

# Codebook Design of Keyblock Based Image Retrieval

Hui LIU and Cai-ming ZHANG

School of Computer Science & Technology, Shangdong Economic Univ.,  
Ji'nan, Shandong Province, 250014

**Abstract.** This paper presents an image retrieval method based on keyblocks combing with interest points, furthermore the generation of codebook is also utilized to enhance the retrieval performance, where the balance between the retrieved precision and time cost can decide the codebook size of this CBIR system. Finally, the proposed method is compared with the method that only depends on interest points or keyblocks.

**Keywords:** CBIR, keyblocks, codebook, retrieved precision

## 1 Introduction

Content-based image retrieval (CBIR) is currently a very important research area on multimedia applications. In order to overcome the deficiencies of global feature, a series of ROI based methods has been proposed<sup>[1]</sup>. The typical systems are Blobworld system<sup>[2]</sup> of University of California, Berkeley and SIMPLIcity system<sup>[3]</sup> of Pennsylvania State University, etc. But only the gray information is considered in these systems, which is not reasonable and has difference with human visual perception.

The target of detecting all interest points is to extract the most representative points in an image. Moravec published a paper in 1977 which mentioned corners detection<sup>[4]</sup>. Now we can receive above interest points detection result by use of Harris Corner method<sup>[5]</sup>, which shows that most points distribute on the border of idiographic object in image.

But we have found that interest points mainly extract features distributing on the edge of objects, instead of paying attention to the placid parts of image, especially to big proportional images, which makes us neglect those idiographic objects inevitably.

In this paper, we propose an efficient method based on the composition of interest points and keyblocks to solve the feature representation problem, which has more comprehensive consideration to object border and local region of image.

## 2 Keyblock Generation

In this section, we present this method to generate many keyblocks (the construction of keyword-like image feature segments) of image and combining these keyblocks

with interest points during the course of image retrieval. Here we use clustering algorithms to generate keyblocks.

## 2.1 Keyblock generation for each semantic class

For each semantic class, a corresponding codebook (consists of the set of selected keyblocks) is generated. Each keyblock in these codebooks will retain its original pixel intensity values and will also carry a class label corresponding to the type of geographic feature it represents.

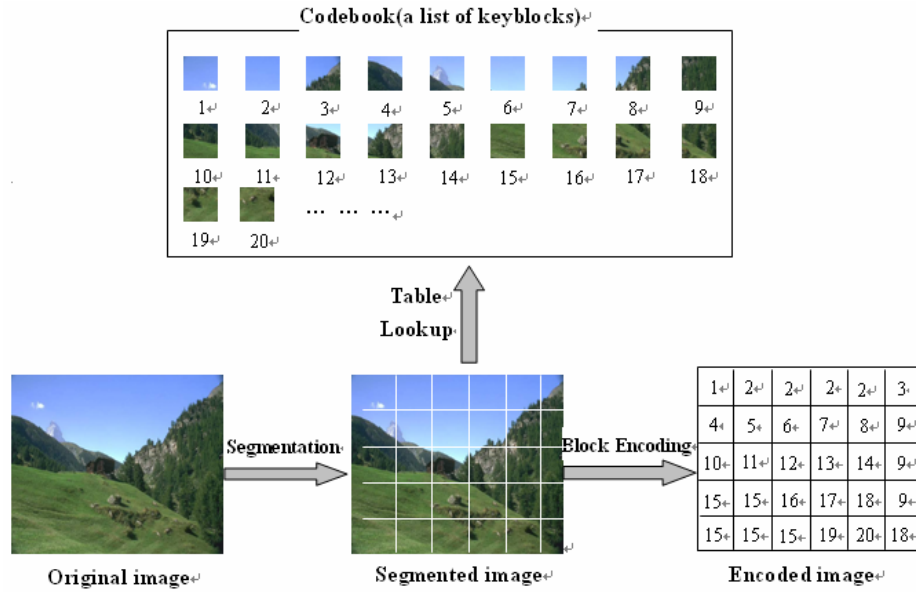


Fig. 1. General procedure for image encoding

Automatic method<sup>[6]</sup> is used in keyblock selection: For each semantic class, domain experts are asked to provide some training images to initiate the standard keyblock generation procedure presented as followed:

Let  $C = \{c_1, \dots, c_i, \dots, c_N\}$  be the "codebook" of keyblocks representing the images, where  $N$  is the codebook size and  $c_i$  ( $1 \leq i \leq N$ ) are the keyblocks. Let  $F$  be a mapping:

$$F: R^k \rightarrow C = \{c_1, \dots, c_i, \dots, c_N \mid c_i \in R^k\}$$

where  $R^k$  is the Euclidean space of dimension  $k$ . Given a sequence  $T = \{t_1, \dots, t_i, \dots, t_n \mid t_j \in R^k\}$ , the mapping  $F$  gives rise to a partition of  $T$  which consists of  $N$  cells  $P = \{p_1, \dots, p_i, \dots, p_N\}$ , where  $p_i = \{t \in T, F(t) = c_i\}$ . For a given distortion function  $d(t, c_i)$ , which is the distance between the input  $t_j$  and output code  $c_i$  (for example, the Euclidean distance), an optimal mapping should satisfy the following conditions:

—*Nearest Neighbor Condition*: For each  $p_i$ , if  $t \in p_i$ , then  $d(t, c_i) \leq d(t, c_j)$ , for all  $j \neq i$ .

—*Centroid Condition*: For a given partition  $P$ , the optimal code vectors satisfy:

$$c_i = \frac{\sum_{t \in p_i} t}{k_i}, 1 \leq i \leq N, \text{ where } k_i \text{ is the cardinality of } p_i$$

The purpose of this stage is to assign semantic meaning to each keyblock, since domain knowledge can be imported. There are a variety of clustering algorithms<sup>[7,etc.]</sup> available which can be applied to data sets of different types. Two commonly-used algorithms that can serve as the basis for this approach are the *Generalized Lloyd Algorithm* (GLA) and the *Pairwise Nearest Neighbor Algorithm* (PNNA)<sup>[8]</sup>.

## 2.2 Codebook merge

The codebooks generated in 2.1 are now merged into a larger codebook. This codebook comprises keyblocks with a range of meanings and can be directly used in image encoding and decoding. However, because the component keyblocks come from different training sets, and unsupervised clustering algorithms may have been employed in 2.1, there may be overlap between the boundaries of clusters centered on the keyblocks. The quality of this codebook will therefore be improved through the fine-tuning process in following 2.3.

## 2.3 Fine-tuning the codebook using learning vector quantization

Fine-tuning is performed in this section by using learning vector quantization (LVQ) algorithms [Kohonen et al. 1995].

The codebook generated in 2.2 will be used as the initial codebook for this process. Each keyblock and each training block has a class label. In each iteration of the clustering algorithm, updates are performed on those two data items (keyblocks)  $c_i$  and  $c_j$  which are nearest to a training input (training block)  $t$ . This update is performed when one of these data items belongs to the correct class while the other belongs to an incorrect class, and  $t$  falls within an update zone defined around the mid-plane of cluster boundaries formed by  $c_i$  and  $c_j$ . Assuming that  $d_i$  and  $d_j$  are the distances of  $t$  from  $c_i$  and  $c_j$ , respectively, this update zone is defined as the region where  $\min(d_i/d_j, d_j/d_i) > T$ , with  $T$  being a threshold with a typical value between 0.5 and 0.7.

At this point, each image is a two-dimensional array of codebook keyblock indices. It can also be considered as a list of keyblocks similar in format to a text document defined by a list of keywords. We can reconstruct the image to test whether the codebook was properly selected (see Fig.1).

## 3 Similarity calculation

In this paper we carry out similarity calculation based on histogram model usually used in information retrieval, which is a vector model with special form. Suppose  $D=\{d_j\}$  is a database which contains  $n$  images, and  $K=\{k_i\}$  is a codebook which

contains  $t$  codewords, the calculation method is described in [9]. We can adjust the value of weight appropriately in terms of our understanding of images during the course of image retrieval, in order to receiving the result we need.

## 4 Experiments and result

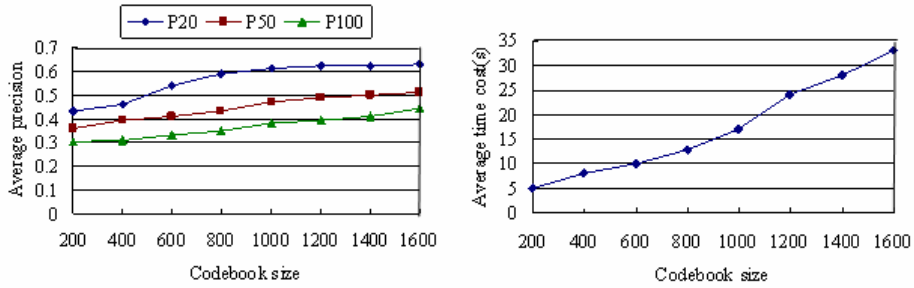


Fig. 2. Precision and time cost for different codebook sizes

In order to test our method, we select 5 categories images (automobile, animal, plane, ship, architecture) from COREL<sup>[10]</sup> database, which is classified by high-level semantics (defined by a large group of human observers as standard ground truth). Then we collect 100 images from each category randomly as the query set and denote the query set as  $QS = \{ I_1, \dots, I_{100} \}$ . These demonstration images constitute 500 times query in total. We choose the first 20 most similar images as retrieval results each time.

### 4.1 Determining the size of the codebook

To determine the size of the codebook, different numbers of clusters have been selected and evaluated. Considering that the value of  $k$ , i.e. the maximum number of the expanded codewords is related with  $N$ , i.e. the size of the codebook. A retrieved image is considered a match if it belongs to the same category of the query image. The average precisions within the top 20(50,100) images are shown in Fig.2 (a), and retrieval time costs for different codebook sizes are given in Fig.2(b).

### 4.2 Performance of experiments

As to each kind of images, its average Normal Precision is the average value of 5 query results, the same to average Normal Recall, as shown in Table 1:

The performance of experiments shows that the instance where  $\omega_1=1$  and  $\omega_2=0$ , which considers the similarity of keyblocks containing interest points plays best, which indicates that the keyblocks which interest points exist on make up the

shortcoming of information only from edges. Furthermore, the variability of  $\omega$  increases the flexibility of retrieval.

**Table 1.** The retrieval results for different  $\omega$  value

Images	Average Normal Precision (%)			Average Normal Recall (%)		
	$\omega_1=1,$ $\omega_2=0$	$\omega_1=0,$ $\omega_2=1$	$\omega_1=0.5,$ $\omega_2=0.5$	$\omega_1=1,$ $\omega_2=0$	$\omega_1=0,$ $\omega_2=1$	$\omega_1=0.5,$ $\omega_2=0.5$
Automobile	85.3	77.5	70.5	69.2	65.1	60.8
Animal	84.1	80.2	65.7	71.3	62.5	59.3
Plane	82.2	72.6	74.1	75.8	70.5	55.1
Ship	79.8	76.1	74.7	62.7	70.1	62.4
Architecture	83.0	81.8	79.3	70.4	63.3	66.9

## 5 Conclusion

The goal is to use keyblock analysis module by constructing codebook to increase the capability of the interest points analysis module by Harris for queries where the placid parts of image was important and carry out similarity calculation based on histogram model. Experiment results have shown some of the images retrieved by this system improved.

## References

1. Wang Xiangyang, Yang Hongying and Hu Fengli. A New Regions-of-Interest Based Image Retrieval Using DWT. Proceedings of ISCIT2005: 127-130.
2. C. Carson, Blobworld: Image segmentation using expectation-maximization and its applications to image querying. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2002, 24(8):1026- 1038.
3. J. Z. Wang, J. Li, G. Wiederhold. SIMPLcity: Semantics-sensitive integrated matching for picture libraries. IEEE Trans. Pattern Analysis and Machine Intelligence, 2001, 23(9): 947-963.
4. H. P. Moravec, towards automatic visual obstacle avoidance, IJCAI, 1977:584.
5. C. Harris, A combined corner and edge detector, in 4th ALVEY vision conference, 1988: 147-151.
6. LEI ZHU, AIBING RAO and AIDONG ZHANG, Theory of Keyblock-Based Image Retrieval, ACM Transactions on Information Systems, Vol. 20, No. 2, April 2002: 224–257.
7. ZHANG, T., An Efficient Data Clustering Method for Very Large Databases. ACM SIGMOD International Conference on Management of Data. Montreal, 1996:103–114.
8. ZHU I , keyblock: an approach for content-based image retrieval[A], ACM multimedia, 2000:157-166.
9. Hui LIU, Jun MA, Research on Image Retrieval based on Interest Points and Keyblocks, Journal of Computational Information Systems, 3(4),2007:1679-1685
10. Corel stock photo library, Ontario, Canada: Corel.