = 3$.

rises gain, with the increase of sizes. This trend exists in both the cases of k optimized and of $k = 3$. As a result, we cannot regard this as an accident. This evident difference with the trend in SVM classifier shows that the behavior of codebook is more complicated and requires further study. As a result, it is not appropriate to apply the conclusions from SVM classifiers to kNN classifiers.

Another set of experiments are conducted to investigate the relationship between vocabulary sizes and universality of the algorithm. In detail, we build three codebooks in three databases, named as voc-event-8, voc-scene-15 and voc-caltech-101. And the obtained codebooks are applied on the image classification in image databases Event-8, Scene-15, and Caltech-101, respectively. And the image classification
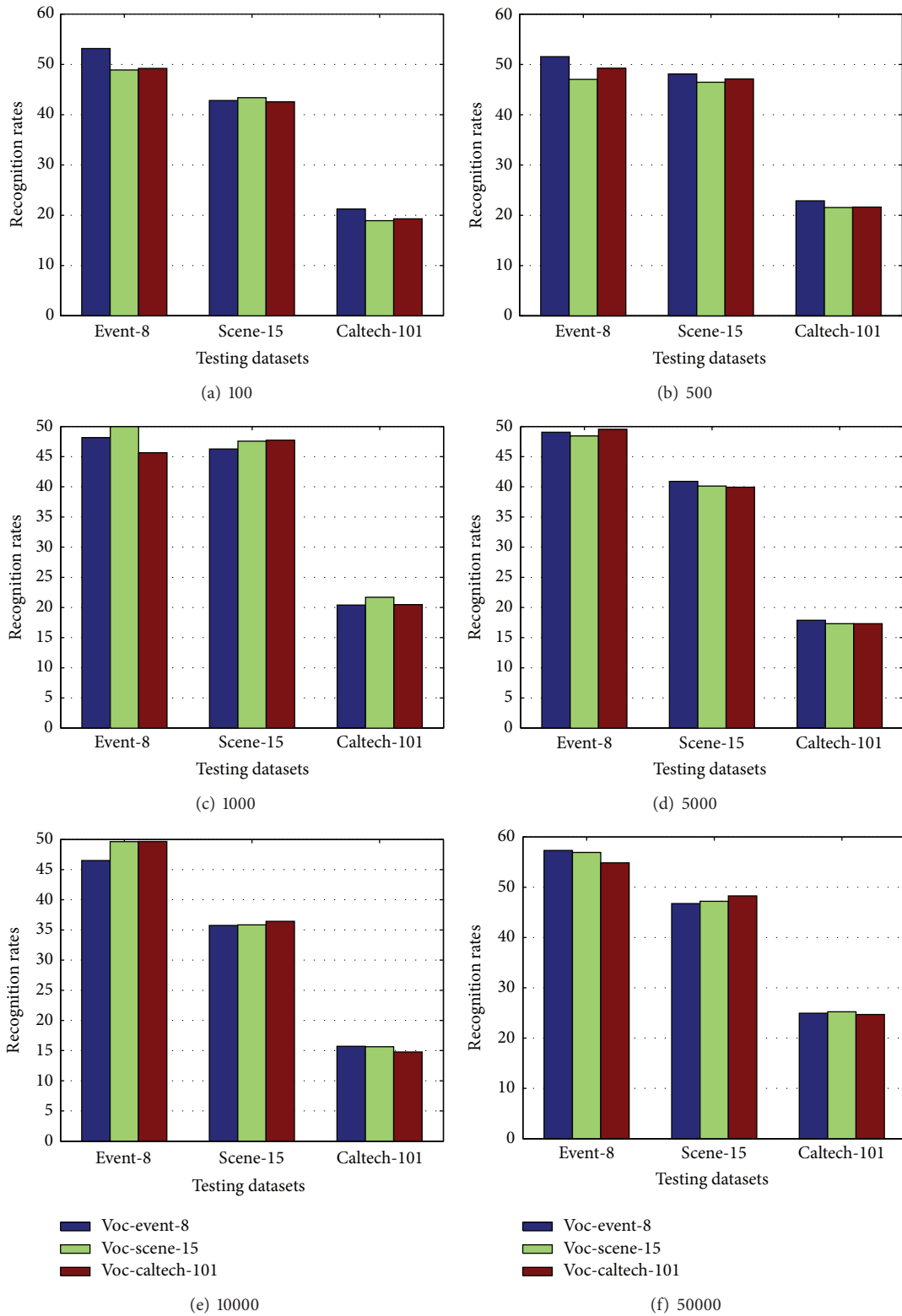
(a) 100

(b) 500

(c) 1000

(d) 5000

(e) 10000

(f) 50000

FIGURE 2: The relationship between universality of the algorithm and vocabulary sizes, with k optimized.

(a) 100

(b) 500

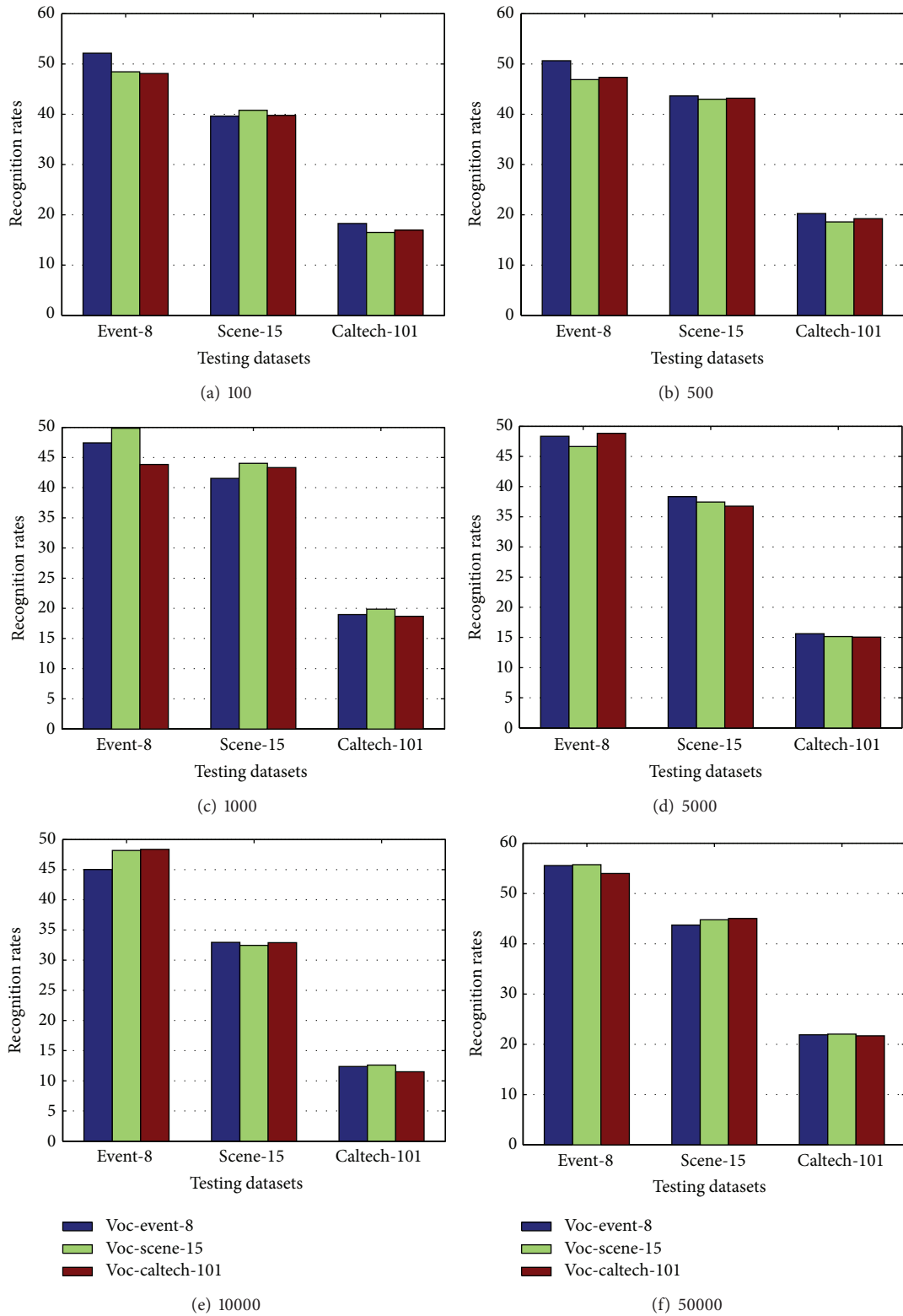(c) 1000

(d) 5000

(e) 10000

(f) 50000

FIGURE 3: The relationship between universality of the algorithm and vocabulary sizes, with $k = 3$.

rates with k optimized and $k = 3$ are compared in Figures 2 and 3, respectively.

From Figures 2 and 3, we can see that the codebook generated from different image database has almost the same recognition rates with the different vocabulary sizes. However, the difference among recognition rates becomes less if the vocabulary size is larger. This observation indicates that codebooks built from different image database can be used to obtain similar recognition rates and thus have a good universality. This further implies that it is unnecessary to build codebook for every image database. It is enough to construct a codebook in one database and apply it to image classification with other datasets.

4. Conclusion

This paper explores the relationships among the vocabulary sizes, classification performance, and universality of codebooks. Specifically, we conduct kNN classification experiments on three image databases with 6 representative vocabulary sizes. The experimental results indicate that if the vocabulary size is large enough, the codebooks built from different image datasets have basically the same discriminative power and can be used as universal codebooks. As a result, it is unnecessary to build codebook for every image database. Another important conclusion is that the relationship between vocabulary sizes and discriminative powers using kNN classifier is different from that of using SVM classifier. This indicates that the behavior of codebook is more complicated than expected, and further works are required in this respect.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] J. Sivic and A. Zisserman, "Video google: a text retrieval approach to object matching in videos," in *Proceedings of the 9th IEEE International Conference on Computer Vision*, pp. 1470–1477, October 2003.

[2] K. Grauman and T. Darrell, "The pyramid match kernel: discriminative classification with sets of image features," in *Proceedings of the 10th IEEE International Conference on Computer Vision (ICCV '05)*, vol. 2, pp. 1458–1465, October 2005.

[3] X. Li, W. Yang, and J. Dezert, "An airplane image target's multifeature fusion recognition method," *Acta Automatic Sinica*, vol. 38, pp. 1298–1307, 2012.

[4] J. Hou and M. Pelillo, "A simple feature combination method based on dominant sets," *Pattern Recognition*, vol. 46, pp. 3129–3139, 2013.

[5] J. Hou, W. X. Liu, and H. R. Karimi, "Exploring the best classification from average feature combination," *Abstract and Applied Analysis*, vol. 2014, Article ID 602763, 7 pages, 2014.

[6] J. Hou, W. X. Liu, and H. R. Karimi, "Sampling based average classifier fusion," *Mathematical Problems in Engineering*, vol. 2014, Article ID 369613, 2014.

[7] J. Hou, B. P. Zhang, N. M. Qi, and Y. Yang, "Evaluating feature combination in object classification," in *Proceedings of the International Symposium on Visual Computing*, pp. 597–606, 2011.

[8] S. Lazebnik, C. Schmid, and J. Ponce, "A maximum entropy framework for part-based texture and object recognition," in *Proceedings of the 10th IEEE International Conference on Computer Vision (ICCV '05)*, pp. 832–838, October 2005.

[9] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo, "Evaluating bag-of-visual-words representations in scene classification," in *Proceedings of the international workshop on Workshop on Multimedia Information Retrieval*, pp. 197–206, September 2007.

[10] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: spatial pyramid matching for recognizing natural scene categories," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 2169–2178, June 2006.

[11] M. Marszałek and C. Schmid, "Spatial weighting for bag-of-features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 2118–2125, June 2006.

[12] V. Viitaniemi and J. Laaksonen, "Spatial extensions to bag of visual words," in *Proceedings of the ACM International Conference on Image and Video Retrieval (CIVR '09)*, pp. 280–287, July 2009.

[13] H. Cai, F. Yan, and K. Mikolajczyk, "Learning weights for codebook in image classification and retrieval," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 2320–2327, June 2010.

[14] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, pp. 2161–2168, June 2006.

[15] T. Li, T. Mei, and S. K. In, "Learning optimal compact codebook for efficient object categorization," in *Proceedings of the IEEE Workshop on Applications of Computer Vision (WACV '08)*, pp. 1–6, January 2008.

[16] P. K. Mallapragada, R. Jin, and A. K. Jain, "Online visual vocabulary pruning using pairwise constraints," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, vol. 2, pp. 3073–3080, June 2010.

[17] S. Alvarez and M. Vanrell, "Texton theory revisited: a bag-of-words approach tocombine textons," *Pattern Recognition*, vol. 45, pp. 4315–4325, 2012.

[18] H. S. Min, S. M. Kim, W. D. Neve, and Y. M. Ro, "Video copy detection using inclinedvideo tomography and bag-of-visual words," in *Proceedings of the International Conference on Multimedia and Expo*, pp. 562–567, 2012.

[19] S. Yin, S. Ding, A. Haghani, and H. Hao, "Data-driven monitoring for stochasticsystems and its application on batch process," *International Journal of Systems Science*, vol. 44, pp. 1366–1376, 2013.

[20] S. Yin, S. Ding, A. Haghani, H. Hao, and P. Zhang, "A comparison study of basic datadriven fault diagnosis and process monitoring methods on the benchmark tennessee eastman process," *Journal of Process Control*, vol. 22, pp. 1567–1581, 2012.

[21] S. Yin, H. Luo, and S. Ding, "Real-time implementation of fault-tolerant control systemswith performance optimization," *IEEE Transactions on Industrial Electronics*, vol. 61, pp. 2402–2411, 2014.

[22] S. Yin, X. Yang, and H. R. Karimi, "Data-driven adaptive observer for fault diagnosis," *Mathematical Problems in Engineering*, vol. 2012, Article ID 832836, 21 pages, 2012.

[23] T. Deselaers, L. Pimenidis, and H. Ney, "Bag-of-visual-words models for adult image classification and filtering," in *Proceedings of the 19th International Conference on Pattern Recognition (ICPR '08)*, pp. 1–6, December 2008.

[24] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: an in-depth study," Tech. Rep., INRIA, 2003.

[25] W. Zhao, Y. Jiang, and C. Ngo, "Keyframe retrieval by keypoints: can point-to-pointmatching help?" in *Proceedings of the ACM International Conference on Image and Video Retrieval*, pp. 72–81, 2006.

[26] C. X. Ries, S. Romberg, and R. Lienhart, "Towards universal visual vocabularies," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '10)*, pp. 1067–1072, July 2010.

[27] J. Hou, Z. S. Feng, Y. Yang, and N. M. Qi, "Towards a universal and limited visual vocabulary," in *Proceedings of the International Symposium on Visual Computing*, pp. 414–424, 2011.

[28] J. Hou, W. X. Liu, W. X. E, W. X. X, Q. Xia, and N. M. Qi, "An experimental study on theuniversality of visual vocabularies," *Journal of Visual Communication and Image Representation*, vol. 24, pp. 1204–1211, 2013.

[29] J. Hou, W. X. Liu, W. X. E, W. X. X, and H. R. Karimi, "On building a universal and compactvisual vocabulary," *Mathematical Problems in Engineering*, vol. 2013, Article ID 163976, 8 pages, 2013.