Near Duplicate Image Detecting Algorithm based on Bag of Visual Word Model

Zhaofeng Li and Xiaoyan Feng
College of Information Engineering Henan Institute of science and technology, Henan Xinxiang, China
Email: Zhfengli@126.com, Fxy@hist.edu.cn

***Abstract*—In recent years, near duplicate image detecting becomes one of the most important problems in image retrieval, and it is widely used in many application fields, such as copyright violations and detecting forged images. Therefore, in this paper, we propose a novel approach to automatically detect near duplicate images based on visual word model. SIFT descriptors are utilized to represent image visual content which is an effective method in computer vision research field to detect local features of images. Afterwards, we cluster the SIFT features of a given image into several clusters by the K-means algorithm. The centroid of each cluster is regarded as a visual word, and all the centroids are used to construct the visual word vocabulary. To reduce the time cost of near duplicate image detecting process, locality sensitive hashing is utilized to map high-dimensional visual features into low-dimensional hash bucket space, and then the image visual features are converted to a histogram. Next, for a pair of images, we present a local feature based image similarity estimating method by computing histogram distance, and then near duplicate images can be detected. Finally, a series of experiments are constructed to make performance evaluation, and related analyses about experimental results are also given.**

***Index Terms*—Near Duplicate Image, Visual Word Model, SIFT, Hash Function, Locality Sensitive Hashing**

I. INTRODUCTION

With the rapid development of the Internet, and the availability of image capturing devices such as digital cameras, image scanners, the size of digital image collections is increasing rapidly. Efficient image retrieving, browsing and retrieval tools are required by users from various domains, including remote sensing, fashion, crime prevention, publishing, medicine, architecture and so on. For this purpose, many general purpose image retrieval systems have been developed. There are two main types: 1) Text-based image retrieval and 2) Content-based image retrieval. The text-based method can be tracked back to 1970s. In such systems, the images are manually tagged by text descriptors, which are then utilized by a database management system to implement image retrieval [1] [2].

On the other hand, the content-based image retrieval refers to the technology which in principle helps to organize digital picture archives by their visual content. With this definition, anything ranging from an image similarity function to a robust image annotation engine falls under the purview of content-based image retrieval. This characterization of content-based image retrieval as a field of study places it at a unique juncture within the scientific community [3] [4]. While we witness continued effort in solving the fundamental open problem of robust image understanding, we also see people from different fields, such as, computer vision, machine learning, information retrieval, human-computer interaction, database systems, Web and data mining, information theory, statistics, and psychology contributing and becoming part of the content-based image retrieval community [5] [6].

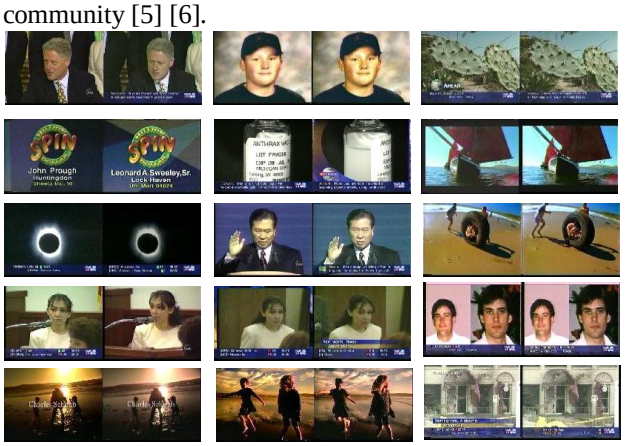

Figure 1. Samples of near-duplicate images

However, with the rapid increment of images on the World Wide Web, there are a great number of near duplicate images in recent years and these images reduce the efficiency of content-based image retrieval. The definition of a near duplicate image varies depending on what photometric and geometric variations are deemed acceptable. The application ranges from exact duplicate detection where no changes are allowed to a more general definition that requires the images to be of the same scene, but with possibly different viewpoints and illumination. As is shown in Fig. 1, several samples of near-duplicate images are given. Detecting near duplicate images in large databases should meet two challenging constraints. Firstly, for a given image, only a small amount of data can be stored in computer memory. Secondly, queries must be very cheap to evaluate. Ideally, enumerating all the duplicates of an image should have complexity close to linear in the number of duplicates returned [7-10].

doi:10.4304/jmm.8.5.557-564

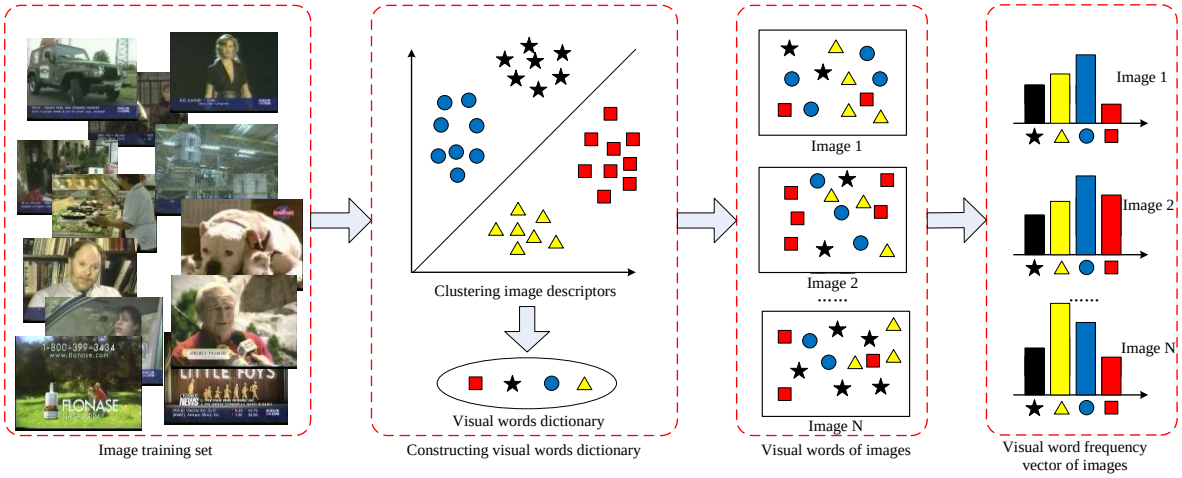

Figure 2. Model of visual word model

In recent years, the research field of near duplicate image detecting is a hot topic for image vision and machine learning. In the following parts, we will analyze the related works of the near duplicate image detecting research.

Zheng et al. present a framework for near-duplicate image detection in a visually salient Riemannian space. A visual saliency model is first used to identify salient regions of the image and then the salient region covariance matrix of various image features is computed. salient region covariance matrix, which lies in a Riemannian manifold, is used as a robust and compact image content descriptor [11].

In paper [12], the authors summarized the extensive work on web image annotation using the large-scale metadata and social information available on the Web, and introduced a system named Arista system, which is a nonparametric image annotation platform built upon two billion web images. Afterwards, the authors propose a highly efficient and scalable duplicate-search technique so that the Arista system can be deployed on a few servers. A few interesting applications such as building large-scale celebrity face database and text-to-image translation are also presented in this paper.

Sinjur proposed a novel, fast algorithm for generating the convex layers on grid points with linear time complexity. Convex layers are extracted from the binary image. The obtained convex hulls are characterized by the number of their vertices and used as representative image features. Hence, a computational geometric approach to near-duplicate image detection stems from these features [13].

In order to solve the problem of traditional near-duplicate image search systems mostly building on the bag-of-local features representation, Xie et al. proposed a novel framework by using graphics processing units. The main contributions of their method lie in the following aspects: (1) A new fast local feature detector coined Harris-Hessian is designed according to the characteristics of GPU to accelerate the local feature detection. (2) The spatial information around each local feature is incorporated to improve its discriminability, supplying semi-local spatial coherent verification. (3) A new pairwise weak geometric consistency constraint algorithm is proposed to refine the search results [14].

Cho et al. present a concentric circle-based Image signature which makes it possible to detect near-duplicates rapidly and accurately An image is partitioned by radius and angle levels from the center of the Image Feature values are calculated using the average or variation between the partitioned sub-regions The feature values distributed in sequence are formed Into an Image signature by hash generation The hashing facilitates storage space reduction and fast matching [15].

Zhou et al. proposed a novel geometric coding algorithm to encode the spatial context among local features for large-scale partial-duplicate Web image retrieval. The proposed geometric coding was made up of geometric square coding and geometric fan coding, which describe the spatial relationships of SIFT features into three geo-maps for global verification to remove geometrically inconsistent SIFT matches [16].

The main innovations of this paper lie in the following aspects:

(1) SIFT descriptor and visual word model are used to represent image visual features.

(2) In order to reduce the time cost of the whole computing process, locality sensitive hashing is utilized to map high-dimensional visual features into low-dimensional hash bucket space.

(3) The task of image visual similarity computing is converted to histogram distance estimating.

(4) Histogram distance is solved by seeking the minimum difference of pair assignments between the two sets. Particularly, this process is implemented by determining the best one-to-one assignment between two sets such that the sum of all differences between two individual elements in a pair is minimized.

The rest of the paper is organized as the following sections. Section 2 introduces the visual word model. Section 3 illustrates the proposed scheme for near duplicate images detecting. In section 4, a series of experiments are designed and implemented to make performance evaluation. Finally, we conclude the whole paper in section 5.

II. PROPOSED SCHEME

A. Overview of the Visual World Model

The bag-of-visual-words model has been widely used in the field of computer vision. Particularly, there are three main phases for the model: 1) Extracting local feature descriptors (such as SIFT), 2) Quantizing local descriptors into a codebook, 3) Converting images to a set of visual words [17] [18].

As shown in Fig. 2, the structure of visual word model includes four parts: 1) Constructing image training set, 2) Constructing visual words dictionary, 3) Selecting visual words of images and 4) Obtaining visual word frequency vector of images

Fig. 2 shows the framework of our proposed visual word model. The key idea of this model is to quantize the continuous high-dimensional space of image features (such as SIFT features [19] [20]) to a specific vocabulary for visual words. This process is implemented through clustering the SIFT features which is chosen from a large-scale image dataset into several clusters by the K-means algorithm. Particularly, the centroid of each cluster is used as a visual word in the given vocabulary. Afterwards, local features are extracted and then allocated onto the closest visual word. Next an image can be represented as a histogram of visual words, based on the number of visual words for each class.

B. Algorithm Description

Following by the recent progress in object recognition, we can represent the images by local interest point descriptors through the scale-invariant feature transform, which is an effective method in computer vision research field to detect local features of images. SIFT descriptors can produce hundreds of feature points. The SIFT feature of each keypoint is made up of a 128-dimensional vector. However, as the curse of dimensionality, a more efficient approach is to reduce the dimension of the original data. To solve the problem of curse of dimensionality, we utilize the locality-sensitive hashing algorithm which is an approximate kNN algorithm to index the local descriptors.

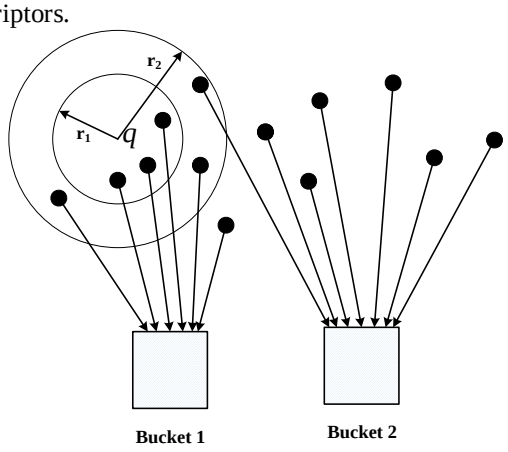

Figure 3. Structure of locality sensitive hashing

As is described in Wikipedia, locality-sensitive hashing is a method of performing probabilistic dimension reduction of high-dimensional data. The main idea is to hash the input items so that similar items are mapped to the same buckets with high probability. This process is different from the conventional hash functions, such as those used in cryptography as in this case the goal is to maximize probability of "collision" of similar items rather than avoid collisions. Particularly, the structure of locality sensitive hashing is shown in Fig. 3, the data sample are mapping to buckets through the locality sensitive hashing algorithm.

Next, the definition of locality sensitive hashing is given at first. For a domain S of the items set with distance measure D, a locality sensitive hashing family is defined as:

Definition 1. A family $\mathrm{H}=\{h: S \to U\}$ is called (r_1, r_2, p_1, p_2)-sensitive for D if for any $v, q \in S$

if $v \in B(q, r_1)$ then $\mathrm{P}_H[h(q)=h(v)] \geq p_1$

if $v \notin B(q, r_2)$ then $\mathrm{P}_H[h(q)=h(v)] < p_2$

where $D(\bullet)$ be a distance function of elements from a set S, and for any $p \in S$, let $B(q,r)$ denote the set of elements from S within the distance r from p. In order for a locality sensitive hash family to be useful, it has to satisfy inequalities $p_1 > p_2$ and $r_1 < r_2$.

One of the main application of locality sensitive hashing is to provide efficient nearest neighbor search algorithms. Consider any locality sensitive hashing family H. The algorithm has two main parameters, which are the width parameter k and the number of hash tables L. In the first step, we define a new family G of hash functions g, where each function set g is obtained by concatenating k functions $h_1, h_2, \cdots, h_k$ from family G. For an input item p, we have $g(p)=[h_1(p), \cdots h_k(p)]$, that is to say, a random hash function g is obtained by concatenating k randomly chosen hash functions. The algorithm then constructs L hash tables, each corresponding to a different randomly chosen hash function g. Formally, for a fixed a and b, the hash function $h_{a,b}(v)$ is defined as.

$$h_{a,b}(V)=\left\lfloor \frac{a \cdot V + b}{r} \right\rfloor \tag{1}$$

The hash function $h_{a,b}(V): R^d \to N$ maps a d dimensional vector V onto a set of integers. Each hash function in the family is indexed by a choice of random a and b, where a is a d-dimensional random vector with entries chosen independently from a Gaussian distribution and b is a real number chosen uniformly from the range $[0, r]$. r defines the quantization of the features and V is the original feature vector. After the hashing process, initial vectors from a higher dimensional vector space could be mapped to a discrete subspace of lower dimension.

For a pair of images I_i and I_j, our local feature based image similarity measuring method can be summarized as the following five steps.

Step 1: Extracting the SIFT descriptors from I_i and I_j, and utilizing the visual word model to represent the SIFT descriptors of the given image.

Step 2: Constructing L LSH hash tables, and each hash table is corresponding to a set of hash functions which is defined in Eq.1, which is very simple to implement.

Step 3: For each hash table ht_l ($1 \le l \le L$), k hash functions of which are represented as $g_l = \{h_1^l, h_2^l, \cdots, h_k^l\}$.

Step 4: For a hash table generated from step 2, SIFT descriptors are mapped to buckets by hash functions which are related the hash table.

Step 5: Following step 4, for a hash table, a histogram is obtained by the way that each bin is corresponding to a bucket of the hash table.

After executing the above steps, each image can be represented as L histograms. Furthermore, local features based visual similarity of a pair of images is computed by estimating the distance between histograms. Particularly, we denote the histogram set of image I_i as $HG(i) = \{t(i)^l \,|\, 1 \le l \le L\}$, where $t(i)^l$ is the l^{th} histogram of image I_i.

As buckets in the given histogram are orderless, we argue that the nominal type is more suitable. Hence, the nominal type based histogram distance estimating method is introduced in our work.

Some notations and symbols should be defined beforehand. An image could be represented by a set of SIFT descriptors, that is, $I = \{s_1, s_2, \cdots, s_n\}$. Let b be a measurement, or feature, which can have one of m values contained in the set, $B = \{b_0, b_1, \cdots b_m\}$ and $s_i \in B$ is satisfied. Hence, the corresponding histogram of image I is denoted as $t(I) = [t_0(I), t_1(I), \cdots, t_{m-1}(I)]$, where $t_j(I)$ is the number of elements which is allocated to the j^{th} bin of histogram $t(I)$.

$$t_j(I) = \sum_{v}^{n} \beta_{uv} \quad \text{where} \quad \beta_{uv} = \begin{cases} 1 & \text{if } s_v = b_u \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The distance between histogram U and V can be considered as the problem of finding the minimum difference of pair assignments between the two sets. The key issue is to determine the best one-to-one assignment between two sets such that the sum of all differences between two individual elements in a pair is minimized. Supposing there are n elements in histogram U and V respectively, minimum difference of pair assignments of U and V is defined as follows.

$$\Theta(U,V) = \min_{X,Y}\left(\sum_{i,j=0}^{n-1} d(x_i, y_i)\right) \tag{3}$$

where $x_i \in U$ and $y_i \in V$. $d(\cdot)$ denotes value of difference between two elements.

$$d(x_i, y_j) = \begin{cases} 0 & \text{if } x_i \text{ and } y_j \text{ belonged to same bin} \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

There is little possibility that the number of key points of two images are equal to each other in practice. Therefore, we use least common multiple to make the number of elements in the two histogram equal.

$$t_k^*(I_p) = \frac{\kappa(N_p, N_q)}{N_p} \cdot t_k(I_p) \tag{5}$$

$$t_k^*(I_q) = \frac{\kappa(N_p, N_q)}{N_q} \cdot t_k(I_q) \tag{6}$$

where the function κ is used to compute the least common multiple of N_p and N_q. Therefore, for a pair of images I_p and I_q, the local feature based image similarity can be computed by histogram sets of the two images as follows.

$$S(I_p, I_q) = \frac{1}{L} \cdot \sum_{l=1}^{L} \frac{\Theta(t^*(I_p)^l, t^*(I_q)^l)}{\kappa(N_p, N_q)} \tag{7}$$

From Eq. 7, we can see that $S(I_p, I_q) \in [0,1]$ is satisfied. Afterwards, we set a threshold δ to determine whether a pair of images are near-duplicate or not. Images I_p and I_q are deemed as near-duplicate to each other, only if $S(I_p, I_q) \ge \delta$.

III. EXPERIMENTS

To test the effectiveness of the proposed near duplicate image detecting method, we design a series of experiments on several image datasets which are used for near duplicate image detecting.

A. Datasets and Performance Evaluation Criteria

To evaluate our approach more objectively and accurately, we construct three large-scale image datasets, which are illustrated as follows.

Dataset 1. INRIA CopyDays dataset [23]

The INRIA CopyDays dataset includes 157 original images which containing a variety of scene types, such as natural, man-made, water, sky, etc. The SYSU_Test contains 977 images randomly chosen from an image collections which contains different image types. These images are downloaded from the Flickr website. In the original INRIA CopyDays dataset [23], there are three main types of transformations, which are 1) Image resizing followed by JPEG compression ranging from JPEG3 to JPEG75, 2) Cropping ranging from 5% to 80% of the image surface, 3) Strong transformations: print and scan, paint, change in contrast, perspective effect, blur, very strong crop and so on.

The former two types of transformations are conducted on each test image. The paper [23] only produced 229 transformed images for the strong transformation. To obtain an overall performance evaluation for the proposed

algorithm, we extend the transformation types of the original image dataset.

Dataset 2. Social images collected from Flickr website.

For Flickr, from a strictly technical point of view, groups are collections of users who select to join such a community of which the members share the same interests. This dataset includes 1000 images collected from 5 Flickr groups, which are 1) "Beautiful, Just Beautiful (P1/C1)", 2) "Photography with Bokeh", "PUPPIES AND POODLES", "Lifehacker Desktop Show and Tell" and "Pet Parade". We collect 200 photos from the above each Flickr group. The ground truth images are obtained through clustering the images from personal image album, two images belonged to the same cluster are considered as near-duplicate images for each other.

Dataset 3. Images collected from Google image search engine through particular queries.

For the third dataset, we use Google image search engine to collect near duplicate images, which use text metadata associated with images to search images relevant to specific query. Two kinds of queries are used to create this dataset including trademark related images and landmark related images. Next, We submit ten trademark and ten landmark to Google image search engine respectively. The trademarks and landmarks utilized to construct Dataset 3 is shown in Table.1 as follows.

For each class, we select top 1,000 images returned by Google image search engine. For a given image category near-duplicate images are detect within one image category by our algorithm. Particularly, Dataset 3 are divided into two sub-classes which are: 1) Dataset 3-Trademark and 2) Dataset 3-Landmark.

In order to evaluate the performance of our proposed near-duplicate detection method, we use Precision, Recall and F-measure. For a given mage dataset i (which is named as d_i), the number of near-duplicate image pairs found in dataset d_i is represented as $N_f(d_i)$ and the number of ground truth near-duplicate image pairs in dataset d_i is represented $N_g(d_i)$. The precision and recall for dataset d_i is then defined in the following equation.

TABLE I. TRADEMARKS AND LANDMARKS UTILIZED IN DATASET 3.

Category	Name
Trademark	BMW, Coca-cola, Sony, Puma, Kodak, Nescafe, Starbucks, Toyota, Pierre Cardin, MIZUNO
Landmark	Sydney Opera House, Temple of the golden pavilion, Bosphorus Bridge, Brandenburg gate, Big Ben, Vienna golden hall, Torre di Pisa, Potala Palace, National Aquatics Center of China, Leaning Tower of Pisa

$$P(d_i) = \frac{\left|N_f(d_i) \cap N_g(d_i)\right|}{\left|N_f(d_i)\right|} \tag{8}$$

$$R(d_i) = \frac{\left|N_f(d_i) \cap N_g(d_i)\right|}{\left|N_g(d_i)\right|} \tag{9}$$

In order to calculate the weighted harmonic mean of precision and recall, the F-measure metric is proposed as follows.

$$F1 = \frac{2 \times P(d_i) \times R(d_i)}{P(d_i) + R(d_i)} \tag{10}$$

Afterwards, another well-known evaluation method which is named mean average precision (MAP) is utilized. The method of MAP computing is shown as follows.

$$AP = \frac{1}{N} \cdot \frac{\sum_{j=1}^{i} r_i}{i} \tag{11}$$

$$MAP = \frac{\sum_{q=1}^{U} AP(q)}{U} \tag{12}$$

where the value of r_i is set to one only if image i is related to topic, otherwise it is set to zero, and U refers to the number of queries.

B. Experimental Results and Related Analysis

Firstly, we will test mean average precision in Dataset 1, to make performance comparison, we compare the proposed scheme with other schemes. Other schemes are use other image descriptors to detect near duplicate images, including 1) Color-based descriptor(Color) [24], 2) Radon-based descriptor(Radon) [25], 3) GIST [26] 4) Bag-of-feature representation (BOF) [27], 5) VLAD [28].

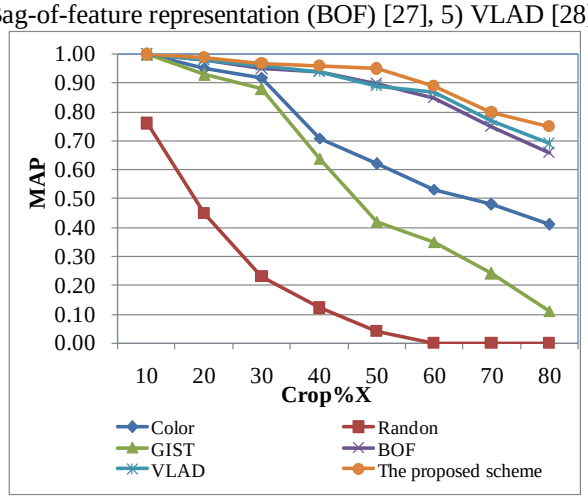

Figure 4. MAP evaluation for different descriptors with Crop%X changing

From Fig. 4 and Fig. 5, we can see that the proposed scheme has better performance under the criteria MAP, the reason lies in that in our scheme SIFT descriptors are mapping to hash buckets by the Locality sensitive hashing algorithm. This method can not only effectively describe image visual contents, but also accurately detect salient objects in images.

TABLE II. PERFORMANCE OF THE PROPOSED SCHEME

Image category	P	R	F1	Image category	P	R	F1
Beautiful, Just Beautiful (P1/C1)	0.661	0.606	0.632	Pierre Cardin	0.780	0.764	0.772
Photography with Bokeh	0.798	0.874	0.834	MIZUNO	0.764	0.805	0.784
PUPPIES AND POODLES	0.761	0.737	0.749	Sydney Opera House	0.720	0.771	0.744
Lifehacker Desktop Show and Tell	0.764	0.762	0.763	Temple of the golden pavilion	0.720	0.668	0.693
Pet Parade	0.844	0.771	0.806	Leaning Tower of Pisa	0.733	0.787	0.759
Kodak	0.787	0.728	0.757	Brandenburg gate	0.785	0.735	0.760
Coca-cola	0.726	0.777	0.751	Vienna golden hall	0.656	0.610	0.632
Sony	0.667	0.654	0.660	Big Ben	0.801	0.803	0.802
Puma	0.789	0.860	0.839	Torre di Pisa	0.659	0.722	0.689
BMW	0.787	0.864	0.844	Potala Palace	0.730	0.673	0.700
Nescafe	0.843	0.829	0.836	National Aquatics Center of China	0.726	0.711	0.719
Starbucks	0.762	0.689	0.723	Bosphorus Bridge	0.658	0.713	0.684
Toyota	0.663	0.631	0.647				

TABLE III. PERFORMANCE OF THE PROPOSED SCHEME WITH DIFFERENT PARAMETER δ

δ	Dataset 1			Dataset 2			Dataset 3-Trademark			Dataset 3-Landmark		
	P	R	F	P	R	F	P	R	F	P	R	F
0.8	0.846	0.763	0.802	0.830	0.736	0.780	0.893	0.801	0.845	0.936	0.880	0.907
0.7	0.817	0.807	0.812	0.812	0.785	0.798	0.877	0.825	0.850	0.918	0.890	0.904
0.6	0.798	0.830	0.814	0.776	0.825	0.800	0.873	0.861	0.867	0.916	0.903	0.909
0.5	0.735	0.856	0.791	0.718	0.854	0.780	0.846	0.864	0.855	0.906	0.933	0.919
0.4	0.725	0.880	0.795	0.709	0.845	0.771	0.825	0.882	0.853	0.862	0.958	0.907
0.3	0.668	0.884	0.761	0.663	0.863	0.750	0.798	0.905	0.848	0.848	0.968	0.904
0.2	0.623	0.895	0.735	0.653	0.879	0.749	0.776	0.917	0.841	0.833	0.976	0.899

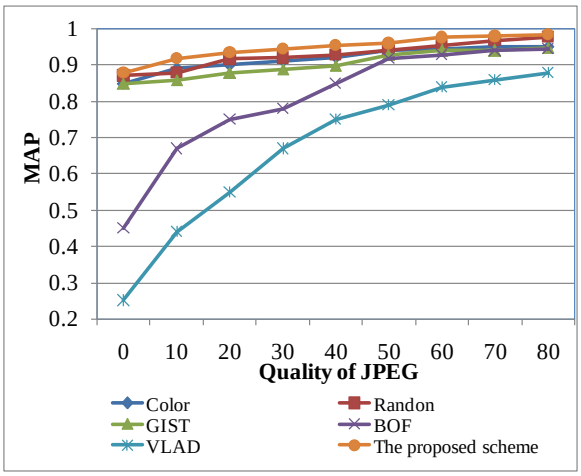

Figure 5. MAP evaluation for different descriptors with Quality of JPEG changing

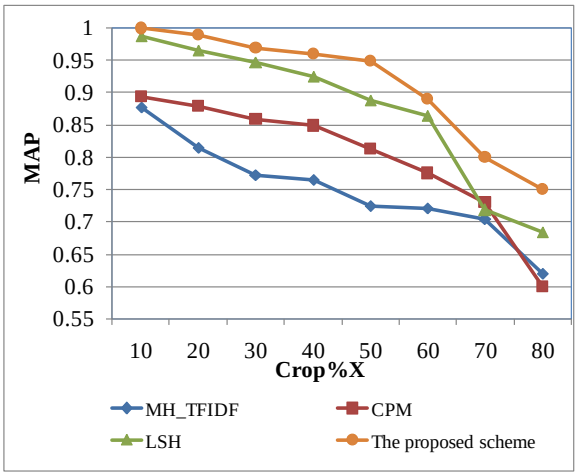

Figure 6. MAP evaluation for different methods with Crop%X changing

Afterwards, other conditional near duplicate image detection methods are compared with the proposed scheme. The methods we choose to make performance evaluation are MH_TFIDF [7], CPM [9] and LSH [10].

It can be known from Fig. 6 and Fig. 7 that the proposed scheme performs better than other methods, because SIFT descriptor can effectively represent local feature of images. On the other hand, in this paper, we represent visual content of image as histograms, and the task of image similarity calculating is converted into estimating the difference between two histograms.

The value of parameter r, k and L are set to 120, 4 and 50 respectively. When threshold δ is set to 0.6, the performance of the proposed scheme is shown in Table.1. For Simplicity, we utilize P, R and $F1$ to represent Precision, Recall and F1 respectively.

As is shown in Table 2, the proposed scheme performs better when δ is equal to 0.6. Particularly, as we use local features to describe image visual contents in this paper, the proposed scheme perform better in the case of the category which contains salient objects. For the given five Flickr groups, the category "Photography with Bokeh" and "Pet Parade" have higher F1 value, because most images in the two groups contain the salient objects.

On the other hand, for the trademark photos in dataset 3, the average F-measure of "Puma" and "BMW" are higher than others, and the reasons lie in that the images related to "Puma" and "BMW" have little diverse visual contents. On the other hand, F1 value of "Sony" is low, as we all know that products of "Sony" trademark have diverse visual contents.

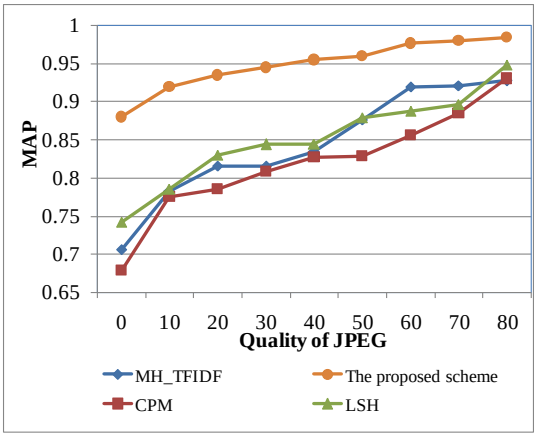

Figure 7. MAP evaluation for different methods with Quality of JPEG changing

Afterwards, it can be seen that that man-made landmarks (such as Big Ben and Leaning Tower of Pisa) have higher F1 than the natural phenomena landmarks (such as Vienna golden hall and Torre di Pisa), the reason lies in that the man-made landmarks are possible containing salient objects.

It also can be seen from Table 2 that with the parameter δ decreasing, the value of precision of each dataset decreasing as well, on the contrary, recall of each dataset increasing. Combining precision and recall, the overall performance F1 reach the max value when δ is set to 0.5 or 0.6.

IV. CONCLUSIONS

We present an approach to detect near duplicate images based on visual word model. Firstly, SIFT features of a given image are clustered into several clusters by the K-means algorithm. Secondly, the centroid of each cluster is represented as a visual word. Thirdly, all the centroids are utilized to construct the visual word vocabulary. Fourthly, locality sensitive hashing is utilized to map high-dimensional visual features into low-dimensional hash bucket space, and the image visual features are converted to a histogram. Finally, near duplicate image detecting process can be implemented by histogram distance computing.

REFERENCES

[1] Liu Ying, Zhang Dengsheng, Lu Guojun, "A survey of content-based image retrieval with high-level semantics," *Pattern Recognition*, 2007, 40(1) pp. 262-282

[2] Feng Deying, Yang Jie, Liu Congxin, "An efficient indexing method for content-based image retrieval," *Neurocomputing*, 2013, 106 pp. 103-114

[3] Datta Ritendra, Joshi Dhiraj, Li Jia, "Image retrieval pp. Ideas, influences, and trends of the new age," *ACM Computing Surveys*, 2008, 40(2), Article No. 5

[4] Li Yangxi, Zhou Chao, Geng Bo, "A comprehensive study on learning to rank for content-based image retrieval," *Signal Processing*, 2013, 93(6) pp. 1426-1434

[5] Song Haiyu, Li Xiongfei, Wang Pengjie, Adaptive feature selection and extraction approaches for image retrieval based on region, *Journal of Multimedia*, 2010, 5(1) pp. 85-92

[6] Yunqi Lei, Xiaoling Gui, Zhenxiang Shi, Feature description and image retrieval based on visual attention model, *Journal of Multimedia*, 2011, 6(1) pp. 56-65

[7] Chum Ondrej, Philbin James, Zisserman Andrew, "Near duplicate image detection pp. min-hash and tf-idf weighting," *Proceedings of the British Machine Vision Conference*, 2008, 3 pp. 4-13

[8] Di Lecce V, Guerriero A, "A comparative evaluation of retrieval methods for duplicate search in image database," *Journal of Visual Languages and Computing*, 2001, 12(1) pp. 105-120

[9] Hu Yiqun, Cheng Xiangang, Chia Liang-Tien, "Coherent Phrase Model for Efficient Image Near-Duplicate Retrieval," *IEEE Transactions on Multimedia*, 2009, 11(8) pp. 1434-1445

[10] Ke Yan, Sukthankar Rahul, Huston Larry, "Efficient near-duplicate detection and sub-image retrieval," *ACM Multimedia*, 2004, 4(1) pp. 4-11

[11] Zheng Ligang, Lei Yanqiang, Qiu Guoping, "Near-Duplicate Image Detection in a Visually Salient Riemannian Space," *IEEE Transactions On Information Forensics And Security*, 2012, 7(5) pp. 1578-1593

[12] Wang Xin-Jing, Zhang Lei, Ma Wei-Ying, "Duplicate-Search-Based Image Annotation Using Web-Scale Data," *Proceedings of the IEEE*, 2012, 100(9) pp. 2705-2721

[13] Sinjur Smiljan, Zazula Damjan, Zalik Borut, "Fast Convex Layers Algorithm for Near-Duplicate Image Detection," *Informatica*, 2012, 23(4) pp. 645-663

[14] Xie Hongtao, Gao Ke, Zhang Yongdong, "Efficient Feature Detection and Effective Post-Verification for Large Scale Near-Duplicate Image Search," *IEEE Transactions on Multimedia*, 2011, 13(6) pp. 1319-1332

[15] Cho Ayoung, Yang Won-Keun, Oh Weon-Geun, "Concentric Circle-Based Image Signature for Near-Duplicate Detection in Large Databases," *ETRI Journal*, 2010, 32(6) pp. 871-880

[16] Zhou Wengang, Li Houqiang, Lu Yijuan, "SIFT Match Verification by Geometric Coding for Large-Scale Partial-Duplicate Web Image Search," *ACM Transactions on Multimedia Computing Communications and Applications*, 2013, 9(1), article No. 4

[17] A. Bolovinou, I. Pratikakis, S. Perantonis, "Bag of spatio-visual words for context inference in scene classification," *Pattern Recognition*, 2013, 46(3) pp. 1039-1053

[18] Jing Zhang, Lei Sui, Li Zhuo, Zhenwei Li, Yuncong Yang, "An approach of bag-of-words based on visual attention model for pornographic images recognition in compressed domain," *Neurocomputing*, 2013, 110 pp. 145-152

[19] Zhou Wengang, Li Houqiang, Lu Yijuan, "SIFT Match Verification by Geometric Coding for Large-Scale Partial-Duplicate Web Image Search," *ACM Transactions on Multimedia Computing Communications and Applications*, 2013, 9(1), Article No.4

[20] Dorado-Munoz Leidy P., Velez-Reyes Miguel, Mukherjee Amit, "A Vector SIFT Detector for Interest Point Detection in Hyperspectral Imagery," *IEEE Transactions on Geoscience and Remote Sensing*, 2012, 50(11) pp. 4521-4533

[21] Slaney Malcolm, Lifshits Yury, He Junfeng, Optimal Parameters for Locality-Sensitive Hashing, *Proceedings of the IEEE*, 2012, 100(9) pp. 2604-2623

[22] Kulis Brian, Grauman Kristen, "Kernelized Locality-Sensitive Hashing," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 34(6) pp. 1092-1104

[23] M. Douze, H. Jégou, H. Sandhawalia, L. Amsaleg, and C. Schmid, "Evaluation of gist descriptors for web-scale

image search," in Proc. *ACM Int. Conf. Image and Video Retrieval*, 2009, pp. 19:1-19:8
[24] M. Gavrielides, E. Sikudova, and I. Pitas, "Color-based descriptors for image fingerprinting," *IEEE Trans. Multimedia*, 2006, 8(4) pp. 740-748
[25] Y. Lei, Y.Wang, and J. Huang, "Robust image hash in radon transform domain for authentication," *Signal Process. pp. Image Commun*, 2011, 26(6) pp. 280-288
[26] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, 2001, 42 pp. 145-175
[27] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in Proc. *IEEE Int. Conf. Computer Vision*, 2003, vol. 2, pp. 1470
[28] H. Jegou, M. Douze, C. Schmid, and P. Perez, "Aggregating local descriptors into a compact image representation," in Proc. *IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, 2010, vol. 0, pp. 3304-3311