Encoding Optimization Using Nearest Neighbor Descriptor

Muhammad Rauf, Yongzhen Huang and Liang Wang

National Lab of Pattern Recognition,
Institute of Automation, Chinese Academy of Science, Beijing 100190, China

Abstract. The Bag-of-words framework is probably one of the best models used in image classification. In this model, coding plays a very important role in the classification process. There are many coding methods that have been proposed to encode images in different ways. The relationship between different codewords is studied, but the relationship among descriptors is not fully discovered. In this work, we aim to draw a relationship between descriptors, and propose a new method that can be used with other coding methods to improve the performance. The basic idea behind this is encoding the descriptor not only with its nearest codewords but also with the codewords of its nearest neighboring descriptors. Experiments on several benchmark datasets show that even using this simple relationship between the descriptors helps to improve coding methods.

Keywords: Nearest neighbor descriptor, Group saliency coding, Soft coding , Local constraint linear coding.

1 Introduction

One of the most important research areas in computer vision is image classification. There are different kinds of techniques used to serve this purpose. All these techniques have their benefits and drawbacks. Some work well on one kind of dataset and others can perform better on other kind of datasets. In all these techniques, the most commonly used framework is the Bag-of-words framework (BoW)[1][2]. This model consists of several steps, which starts from feature extraction and ends with classification. The hierarchy of these steps is, after feature extraction a codebook is generated and followed by feature coding, and before the classification feature pooling is performed.

All these steps have their own importance in the whole process of image classification using BoW. In recent years, encoding attracts lots of attention. There are different kinds of encoding methods that have been introduced to get better performance. Recent work[4][5] show that different coding methods perform different, even under the same framework. Soft voting outperforms hard voting[1] and the fisher kernel[6] has better performance than soft voting [3] with the same number of code words. These three are voting based methods and if we compare these voting based methods with reconstruction based

coding[4], like local constraint linear coding (LLC)[7], we find that LLC has better results than the voting based coding methods. On the other hand the saliency[8] and group saliency coding[9] methods have implementation advantages over the reconstruction based coding, and perform faster than LLC. There are other coding methods introduce to improve the performance e.g., Laplacian sparse coding[10], multi-layer group sparse coding[11], improved Fisher kernel coding[12], Local tangent-based coding methods[13] and many more.

One thing that is common in all these methods is to encode one descriptor with codewords. In this process, we exactly do not know the relationship between a descriptor and its adjacent descriptors. If the descriptor extraction is not very dense then what are the influence of one descriptor to its neighboring descriptors and their codewords, i.e., the codewords used to encode descriptors. The main focus of our work is, to encode the descriptor by using the nearest neighbor descriptor's (NND) codewords and observe the change in performance. We explore a relationship between descriptors and by using this relationship, we update the codewords of descriptors. Our proposed technique is very simple and easy to implement.

The rest of the paper is arranged as follows. In Section 2 we introduce our proposed method in detail. In Section 3 first we discuss the datasets and the coding methods, and afterwards we evaluate our proposed technique. At the end in Section 4 we present the conclusion and our future work.

2 Nearest Neighbor Descriptor

The proposed method not only considers the structure of K-nearest codewords to a descriptor, but also takes account of the structure of neighboring descriptor codewords. We present a new technique that uses the descriptor-to-descriptor relationship during the encoding process. Results show that the locality of the descriptors has a very important role in encoding.

Our implementation is done in two different phases. First, we find K-nearest codewords of a descriptor and finally we update each descriptor's codewords based on the NND codewords. Let $X = [x_1, x_2, ..., x_N] \in R^D \times N$ be N D-dimensional descriptor form an image, and $B = [b_1, b_2, ..., b_M] \in R^D \times M$ be a codebook with M codewords.

2.1 Local Code Assignment:

In this phase, we encode the descriptor with K codewords using the existing encoding methods. K is set to be a small number[20] and $[b_1, b_2, ..., b_K]$ is K closest codewords of x e.g., K=3 in Fig. 1(a). This is the local assignment of the nearest codewords to the descriptor. In the next phase, we generate new codewords that is based on the descriptor's and its neighboring descriptor's codewords.

2.2 Nearest Neighboring Descriptor

The position of the descriptor from its neighboring descriptor has an important factor during encoding. We update codewords for every descriptor by using its codewords and codewords of its NND. Suppose Y_i is the set of codewords of descriptor x_i and Z_i is the set of codewords of NND of x_i. We choose the codewords which are used to encode x_i by using Equation 1:

$$Y_i' = Y_i \cup (Z_i \setminus Y_i), \qquad (Z_i \setminus Y_i) = \{b \in Z_i | b \notin Y_i\} \tag{1}$$

where Y_i' is the updated set of codewords of the descriptor x_i and $'\setminus'$ stands for the relative complement function. $(Z_i \setminus Y_i)$ represents the relative complement of Y_i in Z_i, the set of codewords that are presented in Z_i but not in Y_i.

Our method is illustrated in Fig. 1. First, we find the nearest neighboring descriptor and then we assign the new codewords to the descriptor according to the nearest neighboring descriptor's codewords. Suppose we are going to encode x_2. The first step is to find the NND of x_2. Consider D_1 and D_2 are two distances between x_2 to x_1 and x_2 to x_3 respectively. Suppose D_1 is less than D_2, so x_1 is the nearest neighboring descriptor of x_2. By using the equation 1, we assign new codewords to x_2.

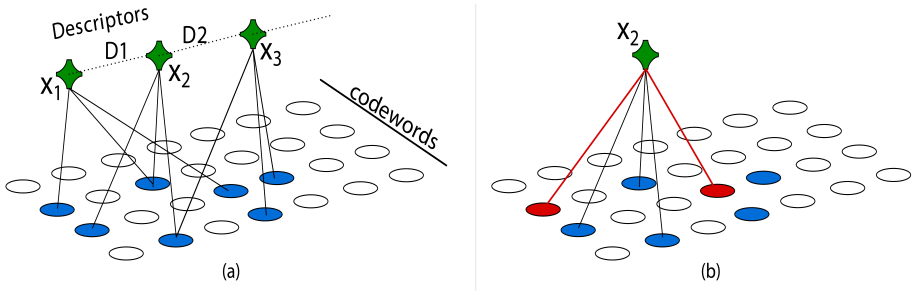

Fig. 1. Code selection on the base of the nearest neighboring descriptor

The distance between codewords and descriptors plays a very important role in the encoding process. The next step is to find the distance of codewords to it's new descriptor. After assigning new distance, we select K nearest codewords.

Suppose b_1 and b_2 are the codewords of the descriptor x_2 as shown in Fig. 2, where b_2 is new codeword of x_2 from it's NND. Suppose d_1 is the descriptor to descriptor distance and d_2 is the distance of codeword to it's original descriptor. We need to calculate d_3, the distance of the codeword to its new descriptor.

We use a simple technique to estimate the new distance of codeword to it's new descriptor. We use Pythagorean theorem[14] to calculate the distance. This will not get the exact distance but it will proximate the distance and improve the speed. According to our observation this estimation error is negligible with

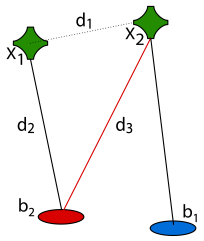

Fig. 2. Distance measurement of codewords to descriptors

respect to the fast performance of our technique. By using equation 2 we calculate d_3.

$$d_3 \approx \sqrt{d_1^2 + d_2^2} \tag{2}$$

In the final stage we supply these codewords to the selected encoding method to finalize the encoding process.

3 Experiments and Discussion

3.1 Datasets

The following four datasets are used for experimental study.

Scene 15[15] There are 4,485 images in the scene 15 dataset and these images belong to 15 different categories, each of which contains 200 to 400 images. We randomly select 100 images for training and the remaining for testing.

Caltech 101[16] This dataset contains 9,145 images from 101 different categories. These categories contain from 31 to 800 different numbers of images. We use the standard setting for this dataset.

VOC2007[17] There are 9,963 images in this dataset distributed into 20 classes. These images vary in their size, scales, viewpoint and other image properties. These images are divided into training and testing sets. VOC2007 is one of the major datasets used in image classification.

UIUC Sports[18] The UIUC Sport dataset consists of 1,574 sports images belonging to 8 different categories. We use this dataset for extensive study of our proposed technique.

3.2 Experimental Setting

We use three different encoding methods from three different encoding classes to observe the performance of our technique, i.e., kernel codebook encoding (KCB)[19], locality constrained linear coding (LLC) and group saliency coding (GSC). For all these methods the codeword size K is 5 and the codebook sizes are set to 512, 1024 and 2048.We use SIFT descriptor[21] for all these experiments.

The evaluation is performed with two different experimental settings. First, we evaluate performance with three different datasets and feature extraction of image is with 10 step size (i.e., extracting a descriptor over every 10 pixels). We use Scene 15, Calthech 101 and VOC2007 datasets for these experiments.In second group of experiments we use the UIUC-Sports dataset to evaluate the performance with 8, 10, 15 and 20 step size of image feature extraction.

3.3 Basic Results

As mentioned above in these experiments, we use three different datasets with one feature extraction size. Results of Caltech 101, Scene 15 and VOC2007 are shown in Fig. 3, Fig. 4 and Fig. 5 respectively. Each figure contains the results by GSC, KCB and LLC. The results of our proposed technique and original methods are compared. These results suggest that the performance of the NND based method is increased.

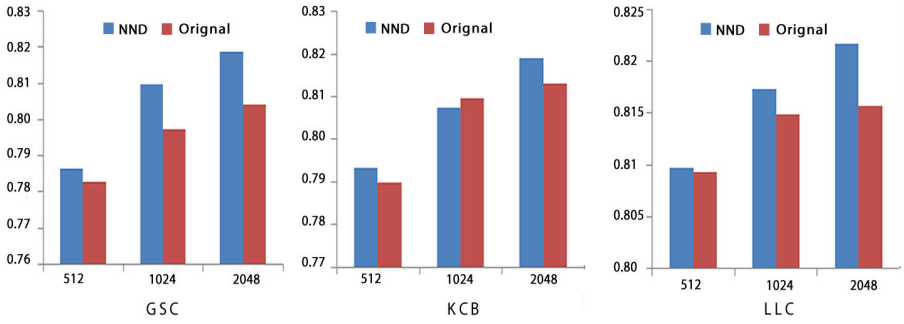

Fig. 3. Experimental results on the Caltech 101 Dataset.

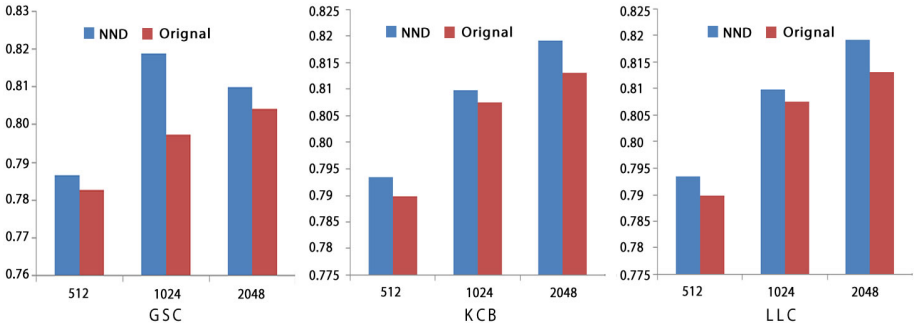

Fig. 4. Experimental results on the Scene 15 Dataset.

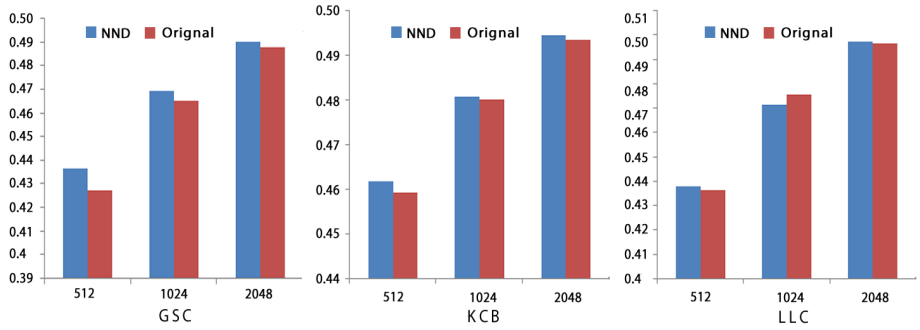

Fig. 5. Experimental results on the VOC2007 Dataset.

These different charts from each group involve three different codebook sizes. From these accuracy bars, it is clear that the accuracy is improved after using our technique. Although the improvement is not very large in some cases but still it has a small change in the performance.

We can observe that even the property of relationship is simple, it is still able to perform well. It should be noted that if we are able to explore a good relationship between the descriptors, we may obtain more improvement. These results show that NND performs with persistent enhancement on different datasets with different encoding methods.

3.4 Different Sampling Rate Evaluation

For further testing our proposed technique, we use UIUC-Sports dataset with different feature extraction sizes. In these experiments we use 20, 15, 10 and 8 step size of image feature extractions respectively.

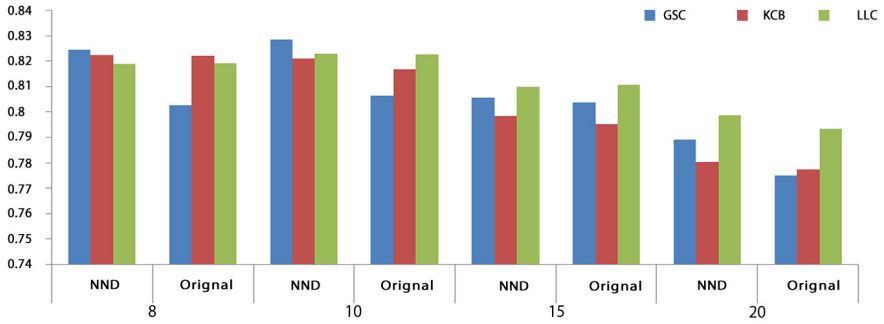

Fig. 6. Experimental results on the UIUC-Sport Dataset with a codebook size 512.

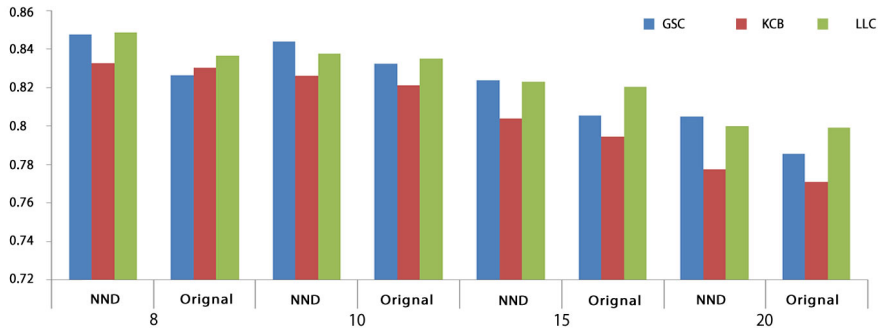

Fig. 7. Experimental results on the UIUC-Sport Dataset with a codebook size 1024.

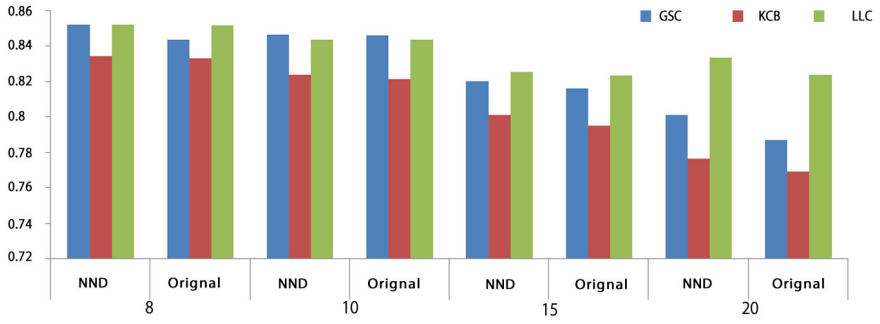

Fig. 8. Experimental results on the UIUC-Sport Dataset with a codebook size 2048.

We evaluate the performance of our proposed technique by comparing with original version of GSC, KCB and LLC. The results shown in Fig. 6, Fig. 7 and Fig. 8 give us clear observation on the performance improvement. Our technique again performs better in all these experimental settings. The performance difference of the NND and the original method is large when size of descriptors are not very dense. In low density of descriptor, the distance between the descriptor is large, so encoding with our proposed technique has more clear effects. This is probably because with the low descriptor density, descriptors are more scattered than with high descriptor density.

4 Conclusion and Future Work

In this paper, we have developed a new technique to improve the existing methods via exploring the relationship between descriptors. Our work has shown that if the relationship between the descriptors are developed in a meaningful way, it can help to get better results in terms of image classification. We have used this technique with GSC, KCB and LLC, and obtained improvement in all evaluation conditions.

Our future work is to extend this technique to video classification. It is believed that this method will generate better performance in video classification due to the finding that our proposed technique has better performance with low descriptor density, which is usually the case in video classification based on the bag-of-words framework.

References

1. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: ECCV, (2004)
2. van Gemert, J. C., Geusebroek, J. M., Veenman, C. J., Smeulders, A. W.: Kernel codebooks for scene categorization. In: ECCV, (2008)
3. van Gemert, J. C., Veenman, C. J., Smeulders, A. W., Geusebroek, J. M.: Visual word ambiguity. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 7, pp. 12711283, (2010)
4. Yu, K., Zhang, T., Gong, Y.: Nonlinear Learning using Local Coordinate Coding. In: NIPS, (2009)
5. Huang, Y., Wu, Z., Wang, L., Tan, T.: Feature coding in image classification: A comprehensive study. IEEE Trans. Pattern Anal. Mach. Intell., 36(3), pp: 493-506, (2014)
6. Perronnin, F., Dance, C.: Fisher kernels on visual vocabularies for image categorization. In: CVPR, (2007)
7. Huang, Y., Huang, K., Yu, Y., Tan, T.: Gong, Y.: Locality-constrained linear coding for image classification. In: CVPR, (2010)
8. Y. Huang, K. Huang, Y. Yu, and T. Tan, Salient coding for image classification. In: CVPR, (2011)
9. Wu, Z., Huang, Y., Wang, L., Tan, T.: Group encoding of local features in image classification. In: ICPR, (2012)
10. S. Gao, I. Tsang, L. Chia, and P. Zhao, Local features are not lonely - laplacian sparse coding for image classification. In: ECCV, (2010)
11. Gao, S., Chia, L. T.: Tsang, I. W.: Multi-layer group sparse coding for concurrent image classification and annotation. In: CVPR, (2011)
12. Perronnin, F., Snchez, J., Mensink, T.: Improving the Fisher kernel for large-scale image classification. In: ECCV, (2010)
13. Yu, K., Zhang, T.: Improved local coordinate coding using local tangents. In: ICML, (2010)
14. `http://en.wikipedia.org/wiki/Pythagorean_theorem`
15. `http://www-cvr.ai.uiuc.edu/ponce_grp/data/scene_categories/scene_categories.zip` (2006)
16. `http://www.vision.caltech.edu/Image_Datasets/Caltech101/101_ObjectCategories.tar.gz`
17. `http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2007/index.html`
18. `http://vision.stanford.edu/lijiali/event_dataset/event_dataset.rar`
19. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Lost in quantization: Improving particular object retrieval in large scale image databases. In: CVPR, (2008)
20. Liu, L., Wang, L., Liu, X.: In defense of softassignment coding. In: ICCV, (2011)
21. David G. L.: Distinctive image features from dcaleinvariant key-points. International Journal of Computer Vision, vol. 2, no. 60, pp. 91110, (2004)