A Robust and Discriminative Feature Representation based on Compact Coding

Jinpeng Yue[1] , Guang Jiang, Hong Hu, and Zhongzhi Shi

Abstract. As human vision system works, the visual features extracted from colours, textures, shapes, etc. are complement and should be synthesized to make a decision. Existing applications mainly base on one kind of features such as the wildly used texture feature. We put forward that complement features should be combined together to improve feature discriminabilty. And we propose a feature combining framework, which benefits from the recent fruitful research on compact coding of visual features. The compact codes are learned to be robust to complex image content variations. We combine colour histogram and texture under the framework, and experiments show that our method provides an effective way to the representation of features and has a wide application.

Keywords: Visual feature · Feature representation · Compact coding

1 Introduction

In computer vision, visual feature extraction is vital for object detection, image retrieval, scene classification, etc. As human vision system makes a decision based on many kinds of information, such as color, textures, shapes, etc, there are many kinds of visual features, such as color histogram, texture, and shape descriptor correspondingly. These features are complement and should be synthesized together in computer vision applications. However, existing applications mainly focus on one kind of features such as the wildly used SIFT [1] texture feature. To improve the feature discriminabilty, we put forward that complement features corresponding to different reacting regions of brain should be binding together to simulate the brain.

Wu [2] proposed a method to bundle two complement feature detecting methods together, i.e. SIFT [1] and MSER [3]. Xie [4] also bundles two detectors to-

[1] Jinpeng Yue (✉) , Guang Jiang, Hong Hu, Zhongzhi Shi
Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences. 100190 Beijing, China;

Jinpeng Yue , Guang Jiang
University of Chinese Academy of Sciences,100049 Beijing, China

e-mail: yuejp@ics.ict.ac.cn

gether. However, these two methods are detector binding methods, not considering the description aspects. The description of visual features is vital for the analysis of visual contents.

Furthermore, recent research has focused on the trade-off between the discriminabilty and compactness of visual feature descriptors. In large scale applications, the spatial consumption of visual features is critical and thus compact coding methods are popular recently. Many of such methods have been proposed to code features into more compact representations, such as binary codes.

We propose a framework named CCF (Coding and Combining Features) to combine feature descriptors together based on the compact coding. Specifically, our contributions are as follows:

- We propose a visual feature combining framework which take advantage of different feature extracting methods.
- In the framework, we encode features into compact binary codes by an effective coding method and combine the codes together to generate a more discriminative feature. The corresponding similarity measurement in the coding space is also given.

The combined feature achieves a good trade-off between discriminabilty and robust, and thus our method achieves a better performance in the experiments.

2 Related Works

According to the famous Marr's visual computing theory [5], Marr formulates an overall framework for the process of vision. The framework is based on three main representations of the image: (1) The primal sketch, which is mainly concerned with the description of the intensity changes in the image and their local geometry (2) The 2.5-D sketch, which is a viewer centered description of orientation, contour, motion and other properties of visible surfaces (3) The 3-D model, which is an object centered representation of 3-D objects. And our work correlates with the 2.5-D sketch.

In Marr's view, the vision process is a set of relatively independent modules. And then different types of information which are encoded in the image can be decoded by independent processes [6]. Meanwhile, the binding problem deals with the question of how features that are processed in parallel are bound to the one unique percept, for different features of an object are processed in different parts of the brain.

From this point of view, visual features extracted by the color, texture, shape, etc. information encoded in the image are processed as independent components in our model.

There are researches that have proposed feature bundling methods. Wu [2] bundles SIFT and MSER feature detecting methods together. SIFT detects corner points and MSER detects interesting regions. Thus bundling these two methods together can improve the robustness of detectors. Xie [4] bundles Harris-Laplace and SURF interest point detector together. Both of the above methods focus on the feature detecting process. Differently, we focus on the description and representation of features.

In computer vision field, many recent research focus on the compact representation of visual feature descriptors, which codes descriptors with more information and less space consumption. As early as 1961, Barlow proposed effect coding theory. Most of existing methods code descriptors in vector space without considering the supervised information. For example, spectral hashing [7] compact descriptors into binary codes by preserving the Euclidean distance in Hamming space. The supervised LDA Hash [8] utilizes the visual relationships between descriptors, but preserves a global structure. We propose a novel compact coding method, which can probe the manifold structure of data. And the information implicit in visual features is utilized in the coding process to get a more robust feature representation.

3 Coding and Combining Features Framework

3.1 Introduction of the Framework

The CCF Framework process an image into a set of codes, and then bundles the codes together to accomplish recognition or detection tasks.

The model is composed of three steps:

- *Visual feature extraction.* Visual features such as color histogram, Gabor filters, intensity histogram, etc. are widely used in content based image analysis. In figure 1, there are only three different features referred as samples. Actually, most of existing feature descriptors can be adopted in the model to simulate the visual processing of brain.
- *Compact coding.* After various features are extracted, we compact them into efficient and effective codes. To preserve as much information as possible during the coding process, we propose a novel coding method which will be introduced in detail in the next section.
- *Learning Combined Features*. The codes generated in the compact coding step are then combined together to make a more discriminative feature representation. Since different features such as color, texture impose different information, they are complement and effective in different situations. Unnumbered lists should use the “Bullet Item” style.

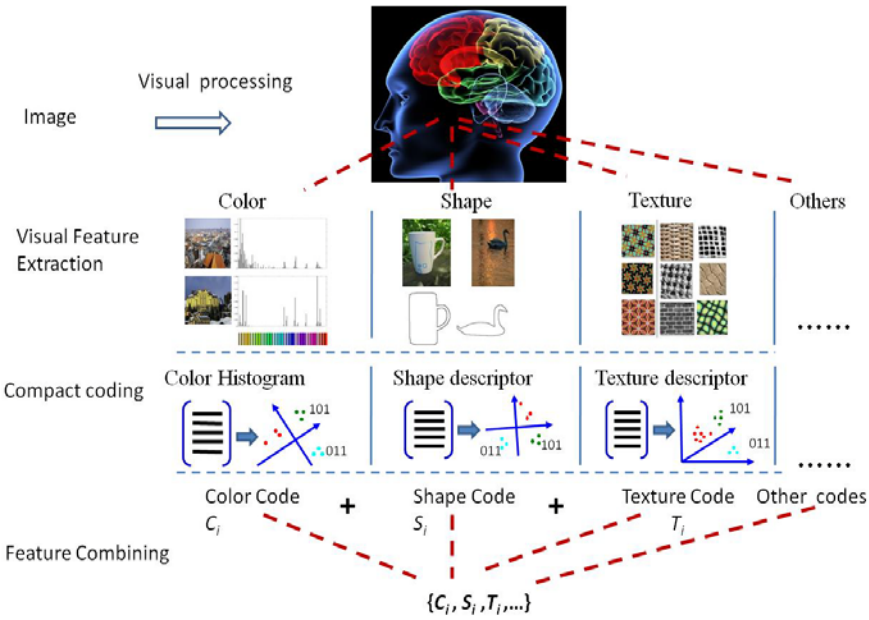

Fig. 1 Coding and Combining Features Framework

3.2 Compact Coding

In this step, our object is not only to encode the features to a more robust and compact codes, but also to bind the features that are visually similar but change slightly in visual contents. We seek a binary coding approach that preserves the visual similarity relationships under the encoding Hamming space. Specifically, we propose a Locality Preserving Coding (LPC) method to learn robust binary codes by exploring the underlying manifold structure of training samples, with effective and efficient optimization. We generate visually similar images by transforming images as is shown in Fig.2. We transform images into scale space by convoluting Gaussian function to simulate blur.

$$G(x,y,\sigma) \quad \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \tag{1}$$

Then binary codes are learned on the images by exploring the manifold structure of descriptors. The objective is to code the visually similar images into similar binary codes. The other transformations such as lighting, Gaussian noise, scaling, etc can also be used to simulate visually similar images. In this section, we take

the blur transformation as example. And in the experimental section, we use affine transformation.

Fig. 2 Samples of visually similar images

We aim to preserve the manifold structure of visual similar features when compacting descriptors to binary codes. Let $X=\{x_1,...,x_n\}$ be a set of d-dimensional visual descriptors. Suppose we want to get a k-bit code y_i of x_i, then k hash functions leading to k Hamming embeddings are needed. We use linear projection and threshold to encode X, and then binary code is computed using the following equation:

$$y_i = \text{sign}\,(Px_i^T + t) \tag{2}$$

where P is a $k*d$ matrix and t is a $k*1$ vector.

The projection matrix P is the key of embedding and threshold t is the key of binarization. To encode the visually similar descriptors into the same binary code, we compute P by solving an optimization problem and utilizing the visual relationships between descriptors.

Given a matrix W with weights characterizing the similarity of two image patches, we want to learn a P to preserve the local structure, which satisfies the following:

$$\text{minimize}: \sum_{ij} \left\| y'_i - y'_j \right\|^2 W_{ij} \tag{3}$$

where y' is the data projected by P, i.e. $y'=Px^T$ and w_{ij} is defined as follows:

$$w_{ij} = \begin{cases} 1, \text{If } x_i \text{ and } x_j \text{ are visully similar} \\ 0, \text{otherwise} \end{cases} \tag{4}$$

The similarity learning method is the spirit of LPP and P can be obtained by solving a generalized Eigen value problem as in [9].

Then we concatenate the generated codes to the final decision code. The similarity between two objects can be measured by the distance between their final decision codes. A smaller distance means the two objects are more similar.

3.3 Learning Combined-Feature

In this section, we take the color and texture features for example to introduce the combined process. Suppose the compact codes generated from color and texture features in the last section are denoted as C and T. Then the final combined code is defined as $\{C \cup T\}$.

The CCF framework can be applied to the object retrieval and recognition task. Take retrieval for example, the similarity between two objects can be measured by the distance of their final decision codes. Suppose the hamming distance between two codes is defined as DIS, then the distance function of two combined features $\{C_1 \cup T_1\}$ and $\{C_2 \cup T_2)$ is DIS(C_1-C_2)+DIS(T_1-T_2). A smaller distance means the two objects are more similar.

4 Experimental Results

In our CCF framework, compact coding and feature learning combined features are two key steps. Since which features should be extracted, encoded and bind needs more discussions, we only evaluate the independent compact coding step here.

Dataset. We simulate the affine transformations of images, utilize SIFT to detect features and generate visually similar image patches. There are totally 10k groups, some of which are illustrated in Figure 3. Each group of image patches is similar in vision but changes in computer representations. The objective of compact coding is to encode one group of patches into the same binary code. We select 5k groups randomly for training and the rest for testing.

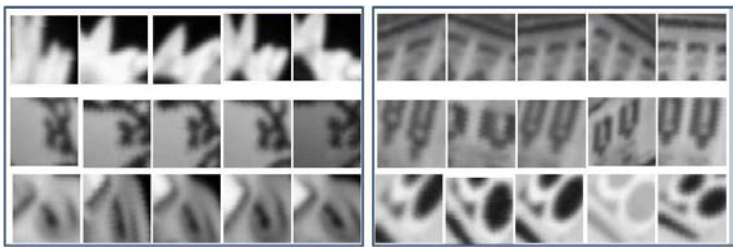

Fig. 3 Samples of evaluation dataset

We compare our compact coding method with LSH (Locality Sensitive Hashing) [10], which is a famous coding method in multimedia retrieval task. The experiment conducts image patch retrieval using our coding method and LSH, the results are shown in Figure 4.

Figure 4 uses recall and rate-of-codes-retrieved curve to evaluate the retrieval task. A higher recall means more correct similar patches are found, which indicates the image patch retrieval accuracy. The rate of codes retrieved indicates the

noises contained in the final results. And thus a lower rate means lower noises and the number of incorrect patches is smaller. We can conclude from the figure that our compact coding method achieves a better performance in this task. The reason is that our coding method not only utilizes the supervised visual information, but also considers the manifold structure of data.

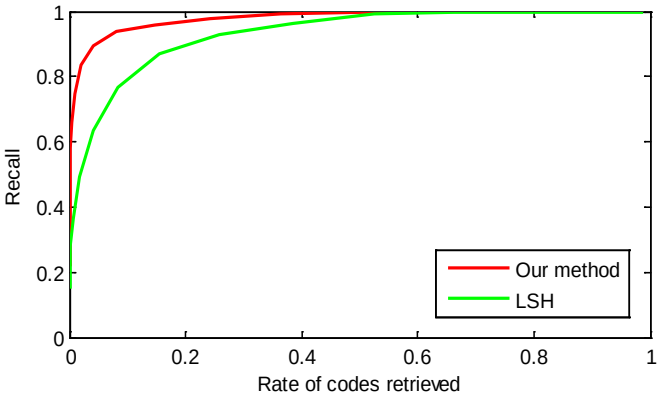

Fig. 4 Comparison of compact coding methods

Then we evaluate the combined feature on the Ukbench [11] image retrieval dataset, which contains 10,200 images including groups of 4 similar images. In this experiment, we combine texture histogram and color histogram together based on the compact coding. The texture feature is extracted by computing texture histogram from image patches, and the dimensionality is 80. The color feature is extracted by computing quantized color histogram and the dimensionality is 27. And the compact code is of 64-bit, which consumes much less memory.

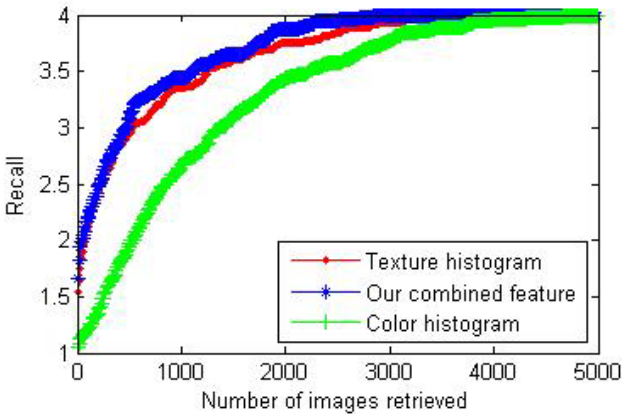

Fig. 5 Combined feature evaluation

From Figure 5, we can see that the combined feature achieves a higher recall than the color histogram and the texture feature. The combined feature utilizes two complementary features and benefit from compact coding, and thus features generated by our CCF framework is more discriminative and robust. Our method is

effective with much less memory consumption, which can be used in large-scale applications.

5 Conclusions

We propose a novel feature combining framework to generate a robust and discriminative feature representation with effective coding technique. In the framework, features are encoded into compact codes which represent features with much more information and less units. The framework can be applied to many other feature extraction methods. Actually, the weights of different features can be utilized and learned from training samples. The importance of different features needs a more deep study in the future.

Acknowledgment. This work was Supported by the National Program on Key Basic Research Project (973 Program) (No. 2013CB329502), National Natural Science Foundation of China (61035003).

References

1. D. Lowe, “Distinctive image features from scale invariant keypoints”. In IJCV 60(2):91-110, 2004.
2. Zhong Wu, Qifa Ke, Michael Isard, and Jian Sun , Bundling Features for Large Scale Partial-Duplicate Web Image Search. CVPR 2009.
3. J.Matas, O. Chum, M. Urban, and T. Pajdla, “Robust wide baseline stereo from maximally stable extremal regions”. In BMVC, p384-393, 2002
4. Hongtao Xie, Ke Gao, et al. Effective and Efficient Image Copy Detection Based on GPU. CVGPU Workshop of ECCV, 2010
5. David Marr. Vision: A Computational Investigation into the Human Representation and Processing of Visual Information. Henry Holt and Company, 1983
6. T.Poggio, Marr's computational approach to vision. Trends in NeuroSciences 1981,Vol. 4, No.10, pp 258-262
7. Weiss, Y., Torralba, A. and Fergus R. Spectral Hashing. Advances in Neural Information Processing Systems, 2008.
8. Strecha, C., Bronstein, A., Bronstein, M., Fua, P. LDAHash: Improved Matching with Smaller Descriptors. IEEE T. PAMI 34(1), 2012.
9. X. He, P. Niyogi. Locality Preserving Projections. In: Neural Information Processing Systems. MIT Press. 2003
10. Datar, M., Immorlica, N., Indyk, P.,et al.2004.Locality-sensitive hashing scheme based on p-stable distributions. SCG 2004, 253-262.
11. D.Nister, H.Stewenius,: Scalable Recognition with a Vocabulary Tree. Computer Vision and Pattern Recognition. 2006