Efficient Image Copy Detection Using Multi-Scale Fingerprints

Journal:	*IEEE MultiMedia*
Manuscript ID:	MMSI-2011-04-0051.R1
Manuscript Type:	SI: Jan-March 2012 - Multimedia in Forensics, Security, and Intelligence
Date Submitted by the Author:	26-Jul-2011
Complete List of Authors:	Ling, Hefei; Huazhong University of Science and Technology, School of Computer Science and Technology Cheng, Hongrui; Huazhong University of Science and Technology, School of Computer Science and Technology Ma, Qingzhen; Huazhong University of Science and Technology, School of Computer Science and Technology Zou, Fuhao; Huazhong University of Science and Technology, School of Computer Science and Technology Yan, WeiQi; QUB
Keywords:	H.5.1.f Image/video retrieval < H.5.1 Multimedia Information Systems < H.5 Information Interfaces and Representation (HCI) < H Information Technology and Systems, copy detection, content identification

Efficient Image Copy Detection Using Multi-Scale Fingerprints

Hefei Ling, *Member, IEEE,* Hongrui Cheng, Qingzhen Ma, Fuhao Zou, and Weiqi Yan, *Senior Member, IEEE*

***Abstract*—Inspired by multi-resolution histogram, we propose a multi-scale SIFT descriptor to improve the discriminability. A series of SIFT descriptions with different scale are first acquired by varying the actual size of each spatial bin. Then principle component analysis (PCA) is employed to reduce them to low dimensional vectors, which are further combined into one 128-dimension multi-scale SIFT description. Next, an entropy maximization based binarization is employed to encode the descriptions into binary codes called fingerprints for indexing the local features. Furthermore, an efficient search architecture consisting of lookup tables and inverted image ID list is designed to improve the query speed. Since the fingerprint building is of low-complexity, this method is very efficient and scalable to very large databases. In addition, the multi-scale fingerprints are very discriminative such that the copies can be effectively distinguished from similar objects, which leads to an improved performance in the detection of copies. The experimental evaluation shows that our approach outperforms the state of the art methods.**

***Index Terms*—Copy detection, fingerprints, multi-scale SIFT descriptor, visual words, histogram intersection**

I. Introduction

As means of identifying illegal image copies, the copy detection has been proposed. In contrast to watermarking approaches, the identification of work is not based on previously inserted marks but on content-based extracted signatures. An image copy detector searches for not only the exact copy but also the transformed versions. According to the definition in literatures [1], two copies should be derived from the same original works. Copies are actually a subset of near-duplicates. Therefore, it is not feasible to directly apply existing near-duplicates detection techniques to copy detection, which leads to a considerable number of false alarms. This is because most methods employ local descriptors which are not distinctive enough to determine whether two similar images are copies. This motivated us to design a more distinctive descriptor.

Generally the direct matching of individual descriptors is very time consuming and prohibitive for large databases. Nowadays approaches to this problem rely more and more on the bag-of-words(BoW) model. This technique has shown to yield very promising results in the area of image retrieval. However, there are some limitations [1] of this method. The fatal shortage is that this method is unscalable to huge databases. In addition, it hurts the performances in terms of precision and recall, as well as efficiency due to two reasons. Firstly, noisy words are generated because the approximate algorithms are usually used to expedite the clustering process in the case of large datasets. Secondly, the visual words are not discriminative enough to distinguish copies from similar objects due to the difficulty in choosing a very large number of clusters, which results in a lot of false positives. Too many false positives, on the one hand lead to the decreases of precision and recall, on the other hand impact the query efficiency because computing the similarities between them and query images consume a large amount of time.

In this paper, we develop a multi-scale SIFT descriptor to improve the discriminability of local descriptions. To index the local features, an entropy maximization based binarization approach is employed to encode the descriptions into binary codes called fingerprints. Since the fingerprint building is of low-complexity, this method is very efficient and scalable to very large databases. In addition, the fingerprints are very discriminative such that the copies can be effectively distinguished from similar objects, which leads to an improved performance in the detection of copies.

The main contributions of this work are as follows: Firstly, we have proposed a multi-scale SIFT descriptor which has similar effect to the multi-resolution histogram, and can improve the discriminability of local features. Secondly, the descriptions are directly converted to binary codes named fingerprints to index the local features. Lastly, we design an efficient search architecture to improve the query speed.

Our proposed method seems to be similar to the Locality-Sensitive Hashing (LSH) method because both of them convert vectors to binary codes. In fact, they are different in many aspects. Firstly, the performance of our method doesn't depend on the number of hash tables. Secondly, indexing one descriptor only requires 4 bytes, instead of 100-500 bytes in the LSH method. Finally, the similarity measurements are also very different. Unlike the LSH approach, our method is based on histogram intersection.

II. Related Work

The copy detection methods based on global features [2] are usually more efficient, but they are less robust to some geometric attacks, especially cropping and change of the aspect ratio, because the geometric attacks desynchronize the information for extracting the global features. To resolve this

Manuscript received on April 15, 2011. This work is supported by the NSF of China under Grant No. 60873226 and 60803112, the Fundamental Research Funds for the Central Universities and Wuhan Youth Science and Technology Chenguang Program.

The authors except Weiqi Yan are with the College of Computer Science, Huazhong University of Science and Technology, Wuhan, Hubei, China. Weiqi Yan is with Institute of ECIT, Queen's University Belfast, Belfast, BT7 1NN, United Kingdom. E-mails: lhefei@hust.edu.cn, chr6999@126.com, mqzhen@163.com, fuhao_zou@hust.edu.cn, and w.yan@qub.ac.uk.

problem, some approaches based on local features have been proposed. In [3], Ke et al. proposed a local-region detector for near-duplicate detection and sub-image retrieval based on PCA-SIFT descriptor [4]. To further improve the retrieval efficiency, Foo et al. [5] proposed a pruning strategy that reduces the number of SIFT features [6]. Whereas, the SIFT descriptor [6] has become popular since it achieves the best evaluation performance among all the local descriptors [7]. To improve the efficiency, Jegou et al. [13] proposed a descriptor named VLAD (vector of locally aggregated descriptors), which aggregates SIFT descriptors into a vector of limited dimension.

The SIFT and PCA-SIFT descriptors have been applied to the retrieval of near-duplicates and similar images and achieved good performances, but it will cause the problems of false positives and ambiguities [8] if they are directly applied to copy detection. This is because these descriptions depend on the local gradient statistics of small patches, which consequently makes them too local. For example, even if two images of the same scene taken from different camera locations, namely without copy relationship between them essentially, it is still regarded as existing copy relationship. Another example is demonstrated in Section 6 of [3], many images containing the same landmarks are considered as near-duplicates due to incorrect match using the similar descriptions on the landmarks.

For efficient similarity search, hashing based methods have been proposed. LSH [9] is one of the most popular approximate nearest neighbor search algorithms used in multimedia applications. The main drawback of the basic LSH method [9] is that it may require a large number of hash tables in order to achieve a good search quality. Qin et al. proposed a new indexing scheme called multi-probe LSH [10] which overcomes this drawback. Although the multi-probe LSH method has substantially outperformed previously proposed LSH methods, the space requirement for the hash tables may still exceed the memory size as the number of multimedia objects increases. To address this issue, recently a few data-aware hashing methods have been proposed via resorting to machine learning. The spectral graph partitioning methods are employed to develop new kinds of hashing schemes such as Spectral hashing [11] and Self-taught hashing [12]. It has demonstrated that both schemes have gained a significant improvement over the existing methods. However, Both of them based on local embedding techniques have some intrinsic deficits: high computational cost and out-of-example issue.

The method based on bags-of-visual words introduced by Sivic and Zisserman in the case of video retrieval [14] has become very popular recently in the field of near-duplicates retrieval. Since the bag-of-words model relies on a simple count of the visual word occurrences in the images, any spatial relations between words are lost. Using spatial information helps discriminate visual words and therefore improves the precision performance. Some use spatial relations to group visual words into visual phrases [15]. The visual words based methods have two main problems. one is that the visual vocabulary construction is time consuming and unscalable to very large databases. The other is that the visual words have so weakly discriminative power that they cannot distinguish copies from similar objects, which results in a lot of false positives and consequently leads to low precision performance and efficiency. Therefore, retrieving duplicate images with high performance remains an open issue.

III. MULTI-SCALE FINGERPRINT EXTRACTION

A. Multi-Scale SIFT Descriptor

The local feature extraction is performed in two steps: detecting regions of interest points and computing their descriptions. For interest point detection, we use Lowes Difference of Gaussian [6] (DoG) detector because it has been shown to be robust and efficient. The range of each local region can be automatically selected according to the characteristic scale of the corresponding interest points.

For computing descriptions, the SIFT descriptor [6] has proven to achieve the best evaluation performance among all the local descriptors by Mikolajczyk [7]. However, they are not distinctive enough to tell the copies from the similar images. The SIFT description is a histogram of the image gradients of a local patch. To some extent, image gradients reflect the textures of image. Therefore, the SIFT description can be considered as a histogram of image texture distribution of a local patch. For color statistics, there exist two different images sharing the same histogram. Similarly, two local patches sharing the same histogram of textures may not be copies. This implies that the SIFT description which only computes a single resolution histogram is not distinctive enough. Inspired by multi-resolution histogram which has proven to be more distinctive than single histogram [16], we propose multi-scale SIFT descriptor. Instead of generating a series of multi-resolution image spaces, we select a series of local patches with different radius and compute the histogram of image gradients for each local patch. The multi-scale SIFT descriptor has similar effect to the multi-resolution histogram, and could improve the discriminability of local features.

The SIFT descriptor of a keypoint (x, σ) is a histogram of the image gradients orientations and locations of the Gaussian scale space $G(\cdot, \sigma)$. As shown in Fig. 1, the histogram bins form a three dimensional lattice with $N_p = 4$ bins for each spatial direction and $N_o = 8$ bins for the orientation for a total of $N_p^2 N_o = 128$ components. Each spatial bin is square with unitary edge. The window $H(x)$ is Gaussian with deviation equal to half the extension of the spatial bin range $N_p/2$. The actual size of a spatial bin is $m\sigma$ where σ is the scale of the keypoint and m is a scale factor. The layout is also rotated so that the axis x_1 is aligned to the direction θ of the keypoint. As the scale factor m increases, more pixels participate in the computation of local statistics of the gradient orientations, thus different scale SIFT descriptions are acquired. By experiments, we set $m = 3, 6, 9, 12$, then four different scale SIFT descriptions $SIFT - i$ (where $i = 1, 2, 3, 4$) are obtained. To form a compact representation, each SIFT description is reduced to 32 dimension by using PCA. Thus all the four scale SIFT descriptions after dimension reduction are combined to an 128-dimension vector, which is called the multi-scale SIFT description of a keypoint.

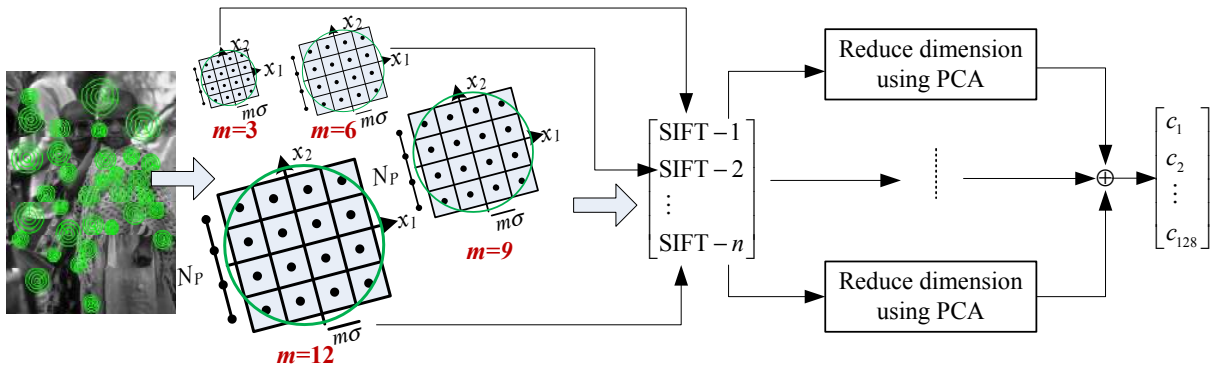

Fig. 1. Multi-scale SIFT descriptor layout. The actual size of a spatial bin is $m\sigma$ where σ is the scale of the keypoint and m is a scale factor. By setting $m = 3, 6, 9, 12$, four different scale SIFT descriptions are acquired.

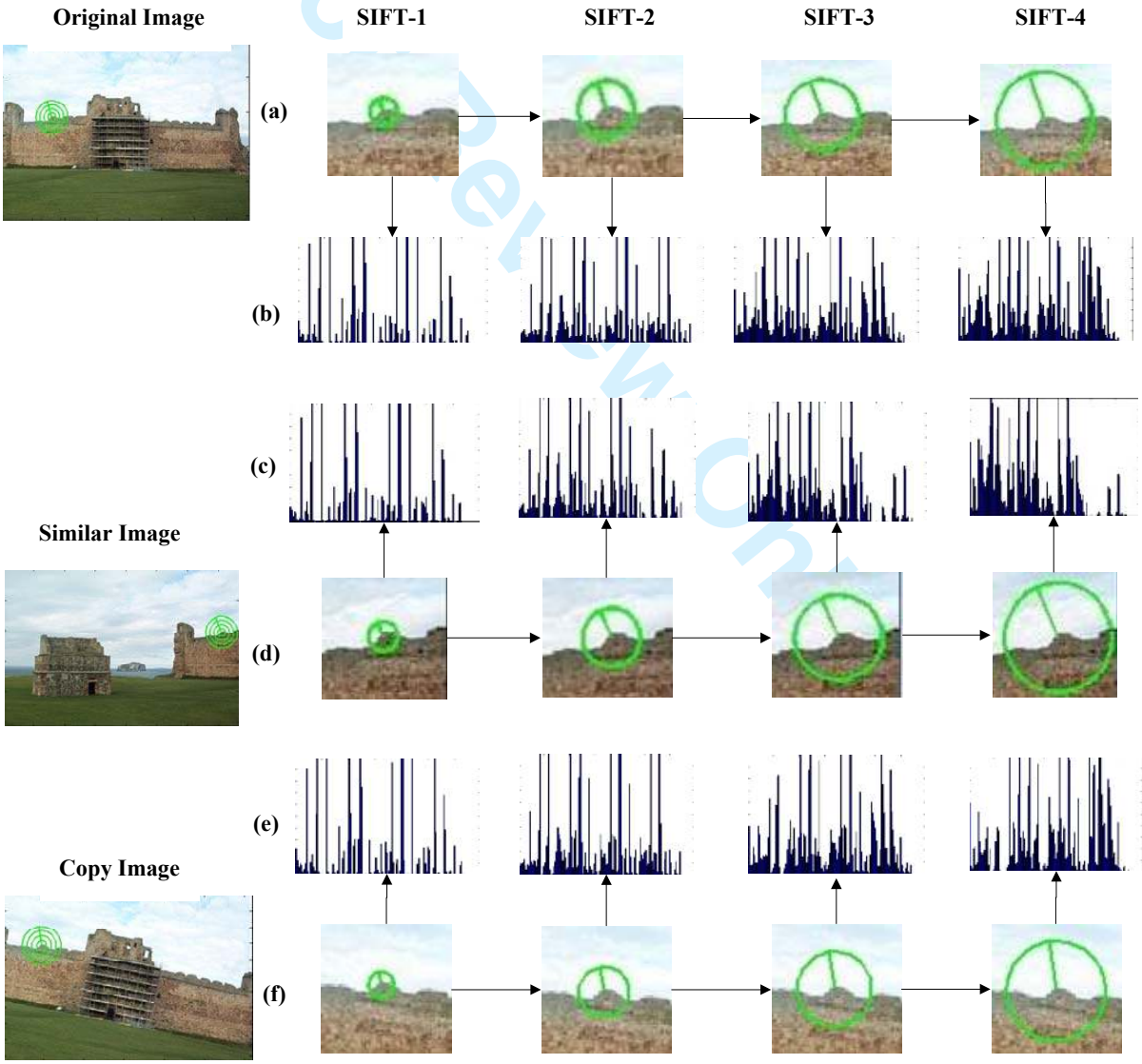

Fig. 2. Examples of three multi-scale histograms. The first, fourth and sixth rows show the multi-scale local patches of the original, similar and copy images, respectively. The second, third and fifth rows show the multi-scale gradient histograms of the three images.

Fig. 2 shows examples of three multi-scale histograms of the original, similar and copy images. The second column of Fig. 2 shows three local patches with identical gradient histogram. The first, fourth and sixth rows show the multi-scale local patches of the original, similar and copy images, respectively. The second, third and fifth rows show the multi-scale gradient histograms of the three images. In each multi-scale gradient histogram, the histograms of corresponding larger scale of the two similar images are different, whereas those of two copy images are similar. This shows that multi-scale SIFT descriptor has more power of discriminability to differentiate similar images, but has little effects on detecting copies.

B. From Vectors to Codes

In order to efficiently search the nearest neighbors of a query vector in a huge dataset, the multi-scale SIFT descriptions need to be converted into binary codes. Given a *D*-dimensional input vector, we want to produce a code of *d* bits encoding the vector representation. This problem can be handled in two steps: 1) a projection that reduces the dimensionality of the vector and 2) a binarization used to encode the resulting vectors.

Dimensionality reduction is an important step in approximate nearest neighbor search, as it impacts the subsequent binarization method. PCA is a standard tool for dimensionality reduction: the eigenvectors associated with the d most energetic eigenvalues of the empirical vector covariance matrix are used to define a matrix $\mathbf{P}$ mapping a vector $x \in \Re^D$ to a vector $y = \mathbf{P}x \in \Re^d$. Matrix $\mathbf{P}$ is the $d \times D$ upper part of an orthogonal matrix.

Given a collection of *n* local features which are represented as *D*-dimensional multi-scale SIFT descriptions $\{x_i\}_{i=1}^n \subset \Re^D$. Let $\mathbf{Y}$ denote $n \times d$ matrix whose *i*-th row is the vector of *i*-th local feature after dimensionality reduction, i.e., $[y_1, y_2, ..., y_n]^T$, the *p*-th column of $\mathbf{Y}$ constitutes a vector v_p, where $p = 1, 2, ..., d$. Suppose that the desired length of code is d bits. We use $b_i \in \{0, 1\}^d$ to represent the binary code for vector y_i, where the *p*-th element of b_i, i.e., b_i^p is 1 if the *p*-th bit of code is on, or 0 otherwise. We now convert the above *d*-dimensional real-valued vectors $y_1, y_2, ..., y_n$ into binary codes via thresholding: if the *p*-th element of y_i is over the specified threshold, $b_i^p = 1$; otherwise, $b_i^p = 0$.

Our binarization method is similar to that employed in [12]. As pointed out by [17], a "good" semantic hashing should be entropy maximizing to ensure efficiency. According to the information theory: the maximal entropy of a source alphabet is attained by having a uniform probability distribution. It means that the entropy of codes over the corpus is large if vectors are mapped to a large number of codes; otherwise, small. To meet the entropy maximizing criterion, we set the threshold for binarising the *p*-th elements of y, i.e., $y_1^p, y_2^p, ..., y_n^p$ to be the median value of v_p. In such way, the *p*-th bit will be '1' for half of the corpus and '0' for the other half. Therefore this thresholding method gives each distinct binary code roughly equal probability of occurring in the local description collection, thus achieves the best effectiveness.

In our method, d is set to 32, such that each local description is converted to a 32-bit code which can be stored as a number.

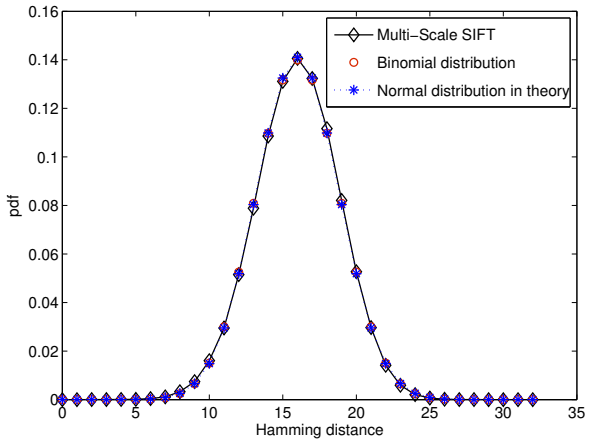

Fig. 3. The probability density function of the Hamming distance distributions of non-copies, including the measured distributions using multi-scale SIFT descriptions, binomial distribution, normal distribution in theory.

The number is regarded as the extracted local fingerprint and stored in the database. The purpose of reducing the dimension of multi-scale SIFT vectors from 128 to 32 is to make preparation for generating compact binary codes, which have proven to be very efficient for retrieval on large-scale data in many hash based methods. Using 32-bits codes to represent multi-scale SIFT descriptions is beneficial to the efficiency, but also results in semantic loss. To lower the semantic loss, we treat two codes whose Hamming distance is below a small number as one word, which will be discussed in detail in Section IV.

C. False Positive Analysis

Two local image patches are declared similar if the Hamming distance (i.e., the number of bit errors) between their fingerprints is below a certain threshold *T*. For the selection of the threshold *T*, the false positive rate P_f and the false negative rate P_n should be considered. The smaller *T*, the smaller the probability P_f will be. On the other hand, a small value of *T* will result in high false negative rate P_n. In practice, P_n is difficult to analyze because there are various image processing operations of which the exact characteristics are unknown. Thus it is usual to analyze the false positive rate P_f for the choice of threshold *T*.

In order to choose a suitable threshold *T*, we assume that the fingerprint extraction process yields random independent and identically distributed (i.i.d.) bits. Then the number of bit errors between the fingerprints from different image patches will have a binomial distribution $B(n, p)$, where *n* is equal to the number of bits extracted and $p(= 0.5)$ is the probability that a $'0'$ or $'1'$ bit is extracted. For binomial distribution $B(n, p)$, the Probability Density Function (PDF) is:

$$pdf_B(k) = \left(\begin{array}{c} n \\ k \end{array} \right) p^k (1-p)^{n-k} = 2^{-n} \left(\begin{array}{c} n \\ k \end{array} \right) \quad (1)$$

Generally the binomial distribution can be approximated by a normal distribution $N(\mu, \sigma)$ with mean $\mu = np$ and standard deviation $\sigma = \sqrt{np(1-p)}$.

TABLE I
VALUES OF FALSE POSITIVE RATE P_f AS T VARIES FROM 0 TO 4.

T	0	1	2	3	4
P_f	7.7E-9	5.7E-8	3.7E-7	2.2E-6	1.1E-5

To determine the distribution of the Hamming distance with real fingerprints, a fingerprint database of about 42,848 features was generated. For each feature, we extracted the multi-scale SIFT description, which is finally converted to a 32-bit fingerprint. Whereafter, the Hamming distances between the fingerprints of 1M randomly selected pairs of features in the database were calculated. Fig. 3 shows the PDF of the Hamming distance distributions acquired by using different models. From this figure, we can observe that the PDF of measured distribution is very close to the binomial distribution and the normal distribution in theory. For the distribution using multi-scale SIFT fingerprints, the false positive rate P_f for the Hamming distance is given as follows:

$$P_f = \frac{1}{2}\mathrm{erfc}\left(\frac{\mu - T}{\sqrt{2}\sigma}\right) \quad (2)$$

where $\mu = np = 16$, $\sigma = \sqrt{np(1-p)} = 2.83$. For a certain value of P_f, the threshold T for the Hamming distance can be determined. As T varies from 0 to 4, the values of false positive rate P_f are listed in Table I. From this table, the threshold could be obtained at a desired false positive rate.

IV. DATABASE QUERY

A. Image Representation and Similarity Measures

In the database, each image is represented as a set consisting of a variable number of fingerprints, each of which is independent and orderless. Assume a vocabulary V of size $|V|$ where each visual word is encoded with its fingerprint. Let an image I be a set of words $F_i \subset V$, and $H(I)$ is the histogram of I which counts the frequency of each word in I. The distance measure between two images is computed as the similarity of sets I_1 and I_2, which is defined as the ratio of the number of elements in the intersection over the union:

$$sim_s(I_1, I_2) = \frac{|I_1 \cap I_2|}{|I_1 \cup I_2|} \quad (3)$$

The advantage of such a choice of the similarity measure is that it enables very efficient retrieval. The drawback is that some relevant information, such as frequencies of each word, is not preserved in the set of visual words representation. Therefore, a histogram intersection is employed to measure the similarity between two images.

Let $H_w(I)$ be the number of visual word F_w presented in the image I. The histogram intersection measure is defined as

$$sim_h(I_1, I_2) = \frac{\sum_w \min(H_w(I_1), H_w(I_2))}{\sum_w \max(H_w(I_1), H_w(I_2))} \quad (4)$$

This similarity measure (4) resembles the $tf - idf$ weighting scheme, while preserving the advantages of very fast retrieval of similar images. Therefore we use this similarity measure in our experiments.

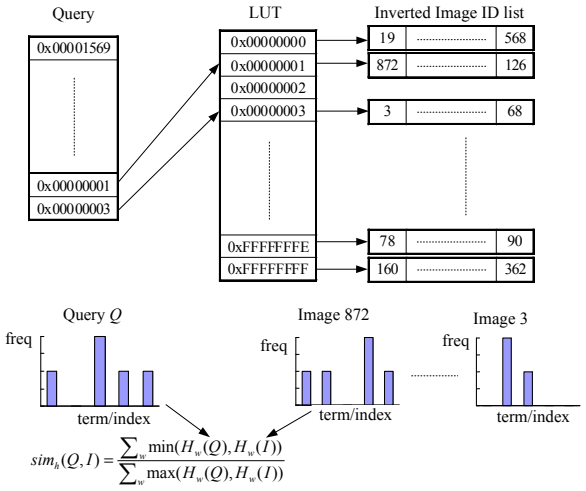

Fig. 4. The efficient search architecture, where an image represented by a set of fingerprints serves as the query.

B. Efficient Search Architecture

To speed up the copy retrieval, a lookup table (LUT) [18] and inverted image ID list are used in the efficient search architecture shown in Fig. 4. The architecture consists of offline preprocess for image dataset and online retrieval for query images. During offline preprocess, all the fingerprints of images in the dataset are extracted and stored in a lookup table LUT with all possible 32 bit fingerprints as an entry. Each entry points to an image ID list storing all the IDs of images which contain this fingerprint, and the word frequency in each image. During the online retrieval process, for a given query image, we first extract the fingerprints from the query image. Then, using the extracted fingerprints as queries, all candidate images which at least contain one of the extracted fingerprints are fast obtained by using the lookup table LUT. Next, using the candidate image IDs as queries, all the word frequencies corresponding to the candidate images are acquired. Finally, we compute the similarities of candidate images by using the histogram intersection measure (Eq. 4). The images whose similarity score is over the predefined threshold are the returned query results.

The lookup table LUT contains all possible 32 bit fingerprints as entries. In theory, the maximal size of LUT is 2^{32} (4G). In practice, the spaces are sparsely filled if the dataset is not very large. During the construction of LUT, only those fingerprints existed in the database are added to the table. Therefore the actual size of LUT depends on the scale of dataset. With the scale of dataset increasing, more fingerprints are generated, which leads to the size increase of LUT. For example, a huge dataset of 10M images may generate 1G fingerprints, only about 300M unique fingerprints are in LUT since some repeated fingerprints exist. Since the actual size of lookup table is small, a hash table is usually used instead of a lookup table. Using a modern PC, a rate of approximately of 200K fingerprint retrievals per second can be achieved. In our experiments, one query image has 160 fingerprints on average. Therefore to locate the entries only needs 0.8 milliseconds. The lookup-table can be implemented in such a way that it

has no impact on the search time.

C. Search Strategy

If we only search for the exact fingerprint, the aforementioned method is sufficient. However, for a heavily degraded image the extracted fingerprints may have some error bits. Therefore, to improve the performance, we attempt to search for the most similar fingerprints. Given one query fingerprint F_0, $N(=\sum_{k=1}^{T} C(32,k))$ similar fingerprints F_i whose Hamming distances with the query fingerprint are below T are treated as the same word with query. Therefore, one new word F_w defined equaling to F_0 corresponds to $N+1$ representations. Thus the number of the visual word F_w in the images I should be updated as $H_w = \sum_{i=0}^{N} H_i$. Then we can compute the similarities between the query image and result images by using Eq. 4.

A question arisen is how to determine the threshold T for Hamming distance between two similar fingerprints. As the threshold T increases, more similar fingerprints are treated as one word, which consequently leads to a much lower false negative rate, but a sharply increasing false positive rate. Too many false positives result in not only low precision, but also low efficiency because a large amount of time needs to be spent on computing the similarities between false positives and query images. Therefore, the choice of T should make a tradeoff among false positive rate, false negative rate and query efficiency.

V. EXPERIMENTS

In this section, we first evaluate the performance of our method as the threshold T for hamming distance varies from 0 to 4. Then, a comparison with the state of the art methods is presented.

A. Datasets and Evaluation Measures

The evaluation is performed on a collection of one million images downloaded from websites. The collection has a lot of images depicting similar content.

To generate copies, each image is modified using Stirmark, a standard benchmark which has been employed to simulate various manipulations of the digital images in some literatures related to copy detection. Howbeit Stirmark, a tool originally designed for watermarking benchmark, emphasizes too much on geometric attacks. To complement this, we conducted five new classes of attacks, which are listed as follows: (1) Colorizing; (2) Changing saturation; (3) Changing intensity; (4) Flipping; (5) Seam carving. Totally, there are 100 alterations which are listed as follows:

(a) SPA (15): Signal Processing Attacks, including colorizing (3), changing saturation (3), changing intensity (3), median filtering (3), Gaussian filtering, sharpening, and frequency mode Laplacian removal (FMLR).
(b) JPEG (12): JPEG compression with quality factors ranging from 90% to 10%.
(c) GLGT (3): General Linear Geometric Transform.
(d) CAR (7): Change of the Aspect Ratio.

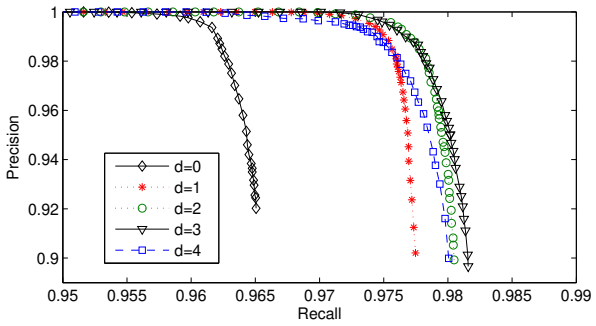

Fig. 5. P-R curves of the proposed method in the case of d=0,1,2,3,4.

(e) LR (5): Line Removal.
(f) RC (16): Rotation + Cropping.
(g) Scaling (5): Scaling with factors ranging from 0.5 to 2.0.
(h) Cropping (9): Cropping the image by a percentage ranging from 1% to 75%.
(i) Shearing (6): Apply affine warp on both x and y axes.
(j) RRS (16): Rotation + Re-Scaling.
(k) RB (1): Random Bending.
(l) Flipping (1).
(m) Seam carving (4).

The number in each bracket refers to the number of copies produced in each class of manipulation.

For the test collection, we randomly select 5k images as the query images, and all the query images are processed to produce 500k copies. then 500k images randomly selected from the rest images serves as non-copies. Consequently we created two image collection, C1 and C2, with aggregated sizes of 500k non-copies and 500k copies, respectively.

Each query image is represented as a set of fingerprints. Each returned image has a score of similarity. The Precision-Recall (P-R) curve is acquired by varying the detection threshold. The images with score over the threshold are considered as a positive class. Let R_P be the number of true copies correctly assigned to the positive class, F_P the number of false copies incorrectly assigned to the positive class, and R_N the number of true copies incorrectly rejected by the positive class. The precision and recall are defined as:

$$precision = \frac{R_P}{R_P + F_P}, recall = \frac{R_P}{R_P + R_N} \quad (5)$$

B. Performance of the Proposed Method

We first evaluate the performance of our method as the threshold for hamming distance varies from 0 to 4. All similar fingerprints to the query are considered as the representations of the same feature, and serve as queries on the test collection. For the Hamming distance d ranging from 0 to 4, the P-R curves are drawn in Fig. 5. From this figure, we can observe that the performance in terms of precision and recall has a remarkable rise as d increases from 0 to 1. As d continues to increase up to 3, there are only slight improvement on performances. However, the performance degrades as d increases up to 4. The method when d=2 can obtain the best performance.

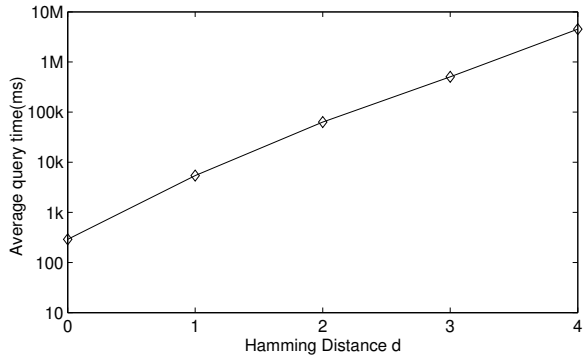

Fig. 6. The average search time per query image as the number of error bits d varies from 0 to 4.

However, the retrieval efficiency for d=2 is much lower than that for d=0. The average search time per query image varying with d is computed and presented in Fig. 6. It can be observed that the time consumed when d=2 is about 63.3s, which is more than 140 times longer than 0.45s which is consumed when d=0. The consumed time increases dramatically as d increases, which is because more candidates are input as queries. Therefore, the choice of d should make a tradeoff between precision and search efficiency. If the dataset is small, d can be set to 1 or 2; otherwise, 0 or 1.

C. Comparison with the State of the Art

For comparison purposes, we have implemented an very efficient method named VLAD [13]. The number of cluster centroids is set to 64 (k=64), then a VLAD vector of 8192 (D=8192) dimensions is acquired for each image. Second, Each VLAD is reduced to 128 dimensional by principle component analysis (PCA). Next, the 128-dimension vector is divided into 16 segments, which is quantized by 256 centroids to obtain 16 8-bit integers using ADC(Asymmetric Distance Computation). At the last step,we implemented IVFADC which combines ADC with an inverted file to restrict the search to a subset of vectors. the parameters of IVFADC is k'=1000 and w=1. For comparison, we have also implemented the method based on Clustering-Defined Visual Words and BoW model (CDVW), in which clustering is performed with K-means algorithm on a dataset of 200k samples. In order to acquire a good discriminability, we set the number of clusters at 100k by experiment. Clustering the local features is really time-consuming, which takes more than ten days.

The precision-recall (PR) curves shown in Fig. 7. We can observe that the proposed method yields very excellent performances in terms of both precision and recall, which are much higher than those achieved by using all the other methods. For VLAD method, as an increasingly number of steps are executed, it leads to more errors, but an improvement on efficiency.

Fig. 8 shows that the average query time of the proposed method, SIFT fingerprints based method, CDVW, and VLAD methods. As the database size of images varies from 1.0E+3 to 1.0E+6, we calculated the average time consumed per query image. From this figure, it is obvious that our method is as

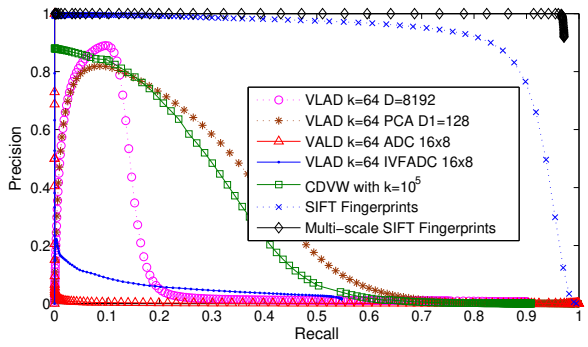

Fig. 7. P-R curves of the proposed method, VLAD and CDVW methods.

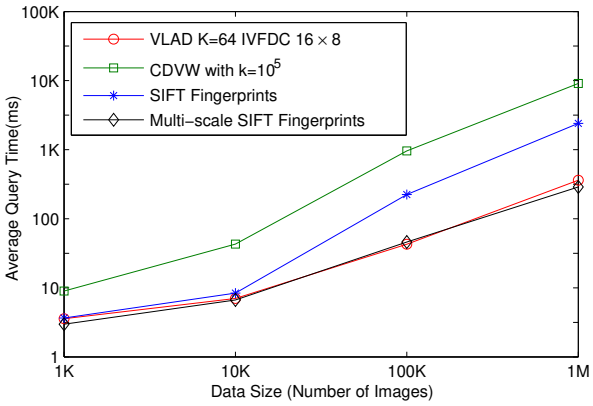

Fig. 8. The average query time per query image as database size (number of images) varies from 1.0E+3 to 1E+6.

efficient as the VLAD method. Both of them are much more time efficient than CDVW and the SIFT fingerprints based method.

VI. CONCLUSIONS

We have demonstrated an efficient image copy detection method using multi-scale SIFT fingerprints. The main idea is to index the local features by using the extracted fingerprints. Inspired by multi-resolution histogram, we propose multi-scale SIFT descriptor to improve the discriminability. Then PCA and entropy maximization based binarization are employed to reduce the dimensionality and encode the vectors into binary fingerprints, respectively. Moreover, an efficient search architecture is designed to improve the retrieval speed. Since the fingerprint building is of low-complexity, this method is very efficient and scalable to very large databases. In addition, the fingerprints are very discriminative such that the copies can be effectively distinguished from similar objects, which leads to an improved performance in the detection of copies.

The experimental results show that our approach outperforms the state of the art methods, including the CDVW, SIFT fingerprints based methods, in terms of precision, recall and efficiency. Compared with the VLAD method, our method can achieve similar efficiency, but much higher precision and recall.

In summary, our method offers a new promising approach for encoding visual words and indexing features, and presents

a viable solution to the challenges of retrieving duplicate images with high performance and efficiency. In the future, we intend to explore the construction of more robust fingerprints, extend the scalability of our approach to an even larger database, and perform a comparative evaluation against other predominant near-duplicate retrieval techniques.

REFERENCES

[1] A. Joly, O. Buisson, and C. Frelicot, "Content-Based Copy Retrieval Using Distortion-Based Probabilistic Similarity Search," IEEE Transactions on Multimedia, vol. 9, pp. 293-306, 2007.

[2] C. Kim, "Content-based image copy detection," Signal Processing: Image Communication, vol. 18, pp. 169-184, 2003.

[3] Y. Ke, R. Sukthankar, and L. Huston, "Efficient near-duplicate detection and sub-image retrieval," in proceedings of ACM International Conference on Multimedia (MM'04), New York, United States, 2004, pp. 869-876.

[4] Y. Ke, R. Sukthankar, and L. Huston, "PCA-SIFT: a more distinctive representation for local image descriptors," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition(CVPR'04), Los Alamitos, USA, 2004, pp. 506-13.

[5] J. J. Foo and R. Sinha, "Pruning SIFT for Scalable Near-Duplicate Image Matching," in Proceedings of the 18th Australasian Database Conference (ADC 2007), Ballarat, Australia, 2007, pp. 63-71.

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision, vol. 60, pp. 91-110, 2004.

[7] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, pp. 1615-1630, 2005.

[8] E. N. Mortensen, H. Deng , and L. Shapiro, "A SIFT descriptor with global context," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR'05), 2005, pp. 184-190.

[9] M. Datar, P. Indyk, N. Immorlica, et al., "Locality-sensitive hashing scheme based on p-stable distributions," in Proceedings of the Annual Symposium on Computational Geometry, Brooklyn, NY, United states, 2004, pp. 253-262.

[10] L. Qin, J. William, W. Zhe, et al., "Multi-probe LSH: efficient indexing for high-dimensional similarity search," in Proceedings of international conference on Very large data bases. Vienna, Austria: VLDB Endowment, 2007.

[11] Y. Weiss, A. B. Torralba, and R. Fergus, "Spectral hashing," in Advances in Neural Information Processing Systems (NIPS), Vancouver, Canada, 2008, pp. 1753-1760.

[12] D. Zhang, J. Wang, D. Cai, et al., "Self-taught hashing for fast similarity search," in International ACM SIGIR Conference on Research and Development in Information Retrieval, Geneva, Switzerland, 2010, pp. 18-25.

[13] H. Jgou, M. Douze, C. Schmid, et al., "Aggregating local descriptors into a compact image representation," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR '10), 2010, pp. 3304-3311.

[14] J. Sivic and A. Zisserman, "Video Google: a text retrieval approach to object matching in videos," in IEEE International Conference on Computer Vision, Nice, France, 2003, pp. 1470-1477.

[15] Q. Zheng, W. Wang, and W. Gao, "Effective and efficient object-based image retrieval using visual phrases," in Proceedings of the 14th annual ACM international conference on Multimedia, Santa Barbara, CA, USA, 2006, pp. 77 - 80.

[16] E. Hadjidemetriou, M. D. Grossberg, and S. K. Nayar, "Multiresolution Histograms and Their Use for Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence,Vol. 26, pp. 831-847, 2004.

[17] S. Baluja and M. Covell, "Learning to hash: Forgiving hash functions and applications," Data Mining and Knowledge Discovery (DMKD),Vol. 17, pp. 402C430, 2008.

[18] J. Haitsma and T. Kalker, "A Highly Robust Audio Fingerprinting System," in Symposium on Music. Information Retrieval (ISMIR), 2002, pp. 107-115.