2011 Sixth International Conference on Image and Graphics

Image Annotation with Multiple Quantization

Guo Qiaojin, Li Ning, Yang Yubin and Wu Gangshan
National Key Laboratory for Novel Software Technology
Nanjing University
Nanjing 210093, China
Email: guoqiaojin@gmail.com, ln@nju.edu.cn, yyb@nju.edu.cn and gswu@nju.edu.cn

***Abstract*—Image annotation plays an important role in image retrieval and understanding. Various techniques have been proposed for assigning keywords to images. One of the most frequently used methods is to search annotated images with similar visual features, and keywords are transfered to new coming images. This leads to the problem of nearest neighbor search, which is a hot topic of pattern recognition, information retrieval, and data compression. In this paper we proposed a fast and effective method for retrieving similar images from large collections of annotated images. The proposed technique employs discrete cosine transform and regular lattice quantization to encode images and search similar images directly with the corresponding codes. This technique is evaluated on image annotation. Similar images are retrieved by utilizing our encoding strategy, and keywords are assigned by utilizing traditional label transfer mechanism. Experimental results show that our method provides competitive performance with traditional methods, and mean while provides one scalable framework for annotating large collections of image dataset.**

***Keywords*-Image Annotation; Quantization; DCT;**

I. Introduction

Automatic image annotation assigns metadata, usually keywords, to images automatically, makes it easier for indexing and maintaining large collections of images, plays an important role in image retrieval systems. It has been studied a lot in the last decades [1][2][3], various machine learning techniques were employed for learning the correspondence between visual features and keywords.

Current techniques of automatic image annotation can be summarized into two categories, learning based annotation and search based annotation. Learning based methods build the conditional or joint probabilistic models of keywords and visual features, such as Decision Tree [4], Random Forest [5], SVM [6], CMRM [7], MBRM [8] etc. Search based methods directly search similar images from traning images and labels are transfered to new images, such as AnnoSearch [9] and SIBA[10].

In this paper, we propose a fast and effective method for retrieving similar image from large set of training images. Each image is quantized into several codes and stored in database, similar images are retrieved by searching images with the same codes, and used for automatic image annotation.

The rest of this paper is organized as following. In Section 2, we discussed several related works. The proposed algorithm with multiple quantization is described in Sect. 3, and experimental results are shown in Sect. 4.

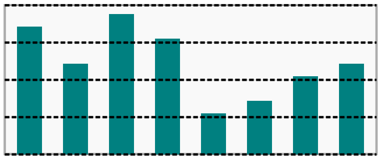


Figure 1. Vector quantization with regular lattice, each dimension of features is quantized into equal bins.

II. Related Work

The k-nearest neighbor (kNN) method is a simple while robust nonparametric classifier for image classification and image annotation, keywords are transferred from nearest neighbors to the test image, and this outperforms the current state-of-art on several large real-world datasets [11]. The drawback of kNN is that all the training images must be stored and requires heavy computation for the annotation of new images.

KD-tree proposed by Bentley [12] divides the training data into different partitions, makes it faster for finding similar instances. It works well on dataset with low dimensions, while KD-tree can not handling high-dimensional data, only subset of the features are used for constructing the tree structure and most of the features are neglected.

Vector quantization is a powerful technique for finding approximate nearest neighbors. A set of codes are generated based on training data, new coming data is encoded by the precomputed codebook and approximated nearest neighbors are extracted by the code. One of the most commonly used method used for vector quantization is k-means [13], which widely used in BoF models for image classification. David [14] built a vocabulary tree with 1M leaf nodes defines a hierarchical quantization by utilizing hierarchical k-means clustering , and works well on real time recognition of CD covers. Brian [15] also employs hierarchical k-means to build codebook by recursively partitioning the data set, and the final vocabulary is compressed by utilizing information bottleneck technique. While k-means suffers from heavy computation while building codebook, especially with large dataset.

978-0-7695-4541-7/11 $26.00 © 2011 IEEE
DOI 10.1109/ICIG.2011.15

Figure 2. Images with the same quantized codes ($N = 2$)

Want [16] quantized features into hash codes for large-scale duplicate detection, similarity of images are measured by the hamming distance of hash codes. This technique was further employed for searching similar images in AnnoSearch [9]. Tinne [17] quantized feature space with a regular lattice to build visual vocabulary, the quantized features are stored using hashing techniques , and achieved promising results in pixel-wise image classification.

III. Image Annotation with Multiple Quantization

Inspired by the work of Want [16] and Tinne [17], each dimension of visual features of images are quantized with regular lattice. Every image if encoded with a set of quantized codes, approximate nearest neighbor discovered without distance calculation for each data. Similar images are retrieved directly by finding training images with the same codes. The while procedure consists of two parts, first, encoding each image with multiple code, second, the storage and searching strategy for retrieve similar images from training dataset.

A. Regular Lattice Quantization

We first review the regular lattice quantization proposed by Want [16] and Tinne [17]. For searching similar images from training dataset, visual features are extracted from images, and each image is represented by a feature vector. Want [16] quantized every feature dimension into two bins after dimension reduction with PCA. Each dimension is splitted into two bins by the mean value. Visually duplicated images are grouped by utilizing hamming distance. Tinne [17] quantized each dimension into four bins. Vectors with D dimensions quantized into N bins will produce N^D codes, which is a huge space, while most of these codes are empty as demonstrated in [17]. The quantization method is shown in Fig. 1, every dimension is rescaled and quantized with regular lattice. The most frequently used codes are selected to build codebook.

Quantization with regular lattice provides a fast and effective method for maintaining and retrieving large collections of images. Similar images will have similar codes, and can be retrieved directly by using the quantized codes. While the quantized steps N is one important parameter which greatly affects the performance. Choosing small N is less discriminative and images of different scenes will have same codes. As shown in Fig. 2, these images comes from various scenes, such as moutain, city and group of people, while sharing the same quantized codes with $N = 2$. Choosing large N will cause exponentially increase of the codebook, and meanwhile separates similar images into different codes, which is more suitable for finding duplicate images.

B. Multiple Quantization

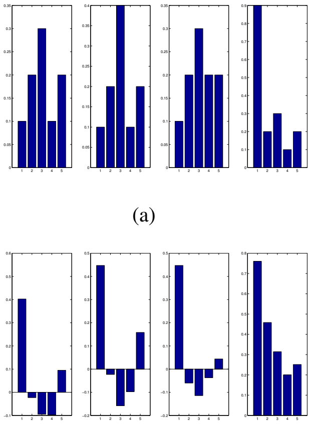
Multiple quantization is designed to fix the problem of regular lattice quantization shown above. Instead of quantizing each image into one single code vector, we generate a set of codes for every image. This leads to less duplication in large dataset. Mean while, similar images are found by searching images with the same code vector directly, without calculating hamming distance.

For encoding with multiple code, we need to represent every image with several different feature vectors with different resolutions. Vectors with coarse resolution contain basic information of images, while details are included in vectors with finer resolutions. Similar images could be found by searching feature vectors with resolutions from low to high. In this paper, we utilize Discrete Cosine Transform (DCT) [18] to convert initial feature vectors into sets of vectors with multiple resolution.

DCT separate signal or image into cosine functions of different frequencies. It is a widely used technique for image compression, such as JPEG-2000. Equation (1) show the detail of one dimensional DCT. Signal $x(n)$ with length N can be represented by sum of cosine functions with spectral coefficients $C(k)$.

$$\begin{aligned}
C(k) &= \alpha(k) \textstyle\sum_{n=0}^{N-1} x(n) \cos[\frac{\pi(n+\frac{1}{2})k}{N}] \quad . \\
x(k) &= \textstyle\sum_{k=0}^{N-1} \alpha(k) C(k) \cos[\frac{\pi(n+\frac{1}{2})k}{N}] \quad . \\
\alpha(k) &= \begin{cases} \sqrt{\frac{2}{N}} & \text{if } k = 0 \ . \\ \sqrt{\frac{1}{N}} & \text{otherwise} \ . \end{cases}
\end{aligned} \tag{1}$$

Figure 3 show four feature vectors $x(n)$ and the correspondence spectral coefficients $C(k)$. Feature vectors are decomposed into sum of cosine function of different frequency, the coefficients called DCT coefficients. Features can be reconstructed by these coefficients depicted in Fig. 3(b). As shown in Fig. 3, First three feature vectors are similar and also have similar DCT coefficients.



(a)

(b)

Figure 3. Feature vectors (a) and the correspondence DCT coefficients (b)

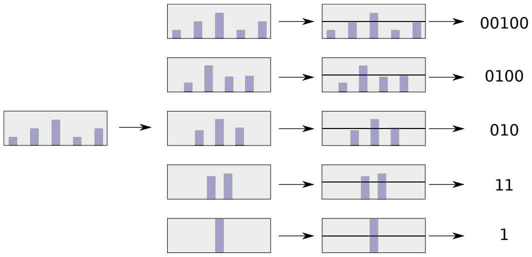


Figure 5. Flowchart of multiple quantization, each feature vector is compressed with different resolutions.

DCT coefficients of lower frequencies are much more significant than higher frequencies. Spectral information with high frequencies contains the details of feature vectors, while basic information are contained in the coefficients of lower frequencies. As a result, high frequencies is usually disregarded for compression. Figure 4 shows the compressed feature vectors of Fig. 3(a), which are reconstructed by inverse DCT with different low-pass filter.

Our target is to find similar feature vectors quickly with vector quantization. Quantizing single vector for each instance may face problems shown in Fig. 2. In this paper we represent every instance with different levels of compressed feature vector. Feature vectors with different resolutions are generated by utilizing DCT, for each vector, every dimension is quantized into 2 bins, such that every instance is represented by a group of codes. The whole procedure is shown in Fig. 5.

C. Storage and Searching

Tinne [17] exploit the fact that most of the bins after a direct discretization of the feature space are empty , and utilize hashing technique to store the unempty codes and correspondent labels. In this paper, we directly store the codes and correspondent image in database. Each image, named I_i, is quantized into a set of codes c_i^j, $j = 1, ..., D$, with D the dimension of features. Each pair $< c_i^j, I_i >$ is stored in database, and codes are used for index by utilizing hashing. Quantizatizing all the training images takes $O(MD)$, with M the number of training images. This takes $O(\log(M))$ for retrieving images with each specified code, and takes $O(\log(M)D)$ with multiple quantization. KNN does not need training, while takes $O(MD)$ for calculating the distance of test image and each image in training dataset.

For each test image, the quantized codes are first generated, and corresponding images are retrieved by search these codes from database. The most frequently k images found are kept as the nearest neighbors of test image.

IV. Performance Evaluation

A. Dataset

In this paper, we utilize the IAPR TC-12[19] for evaluation. This benchmark consists of 20,000 natural images taken from locations around the world, including pictures of sports, people, animals, cities, landscapes and many other aspects. Each image is initially associated with a text caption in English, German and Spanish. Keywords are extracted using the TreeTagger part-of-speech tagger [11]. The whole dataset consists of 17,825 images for training and 1,980 for testing, with a dictionary size of 291.

B. Visual Features

For every training image, We extract color features from images in RGB and HSV color space respectively for annotation. Histograms are generated by quantizing every channel into pre-defined number of bins. In this paper, every channel are splitted into 8 bins. We also extract histogram of gradients (HOG)[20] of every image, with 16 directions. Every image is represent by one vector with length D.

C. Evaluation Criterion

For evaluating the performance of proposed quantization method, we compared with kNN on image annotation. Similar images are retrieved with kNN and proposed quantization method, keywords of these images vote for annotation equally, and top 5 keywords are assigned to each test image. We compared mean precision and recall of annotation on IAPR TC-12 dataset, as well as the number of total keywords recalled (N+). Precision is defined as the ratio of retrieved positive images to the total number retrieved, recall is defined as the ratio of the number of retrieved positive images to the total number of positive images in the training set, and N+ denotes the number of recalled keywords.

D. Experimental Results

Figure 6 and Figure 7 depict the searched results by using kNN and proposed method on IAPR TC-12 dataset. The first column is the test image and the remaining columns

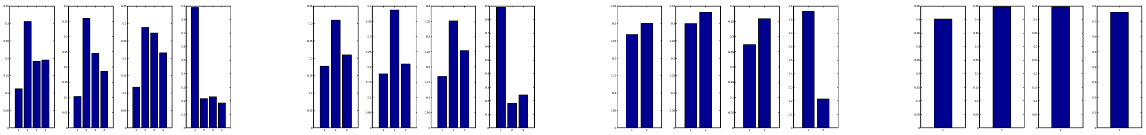

(a) (b) (c) (d)

Figure 4. Compressed feature vectors with different resolutions

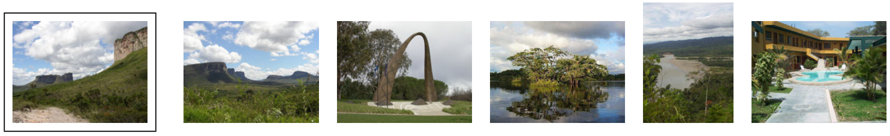

(a) (b) (c) (d) (e) (f)

Figure 6. Nearest neibghbor searh results using knn, the first column is the test image and the rest are search results.

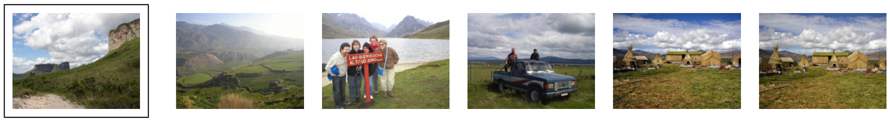

(a) (b) (c) (d) (e) (f)

Figure 7. Nearest neibghbor searh results using multiple quantization, the first column is the test image and the rest are search results.

Table I
ANNOTATION PERFORMANCE OF MULTIPLE QUANTIZATION(MQ) AND KNN

k	Precision		Recall		N+	
	kNN	MQ	kNN	MQ	kNN	MQ
5	30.29%	25.50%	23.28%	20.84%	186	196
10	33.97%	29.03%	25.38%	23.23%	168	175
15	36.33%	32.64%	25.91%	24.24%	147	170
20	38.84%	35.69%	26.23%	24.75%	134	162
25	41.22%	37.61%	26.20%	25.22%	124	156
30	42.97%	39.45%	26.08%	25.25%	117	156
40	46.67%	43.13%	26.32%	25.51%	104	145
50	48.63%	45.63%	25.75%	25.65%	96	139
100	54.32%	50.56%	25.37%	25.37%	66	120

are searched neighbors by using different algorithms. Images retrieved with multiple quantization contains different details while share similar background, While kNN provides exactly nearest neighbors from current feature space.. For example, people and car appear in Fig. 7(c) and Fig. 7(d), which are different with Fig. 7(a), while all the retrieved images share the same background, such as “sky“, “cloud“ and “vegetations“, which included in the dictionary of IAPR TC-12 dataset.

We compared the annotation performance of kNN and multiple quantization (MQ) with different number of k. The experimental results are shown in Table 1. Each row of Table 1 shows the performance of two algorithms with different k. Both precision and recall increase with the increment of k, while N+ decreases with the increment of retrieved similar images. Baseline method gets slightly better precision and recall than proposed method with the same k. This should be caused by difference of approximate and exact nearest neighbor search strategy, kNN gets the exact nearest neighbors of test image while our method gets approximate neighbors, shown as Fig. 6 and Fig. 7. While the proposed method achieved better performance while using k larger than baseline method. With the increment of k, the proposed method gets better N+ than baseline method, which means that less keywords are disregarded for improving precision and recall. Mean while, the time cost is much less than baseline with the increment of training data.

V. CONCLUSIONS AND FUTURE WORKS

We proposed a new model for image annotation by integrating DCT and regular lattice quantization. Training images are quantized with regular lattices and stored in database. Images quantized into the same bins are supposed to have similar annotations. Our model provides fast and effective scheme for annotation with large collections of training images. While in this paper, the similar images are searched by comparing global visual features, finding images with globally similarity needs large scale of training images. Better performance could be achieved by utilizing local similarity, which will be evaluated in our future work.

Acknowledgment

This work is supported by the "973" Program of China (Grant No. 2010CB327903), the National Natural Science Foundation of China (Grant Nos. 60875011, 60723003), and the Key Program of Natural Science Foundation of Jiangsu Province, China (Grant BK2010054).

References

[1] C. Tsai and C. Hung, "Automatically annotating images with keywords: A review of image annotation systems," *Recent Patents on Computer Science*, vol. 1, no. 1, pp. 55–68, 2008.

[2] A. Hanbury, "A survey of methods for image annotation," *Journal of Visual Languages & Computing*, vol. 19, no. 5, pp. 617–627, 2008.

[3] Y. Liu, D. Zhang, G. Lu, and W. Ma, "A survey of content-based image retrieval with high-level semantics," *Pattern Recognition*, vol. 40, no. 1, pp. 262–282, 2007.

[4] J. Huang, S. Kumar, and R. Zabih, "An automatic hierarchical image classification scheme," in *Proceedings of ACM International Conference on Multimedia*, pp. 219–228, 1998.

[5] A. Bosch, A. Zisserman, and X. Muoz, "Image classification using random forests and ferns," in *Proceedings of IEEE International Conference on Computer Vision*, pp. 1 –8, 2007.

[6] C. Cusano, G. Ciocca, and R. Schettini, "Image annotation using svm," in *Proceedings of SPIE*, vol. 5304, pp. 330–338, 2004.

[7] J. Jeon, V. Lavrenko, and R. Manmatha, "Automatic image annotation and retrieval using cross-media relevance models," in *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pp. 119–126, 2003.

[8] S. Feng, R. Manmatha, and V. Lavrenko, "Multiple bernoulli relevance models for image and video annotation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2004.

[9] X. Wang, L. Zhang, F. Jing, and W. Ma, "AnnoSearch: Image Auto-Annotation by Search," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 1483–1490, 2006.

[10] C. Wang, F. Jing, L. Zhang, and H. Zhang, "Scalable search-based image annotation of personal images," in *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pp. 269–278, 2006.

[11] A. Makadia, V. Pavlovic, and S. Kumar, "A new baseline for image annotation," in *Proceedings of European Conference on Computer Vision*, pp. 316–329, 2008.

[12] J. Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, 1975.

[13] S. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 2002.

[14] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 2161–2168, 2006.

[15] B. Fulkerson, A. Vedaldi, and S. Soatto, "Localizing objects with smart dictionaries," in *Proceedings of European Conference on Computer Vision*, pp. 179–192, 2008.

[16] B. Wang, Z. Li, M. Li, and W. Ma, "Large-scale duplicate detection for web image search," in *Proceedings of IEEE International Conference on Multimedia and Expo*, pp. 353–356, 2006.

[17] T. Tuytelaars and C. Schmid, "Vector quantizing feature space with a regular lattice," in *Proceedings IEEE International Conference on Computer Vision*, pp. 1–8, 2007.

[18] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transfom," *IEEE Transactions on Computers*, vol. 100, no. 1, pp. 90–93, 2006.

[19] M. Grubinger, P. Clough, H. Muller, and T. Deselaers, "The iapr tc-12 benchmark: A new evaluation resource for visual information systems," in *Proceedings of International Workshop OntoImage*, pp. 13–23, 2006.

[20] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 886–893, 2005.