SIFT Match Verification by Geometric Coding for Large-Scale Partial-Duplicate Web Image Search

WENGANG ZHOU and HOUQIANG LI, University of Science and Technology of China
YIJUAN LU, Texas State University
QI TIAN, University of Texas at San Antonio

Most large-scale image retrieval systems are based on the bag-of-visual-words model. However, the traditional bag-of-visual-words model does not capture the geometric context among local features in images well, which plays an important role in image retrieval. In order to fully explore geometric context of all visual words in images, efficient global geometric verification methods have been attracting lots of attention. Unfortunately, current existing methods on global geometric verification are either computationally expensive to ensure real-time response, or cannot handle rotation well. To solve the preceding problems, in this article, we propose a novel geometric coding algorithm, to encode the spatial context among local features for large-scale partial-duplicate Web image retrieval. Our geometric coding consists 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. Our approach is not only computationally efficient, but also effective in detecting partial-duplicate images with rotation, scale changes, partial-occlusion, and background clutter.

Experiments in partial-duplicate Web image search, using two datasets with one million Web images as distractors, reveal that our approach outperforms the baseline bag-of-visual-words approach even following a RANSAC verification in mean average precision. Besides, our approach achieves comparable performance to other state-of-the-art global geometric verification methods, for example, spatial coding scheme, but is more computationally efficient.

Categories and Subject Descriptors: I.2.10 [**Vision and Scene Understanding**] VISION

General Terms: Algorithms, Experimentation, Verification

Additional Key Words and Phrases: Image retrieval, partial duplicate, large scale, rotation-invariant, geometric square coding, geometric fan coding

ACM Reference Format:

Zhou, W., Li, H., Lu, Y., and Tian, Q. 2013. SIFT match verification by geometric coding for large-scale partial-duplicate Web image search. ACM Trans. Multimedia Comput. Commun. Appl. 9, 1, Article 4 (February 2013), 18 pages.
DOI = 10.1145/2422956.2422960 http://doi.acm.org/10.1145/2422956.2422960

This work is supported in part by the Fundamental Research Funds for the Central Universities of China (WK2100230003) to H. Li, in part by Research Enhancement Program (REP), start-up funding from the Texas State University and DoD HBCU/MI grant W911NF-12-1-0057 to Y. Lu, and in part by NSF IIS-1052851, Faculty Research Awards by Google FXPAL, NEC Laboratories of America, and ARO grant W911BF-12-1-0057 to Q. Tian.

Authors' addresses: W. Zhou, H. Li, Department of EEIS, University of Science and Technology of China, Hefei 230027, P. R. China; Y. Lu, Department of Computer Science, Texas State University at San Marcos, TX 78666; Q. Tian (corresponding author), Department of Computer Science, University of Texas at San Antonio, TX 78249; email: qitian@cs.utsa.edu.

DOI 10.1145/2422956.2422960 http://doi.acm.org/10.1145/2422956.2422960

1. INTRODUCTION

As more and more people become users of Tineye [2008] and Google Similar Image Search [2009], partial-duplicate image search has been attracting more and more attention in recent years. Partial-duplicate images are those images, part of which are usually cropped from the same original image, that are edited with modification in color, scale, rotation, partial occlusion, etc. Figure 1 illustrates some instances of partial-duplicate images from the Web. From these examples, we can find that they are partial duplicates of the original image with different appearances but still sharing some duplicated patches.

The task of partial-duplicate Web image search is to find all the partial-duplicate versions of a given query from a large Web image database. Partial-duplicate image search can be widely used in many applications, such as image/video copyright violation detection, finding out where an image comes from or how it is being used, duplicate image annotation, etc. Besides, it can also facilitate many multimedia applications, such as semantic concept inference [Tang 2009] and image annotation [Tang 2010].

Most of the state-of-the-art approaches in large-scale image retrieval rely on the bag-of-visual-words model [Sivic and Zisserman 2003]. It quantizes local features [Lowe 2004] extracted from images to visual words and indexes images via inverted file structure. Although the bag-of-visual-words model makes it possible to represent, index, and retrieve images like documents, it suffers from visual word ambiguity and feature quantization error. Those unavoidable problems greatly decrease retrieval precision and recall, since different features may be quantized to the same visual word, causing many false local matches between images.

To tackle these problems, many geometric verification [Chum et al. 2009; Jegou et al. 2008; Philbin et al. 2007; Sivic and Zisserman 2003; Wu et al. 2009; Zhou et al. 2010] approaches have been proposed in recent few years to address the negative influence of these false matches to improve retrieval performance. Many of them are local geometric verification approaches, such as spatially nearest neighbors [Sivic and Zisserman 2003], geometric min-hashing [Chum et al. 2009], and bundled feature [Wu et al. 2009]. Since these approaches only can verify spatial consistency of features within some local areas in images, they will fail if there is geometry inconsistency among local areas. Therefore, global geometric verification methods are demanded.

RANSAC [Fischler and Bolles 1981] is the most popular method for global geometric verification. It can fully estimate the transformation model between images and then detect spatially inconsistent pairs. However, due to the high computational cost, usually it is only applied to some top-ranked image results, which is not sufficient for good recall in large-scale image retrieval. To address such a problem, spatial coding [Zhou et al. 2010] is proposed to efficiently check spatial consistency globally. It uses spatial maps to record the spatial relationship of all matched feature pairs. But spatial coding is very sensitive to rotation due to the intrinsic limitation of the spatial map. Although it can weakly handle rotated images by trying a set of predefined angles on the query image, much more time cost will be introduced to retrieve freely rotated duplicate images.

In this article, we propose a novel geometric coding scheme for global spatial verification of SIFT matches, which is both efficient and effective for partial-duplicate image search. We select the SIFT feature [Lowe 2004] for image representation and make full use of SIFT properties. Generally, a SIFT feature is characterized with several property values: a 128D descriptor, a 1D dominant orientation (ranging for $-\pi$ to π), a 1D characteristic scale, and the (x, y) coordinates of the key point. In our approach, a SIFT descriptor is used based on the bag-of-visual-words model, while the orientation, scale, and key point position are all exploited to build our geometric coding maps. In image search, local matches are first discovered through feature quantization. To verify the SIFT matches of two images, we use Geometric Square Coding (GSC) and Geometric Fan Coding (GFC) to encode the relative

Fig. 1. Examples of partial-duplicate Web images. Top: KFC logo; bottom: Beijing 2008 Olympic logo. The partially duplicated patches in the top row are highlighted with red bounding box.

spatial positions of local features in images. Then through spatial verification based on three geometric maps, the false matches of SIFT features can be removed effectively and efficiently, resulting in better accuracy.

The rest of the article is organized as follows. Section 2 reviews the related work. Section 3 discusses our approach in details. Experimental results are provided in Section 4. Finally, we draw the conclusion in Section 5.

2. RELATED WORK

In recent years, large-scale image retrieval [Chum et al. 2007a, 2007b; Jegou et al. 2008; Nister and Stewenius 2006; Philbin et al. 2007; Wu et al. 2009; Zhang et al. 2009, 2010, 2011] with local features has been significantly advanced based on the bag-of-visual-words model [Sivic and Zisserman 2003]. The major contribution of the bag-of-visual-words model is that it can achieve scalability for large-scale image retrieval by quantizing local features to visual words. Popular local features include SIFT [Lowe 2004], SURF [Bay et al. 2006], MSER [Matas et al. 2002], and so on. Local feature quantization not only makes image representation compact, but also makes it possible to index images with inverted file structure, which greatly reduces the number of candidate images for comparison.

However, local feature quantization reduces the discriminative power of local descriptors. Different descriptors may be quantized to the same visual word and cannot be distinguished from each other. On the other hand, with visual word ambiguity, descriptors from local patches of different semantics may also be very similar to each other. Such quantization error and visual word ambiguity will cause many false matches of local features between images and therefore decrease retrieval precision and recall.

To reduce the quantization error, two kinds of approaches have been proposed recently. The first one is to improve the discriminative power of local features. Soft quantization [Philbin et al. 2008; Jegou et al. 2007] and Hamming embedding [Jegou et al. 2008] are two representative works. Soft quantization quantizes a SIFT descriptor to multiple visual words. Hamming embedding enriches the visual word with compact information from its original local descriptor with Hamming codes.

The second category of approaches focus on utilizing geometric information in images to improve retrieval precision. These approaches can be summarized into preprocessing or postprocessing approaches. Inspired by shape context [Belongie et al. 2002; Oliva and Torralba 2001], the motivation of preprocessing approaches is to encode spatial context of local features into image representation. In Hoang et al. [2010], a new image content representation scheme is proposed to describe the spatial layout with triangular relationships of visual entities for scene retrieval. In Gao et al. [2010], a spatial-bag-of-features scheme is used to encode geometric information of objects within an image. It

projects local features of an image to different directions to generate an ordered bag-of-features for image search. However, with the large amount of local features in images, it is hard for the preprocessing approaches to fully encode various spatial relationships. It makes image representation very complex and increases image matching time. In Zhang et al. [2011], a Geometry-preserving Visual Phrase (GVP) is proposed to encode the spatial information of local features, including both co-occurrences and the local and long-range spatial layouts of visual words. With little increase in memory usage and computational time, retrieval accuracy improvement is witnessed. However, GVP only captures the translation invariance. Although its extension to scale and rotation invariance can be achieved by increasing dimension of the offset space, more memory usage and runtime will be incurred. In Wang et al. [2011], the statistics in the local neighborhood of an invariant feature is used as its spatial context to enhance the discriminative power of the visual word.

The postprocessing approaches try to avoid these problems. They do not change the image representation and image matching scheme. Instead, after obtaining the local matches between images, they use geometric consistency to filter those false matches. Since the number of matched features is much smaller than the number of features in the image, postprocessing approaches can be very efficient.

The locally spatial consistency of some spatially nearest neighbors is used in Sivic and Zisserman [2003] to suppress false visual-word matches. However, the nearest-neighbors' spatial consistency only imposes very loose geometric constraints and may be sensitive to the image noise from background clutter. In Jegou et al. [2008], Weak Geometric Consistency (WGC) is used to filter false local matches. A constraint is imposed that correct local matches should exhibit similar characteristic scale and rotation changes. Therefore, histograms of characteristic scale and dominant orientation differences have obvious peaks, which are used to identify false local matches with orientation or scale differences far from those peaks. In Zhao et al. [2010], WGC is enhanced by including translation information. An additional assumption is made that the correct matches follow consistent translation transformation. Bundled feature [Wu et al. 2009] assembles features in local MSER [Matas et al. 2002] regions to increase the discriminative power of local features. The local geometric consistency is measured by projecting feature positions along horizontal and vertical directions in local MSER regions. However, when an image suffers from rotation changes, such projection will yield a different local geometric representation and cause incorrect geometric measurement. Geometric min-hashing [Chum et al. 2009] constructs repeatable hash keys with loosely local geometric information for more discriminative-break description.

All of the preceding postprocessing approaches only verify spatial consistency of features within local areas instead of the entire image plane. Although computationally efficient, they cannot capture the spatial relationship between all features, which makes it hard to detect all false matches and hence obtains limited precision improvement.

To capture geometric relationships of all features in the entire image, a global geometric verification method such as RANSAC [Chum et al. 2004; Fischler and Bolles 1981; Philbin et al. 2007] is often used for this task. It randomly samples a subset of matching feature pairs many times to estimate an optimal transformation model. RANSAC can greatly improve retrieval precision. However, it is computationally expensive. In practice, it is usually applied on the subset of the top-ranked candidate images, which may not be sufficient to achieve good recall in large-scale image retrieval systems. In the content-based image search system VisualSEEk [Smith and Chang 1996], 2D strings [Chang et al. 1987] are adopted to represent images with multiple color regions for comparison. 2D strings represent an image as a symbolic projection along the x and y directions. Then, the image retrieval problem is converted to a problem of 2D sequence matching.

The spatial coding approach [Zhou et al. 2010] is another global geometric verification method proposed to remove false matches based on spatial maps. The problem of spatial coding is that it requires

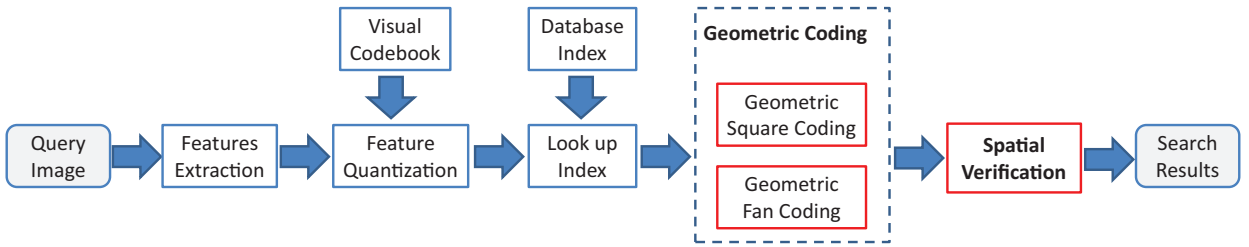

Fig. 2. Our image search framework.

that the duplicated patches in the query and the matched image share the same or very similar spatial configuration and cannot handle rotation very efficiently. Although weak rotation invariance can be achieved by rotating the query image with some predefined angles [Zhou et al. 2010], in practice, it will be time consuming to enumerate all possible rotation angles to search freely rotated target images.

In this article, our motivation is to design an efficient global geometric verification scheme, which can achieve both rotation and scale invariance, and is not sensitive to background clutter. We propose two coding schemes, that is, geometric square coding and geometric fan coding, to strictly describe the geometric context of local features for global spatial verification. Our approach can efficiently and effectively address images with arbitrary rotation changes.

3. OUR APPROACH

Based on the bag-of-visual-words model, the framework of our large-scale partial-duplicate image search system is illustrated in Figure 2. Our main contribution lies in geometric coding and spatial verification, as highlighted with red bounding box. In our approach, we adopt SIFT feature [Lowe 2004] for image representation. In Section 3.1, we apply the SIFT descriptor for vector quantization. In Section 3.2, the key point location, orientation, and scale of the SIFT feature are exploited for geometric coding maps generation. In Section 3.3, we explain how to perform spatial verification with those geometric coding maps. More details of the framework are discussed in Section 4.1.

3.1 Feature Quantization

We index images on a large scale with an inverted index file structure for retrieval. Before indexing, SIFT features are quantized to visual words based on the bag-of-visual-words model [Sivic and Zisserman 2003]. A quantizer is defined to map a SIFT descriptor to a visual word. The quantizer can be generated by clustering a sample set of SIFT descriptors and the resulting cluster centroids are regarded as visual words. During the quantization stage, a novel feature will be assigned to the index of the closest visual words. In our implementation, we use the hierarchical visual vocabulary tree approach [Nister and Stewenius 2006] for visual vocabulary generation and feature quantization. With feature quantization, any two features from two different images quantized to the same visual word will be considered as a local match across two images.

3.2 Geometric Coding

The spatial context among local features of an image is critical in identifying duplicate image patches. After SIFT quantization, SIFT matches between two images can be obtained. However, due to quantization error and visual word ambiguity, the matching results are usually polluted by some false matches. Generally, geometric verification can be adopted to refine the matching results by discovering the transformation and filtering false positives [Philbin et al. 2007]. Since full geometric

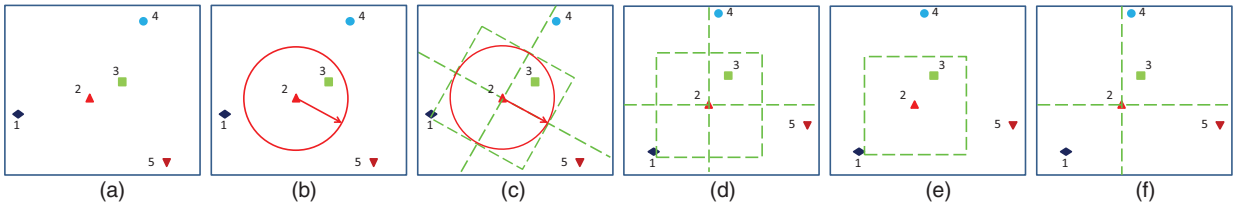

Fig. 3. Illustration of image plane division with the key point of feature 2 as reference point. (a) Five SIFT features in image; (b) key point of feature 2 displayed as vector indicating scale, orientation, and location (red arrow); (c) image plane division with lines and square (green dashed lines) with the key point of feature 2 as reference point; (d) image plane rotation from (c); (e) and (f): image subdivisions from (d).

verification with RANSAC [Chum et al. 2004; Fischler and Bolles 1981] is computationally expensive, it is only used as a postprocessing stage to process initially top-ranked candidate images. A more efficient scheme to encode the spatial relationships of visual words is desired. With such motivation, we propose the geometric coding scheme.

The key idea of geometric coding is to encode the geometric context of local SIFT features for spatial consistency verification. Our geometric coding is composed of two types of coding strategies, that is, geometric square coding and geometric fan coding. The difference between the two strategies lies in the way the image plane is divided according to an invariant reference feature. Before encoding, the image plane has to be divided with a certain criterion that can address both rotation invariance and scale invariance. We design the criterion via the intrinsic invariance merit of the SIFT feature.

Figure 3 gives a toy example of image plane division with the key point of feature 2 as reference point. Figure 3(b) illustrates an arrow originated from the key point of feature 2, which corresponds to a vector indicating the characteristic scale and dominant orientation of the SIFT feature. Using the key point of feature 2 as origin and direction of the arrow as the major direction, two lines horizontal and vertical to the arrow of feature 2 can be drawn. Besides, centered at the same key point, a square is also drawn along these two lines, as shown in Figure 3(c). The side length of the square is proportional to the characteristic scale of feature 2. For comparison convenience, we rotate all features to align the red arrow to be horizontal, as shown in Figure 3(d). After that, the image plane division with two coordinate axial lines and a square can be decomposed into two kinds of subdivisions, as shown in Figure 3(e) and (f), which will be used for geometric square coding and geometric fan coding, respectively. The details are discussed in the following two subsections.

3.2.1 *Geometric Square Coding.* Geometric Square Coding (GSC) encodes the geometric context in the axial direction of reference features. In GSC, with each SIFT feature as reference origin, the image plane is divided by squares. A square coding map, called S-map, is constructed by checking whether other features are inside or outside of the square.

To achieve rotation-invariant representation, before checking relative position, we adjust the location of each SIFT feature according to the SIFT orientation of the reference feature. For instance, given an image I with M features $\{f_i(x_i, y_i)\}, (i = 1, 2, \ldots, M)$, with feature $f_i(x_i, y_i)$ as reference point, the adjusted position $f_j^{(i)}(x_j^{(i)}, y_j^{(i)})$ of $f_j(x_j, y_j)$ is formulated as

$$\begin{pmatrix} x_j^{(i)} \\ y_j^{(i)} \end{pmatrix} = \begin{pmatrix} \cos(\phi_i) & -\sin(\phi_i) \\ \sin(\phi_i) & \cos(\phi_i) \end{pmatrix} \cdot \begin{pmatrix} x_j \\ y_j \end{pmatrix}, 1 \le i, j \le M, \tag{1}$$

where ϕ_i is a rotation angle equal to the SIFT orientation of the reference feature f_i.

S-map describes whether other features are inside or outside of a square defined by the reference feature. For image I, its S-map is defined as

$$Smap(i,j) = \begin{cases} 1 & \text{if} \max\left(\left|x_j^{(i)} - x_i^{(i)}\right|, \left|y_j^{(i)} - y_i^{(i)}\right|\right) < s_i, \\ 0 & otherwise \end{cases} \tag{2}$$

where s_i is a half-side-length proportional to the SIFT scale of feature $f_i : s_i = \alpha \cdot scl_i$, α is a constant. The impact of α will be studied in the experiment in Section 4.2.2.

To describe the relative positions more strictly, we advance to general squared maps. For each feature, n concentric squares are drawn, with an equally incremental step of the half-side-length on the image plane. Then, the image plane is divided into $(n+1)$ nonoverlapping regions. Correspondingly, according to the image plane division, a generalized geo-map should encode the relative spatial positions of feature pairs. The general S-map is defined as GS

$$GS(i,j) = \frac{\max\left(\left|x_j^{(i)} - x_i^{(i)}\right|, \left|y_j^{(i)} - y_i^{(i)}\right|\right)}{s_i}, \tag{3}$$

where s_i is the same as that in Eq. (2).

Intuitively, we can also select a ring or circle for image plane division. In such a case, there is no need to adjust the coordinates of local features. We define the corresponding geometric map as GR

$$GR(i,j) = \left\lfloor \frac{d_{i,j}}{s_i} \right\rfloor, \tag{4}$$

where $\lfloor x \rfloor$ denotes the nearest integer less than or equal to x, $d_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$, $s_i = \alpha \cdot scl_i$, scl_i is the scale parameter of SIFT feature v_i, α is a constant.

From the previous discussion, it can be seen that GS and GR are two kinds of geometric coding maps based on different strategies of image plan division. Although similar results can be expected with GS and GR, square in GSC fits the image shape (i.e., rectangle) better than circles or rings. In our experiments, we selected the GS defined in Eq. (3) instead of GR for geometric verification.

3.2.2 *Geometric Fan Coding.* Geometric square coding only considers the relative spatial position in radial direction, and ignores the constraints along horizontal and vertical direction. To overcome this drawback, we propose a Geometric Fan Coding (GFC) scheme. In geometric fan coding, we take each SIFT feature as reference point and divide the image plane into some regular fan regions. Then two fan coding maps, that is, H-map and V-map, are constructed by checking which fan region other features fall into.

Our Geometric Fan Coding (GFC) is inspired by the spatial coding scheme [Zhou et al. 2010]. The key difference is that, key point locations are first adjusted as in Figure 3 before comparing the relative spatial positions. With each SIFT feature as reference point, other SIFT key points' locations are rotated counterclockwise by the SIFT orientation angle of the reference feature. The motivation is to achieve rotation invariance, without the strong constraints that the duplicated patches in two comparison images share the same or very similar spatial configuration, as imposed in Zhou et al. [2010].

Geometric fan coding encodes the relative spatial positions between each pair of features in an image. Based on the adjusted new positions of SIFT feature in Eq. (1), two binary geometric maps, called H-map and V-map, are generated. H-map and V-map describe the relative spatial positions between each feature pair along the horizontal and vertical directions, respectively. They are formulated as follows.

$$Hmap(i,j) = \begin{cases} 0 & \text{if } x_j^{(i)} \le x_i^{(i)} \\ 1 & \text{if } x_j^{(i)} > x_i^{(i)} \end{cases} \tag{5}$$

$$Vmap(i,j) = \begin{cases} 0 & \text{if } y_j^i \le y_i^i \\ 1 & \text{if } y_j^i > y_i^i \end{cases} \tag{6}$$

The geometric maps can be interpreted as follows. In row i, feature f_i is selected as the reference point, and the image plane is decomposed into four quadrants along horizontal and vertical directions. H-map and V-map then show which quadrant other features fall into.

In fact, the representation of geometric context among local features with H-map and V-map is still too weak. We can put forward the geometric fan coding to more general formulations, so as to impose stricter geometric constraints. The image plane can be divided into $4 \cdot r$ parts, with each quadrant evenly divided into r fan regions. Accordingly, two general fan coding maps GH and GV are required to encode the relative spatial positions of all SIFT features in an image.

For a division of image plane into $4 \cdot r$ parts, we decompose the division into r independent subdivisions, each uniformly dividing the image plane into four quadrants. Each subdivision is then encoded independently and their combination leads to the final fan coding maps. In each subdivision, to encode the spatial context of all features by the left-right and below-above comparison, we just need to rotate all the feature coordinates and the division lines counterclockwise, until the two division lines become horizontal and vertical, respectively.

The general fan coding maps GH and GV are both 3D and defined as follows. Specially, with feature f_i as reference, the location of feature f_j is rotated counterclockwise by $\theta_i^{(k)} = \frac{k \cdot \pi}{2 \cdot r} + \phi_i$ degree ($k = 0, 1, \ldots, r-1$) according to the image origin point, yielding the new location $f_j^{(i,k)}(x_j^{(i,k)}, y_j^{(i,k)})$ as,

$$\begin{pmatrix} x_j^{(i,k)} \\ y_j^{(i,k)} \end{pmatrix} = \begin{pmatrix} \cos(\theta_i^{(k)}) & -\sin(\theta_i^{(k)}) \\ \sin(\theta_i^{(k)}) & \cos(\theta_i^{(k)}) \end{pmatrix} \cdot \begin{pmatrix} x_j \\ y_j \end{pmatrix}. \tag{7}$$

Here ϕ_i is the SIFT orientation angle of f_i, as used in Eq. (1). Then GH and GV are formulated as

$$GH(i,j,k) = \begin{cases} 0 & \text{if } x_j^{(i,k)} \le x_i^{(i,k)} \\ 1 & \text{if } x_j^{(i,k)} > x_i^{(i,k)} \end{cases}, \tag{8}$$

$$GV(i,j,k) = \begin{cases} 0 & \text{if } y_j^{(i,k)} \le y_i^{(i,k)} \\ 1 & \text{if } y_j^{(i,k)} > y_i^{(i,k)} \end{cases}. \tag{9}$$

In geometric fan coding, the factor r controls the strictness of geometric constraints and will affect verification performance. We will study its impact in Section 4.2.3.

From the preceding discussion, it can be seen that both geometric square coding and geometric fan coding can be efficiently performed. However, it will take considerable memory to store the whole geometric maps of all features in an image. Fortunately, that is not necessary at all. Instead, we only need keep the orientation, scale, and x- and y-coordinate of each SIFT feature, respectively. When checking the feature matching of two images, we just need geometric clues of these SIFT matches, which will be employed to generate geometric maps for spatial verification in real time. Since the SIFT matches are often only a small set of the whole feature set of an image, the corresponding memory cost on these geometric coding maps is relatively low. The details are discussed in the next section.

3.3 Spatial Verification

Since the focused problem is partial-duplicate image retrieval, there is an underlying assumption that the target image and the query image share some duplicated patches, or in other words, share some local features with consistent geometry. Due to the unavoidable quantization error and visual word

ambiguity, there always exist some false SIFT matches, which will degrade image similarity measurement. To more accurately define the similarity between images, spatial verification with geometric coding can be used to remove such false matches.

Denote that a query image I_q and a matched image I_m are found to share N matching pairs of local features. Then the geo-maps of these matched features for both I_q and I_m can be generated and denoted as (GS_q, GH_q, GV_q) and (GS_m, GH_m, GV_m) by Eq. (3), Eq. (8), and Eq. (9), respectively. After that, we can compare these geometric maps to remove false matches as follows.

Since the general geometric fan coding maps are binary, for efficient comparison, we perform a logical Exclusive-OR (XOR) operation on GH_q and GH_m, GV_q and GV_m, respectively.

$$V_H(i, j, k) = GH_q(i, j, k) \oplus GH_m(i, j, k) \tag{10}$$

$$V_V(i, j, k) = GV_q(i, j, k) \oplus GV_m(i, j, k) \tag{11}$$

Ideally, if all N matched pairs are true, V_H and V_V will be zero for all their entries. If some false matches exist, the entries of these false matches on GH_q and GH_m may be inconsistent, and so are those on GV_q and GV_m. Those inconsistencies will cause the corresponding exclusive-OR result of V_H and V_V to be 1. We define the inconsistency from geometric fan coding as follows.

$$F_H(i, j) = \bigcup_{k=1}^{r} V_H(i, j, k) \tag{12}$$

$$F_V(i, j) = \bigcup_{k=1}^{r} V_V(i, j, k) \tag{13}$$

The inconsistency from geometric square coding is defined as

$$F_S(i, j) = \left| GS_q(i, j) - GS_m(i, j) \right|. \tag{14}$$

Consequently, by checking F_H, F_V, and F_S, the false matches can be identified and removed. Denote

$$T(i, j) = \begin{cases} 1 & \text{if } F_S(i, j) > \tau \text{ and } F_H(i, j) + F_V(i, j) > \beta \\ 0 & \text{otherwise} \end{cases}, \tag{15}$$

where β and τ are constant integers. When τ or β is greater than zero, T in Eq. (15) can tolerate some drifting error of relative positions of local features. The impact of τ and β will be studied in the later experiments in Section 4.2.2 and Section 4.2.3, respectively.

Ideally, if all matched pairs are true positives, the entries in T will be all zeroes. If false matches exist, the entries of those false matches on those coding maps may be inconsistent. Those inconsistencies will cause the corresponding entries in T to be 1. We can iteratively remove such match that causes the most inconsistency, until all remain matches are consistent with each other.

When two images contain multiple partial-duplicated objects and each object has different changes in scale or orientation, the preceding manipulation will only discover the dominant duplicated objects with the largest number of local matches. However, the extension to identify multiple partial-duplicate objects is straightforward. To address this issue, we can first find those matches corresponding to the dominant duplicated object and then focus on the sub-geo-maps of the remaining matches. Those matches corresponding to the second dominant object can be identified in a similar way. Such an operation can be performed iteratively, until all partial-duplicate objects are discovered.

Figure 4 shows two instances of the spatial verification with geometric coding on a relevant image pair and an irrelevant image pair. Initially, both image pairs have many matches of local features. For the upper "Apollo" example, after spatial verification via geometric coding, 9 false matches are identified and removed, while 12 true matches are satisfactorily kept. For the second instance, although

Fig. 4. An illustration of spatial verification with geometric coding on a relevant pair (first row) and an irrelevant pair (second row). (left column): Initial matches after quantization; (middle column): False matches detected by spatial verification; (right column): True matches that pass the spatial verification. (Best viewed in color PDF)

they are irrelevant in content, 17 SIFT matches still exist after quantization. However, by spatial verification, only one pair of matches is kept. With those false matches removed, the similarity between images can be more accurately defined and that will benefit retrieval accuracy. The philosophy behind the effectiveness of our geometric verification approach is that the probability of two irrelevant images sharing many spatially consistent visual words is very low.

4. EXPERIMENTS

4.1 Experiment Setup

Dataset. Our basic dataset contains one million images crawled from the Web. Two ground-truth datasets, that is, DupImage dataset [2011] and Copydays dataset [2008], are used for evaluation, respectively. The descriptions of these two datasets are given in the following.

(1) *DupImage dataset*. DupImage dataset contains 1104 manually labeled partial-duplicate Web images of 33 groups collected from the Web. Images in each group are partial duplicates of each other. Some typical examples are shown in Figure 1, Figure 4, and Figure 11. These ground-truth images are open to the public with the link: http://www.cs.utsa.edu/~wzhou/data/DupGroundTruth Dataset.tgz.

(2) *Copydays dataset*. There are 3369 images in the Copydays dataset generated from 175 original images. Each original image is subjected to three kinds of artificial attacks: "JPEG", "cropping" and "strong". The cropped images suffer from 10% to 80% of the image surface removed. In "JPEG" attacks, each image is scaled to 1/16 (pixels) with nine different JPEG quality factors. Images from the third attacks of "strong" are obtained by printing and scanning, blurring, painting, rotating, and so on. In summary, for each original image, there are nine cropped images, nine JPEG attached images, and 2 to 6 images with "strong" attacks.

Local features. We use the standard SIFT feature [Lowe 2004] for image representation. Key points are detected with the Difference-of-Gaussian (DoG) detector, and a 128-dimensional orientation histogram (SIFT descriptor), together with scale and dominant orientation, is extracted to describe the local patch around the key points. Before feature extraction, large images are scaled to no larger than 400×400.

Since the ground-truth images do not exhibit diverse rotation changes, it will be insufficient to demonstrate the rotation-invariant capability of our geometric coding approach. To address this issue,

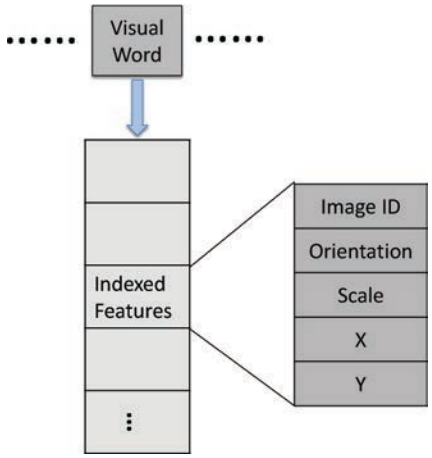

Fig. 5. Inverted file structure for index.

instead, for each query image, we randomly generate an angle ranging from $-\pi$ to π, and rotate the query image counterclockwise by the angle. Since the descriptor and scale in the SIFT feature are invariant to rotation change, we just need to modify the orientation, and x- and y-coordinates of each SIFT feature accordingly in each query image for testing.

Indexing. We use an inverted file structure to index images. As illustrated in Figure 5, each visual word is followed by a list of indexed features that are quantized to the visual word. Each indexed feature records the ID of the image where the visual word appears. Besides, as discussed in Section 3.2, for each indexed feature, we also need keep its SIFT orientation, scale, and the x- and y- coordinate, which will be used for generating geometric coding maps for retrieval.

Evaluation and retrieval. To evaluate the performance with respect to the size of dataset, we build several smaller datasets by sampling the basic one-million dataset. For the DupImage dataset, 100 representative query images are selected from the ground-truth dataset for evaluation comparison. For the Copydays dataset, all original and attacked images are used as queries for evaluation comparison. We adopt mean Average Precision (mAP) [Philbin et al. 2007] to evaluate the performance of all approaches.

In retrieval, each visual word in the query image casts a vote for its matched images. Instead of selecting the *tf-idf* weight [Sivic and Zisserman 2003; Nister and Stewenius 2006] to distinguish different matched features, we simply count the number of SIFT matches that pass our spatial verification. Image similarity is formulated as the number of true matches, with a term to distinguish images with the same amount of true matches by their feature amount.

4.2 Impact of Parameters

The performance of our approach is related with five parameters: visual codebook size, α and τ in GSC, and r and β in GFC. In the following, we will study their impacts with the *DupImage* dataset and select the optimal values.

4.2.1 *Visual Codebook Size.* Visual codebook size reflects the space partition degree of the SIFT descriptor. The larger the visual codebook, the finer the high-dimensional descriptor space is divided. We test three different sizes of visual codebooks on the 1M image database. The result is shown in Table I.

Table I. Mean Average Precision (mAP) and Average Time Cost per Query to Each Visual Codebook

Codebook size	130K	500K	1M
mAP	0.554	0.544	0.540
Time cost (s)	3.53	0.64	0.16

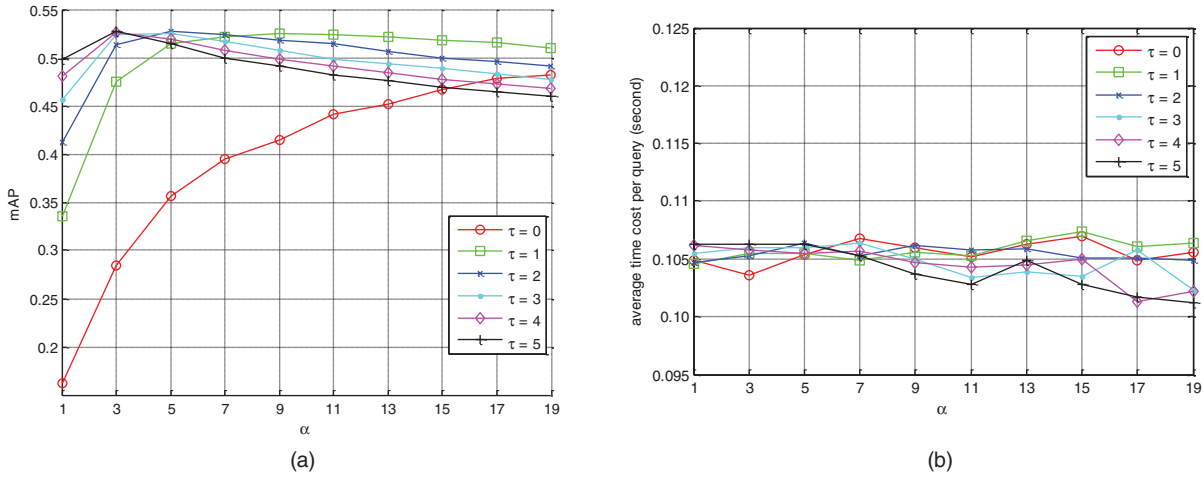

Fig. 6. Performance comparison of geometric fan coding with different α and τ on the one-million image dataset. (Best viewed in color PDF)

From Table I, it can be observed that when the size of descriptor visual codebook increases from 130K to 500K, the mAP drops a little while the time cost per query decreases sharply. When using a smaller codebook, similar features will be more likely to be quantized to the same visual word, which helps to gain improvement in accuracy. Although more false local matches will be incurred, they can be effectively discovered and removed by our geometric verification. To make a trade-off in mAP and time cost, we select the 1M visual codebook, which is used in the later experiments.

4.2.2 *Impact of α and τ.* In geometric square coding, the factor α and τ work together to cast geometric consistency constraints on the relative spatial positions between each pair of SIFT features. We also need evaluate their joint impact on retrieval performance so as to select the optimal values.

We test the performance of our geometric square coding using different values of α and τ on the 1M dataset, without geometric fan coding constraints. Intuitively, smaller α value defines stricter spatial relationships. However, it suffers from the scale detection error in the SIFT feature. Similarly, the parameter τ also tunes the strictness of spatial constraints. Small values of τ will help tolerate digital errors in estimating the scale of features during detection. When τ increases, the imposed constraint will become loose and more noisy matches will be found.

As shown in Figure 6(a), with the increasing of α, the mAP performance first increases, and then gradually drops after reaching its maximum. Since the computational complexity of the geometric square coding map is independent of α and τ, the time cost should be similar with different α and τ, as demonstrated in Figure 6(b). Considering both mAP and average time cost, we select $\alpha = 5$ and $\tau = 2$.

4.2.3 *Impact of r and β.* In geometric fan coding, the factor r determines the division degree of the image plane, while the factor β tunes the strictness of geometric consistency verification. In fact,

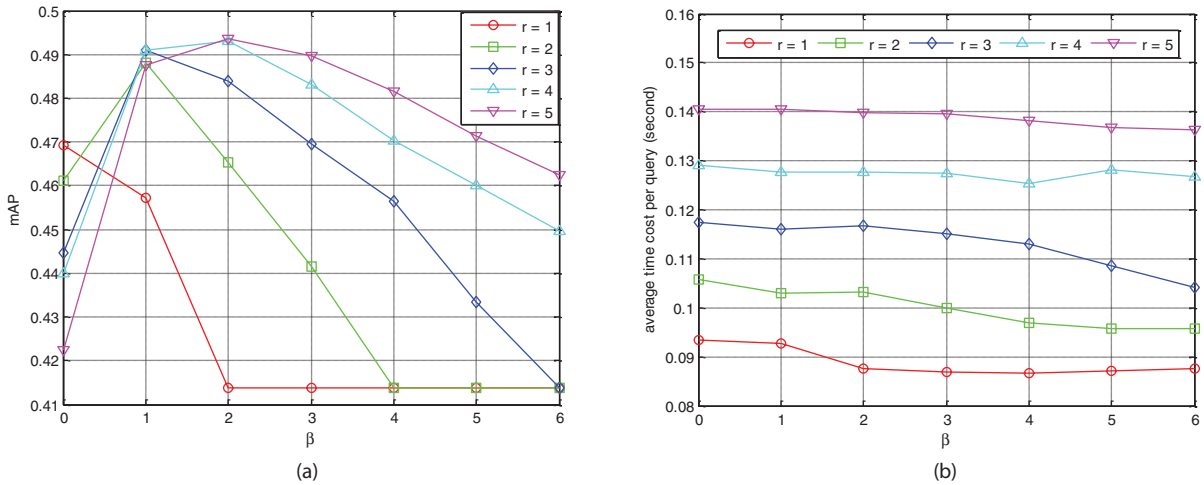

Fig. 7. Performance comparison of geometric fan coding with different r and β on the one-million image dataset. (a) mAP; (b) average time cost per query. The best result is achieved with $r = 4$ and $\beta = 2$. (Best viewed in color PDF)

r and β work interactively to impose geometric constraints on the relative spatial positions among local matches. Therefore, we need to evaluate the impact of them together on retrieval performance.

We test the performance of our geometric fan coding using different values of r and β on the 1M dataset, without geometric squaring coding constraints. The mAP performance and time cost are illustrated in Figure 7. Intuitively, larger values of r and β cast stricter geometric constraint and better performance is expected. However, due to the unavoidable detection errors of the SIFT key point position and SIFT orientation, a strong geometric constraint will be subjected to the drifting error from relatively spatial positions, resulting in low accuracy, as shown in Figure 7(a). For time cost, large r will introduce more computation in the fan coding maps, and increase the required time cost, as shown in Figure 7(b). Considering both mAP and time cost, the best trade-off is made when $r = 4$ and $\beta = 2$.

4.3 Evaluation on DupImage Dataset

Three approaches are considered for comparison. The parameters of those comparison approaches are tuned based on the suggestion in the corresponding papers. The first one is the bag-of-visual-words approach with visual vocabulary tree [Nister and Stewenius 2006], denoted as the "baseline" approach. A visual vocabulary of 1M visual words is adopted. In fact, for the baseline, different sizes of visual codebooks have been tested and the 1M visual codebook is found to generate the best overall performance. The second one is reranking via geometric verification, which is based on the estimation of an affine transformation by a variant of RANSAC [Chum et al. 2004] as used in Philbin et al. [2007]. We call this method "RANSAC". In the experiment, all candidate images with no less than three local matches are involved in the RANSAC-based reranking.

The third one is Spatial Coding (SC) [Zhou et al. 2010], which generates spatial maps for spatial verification. Since spatial coding requires that the duplicated patches in different images share the same or very similar spatial configuration, it cannot directly deal with images with rotation changes. As discussed in Zhou et al. [2010], it can achieve rotation invariance by merging the results of a set of new queries, which are obtained by evenly rotating the query image with a few predefined angles $\frac{k\cdot\pi}{m}$, $k = 0, 1, \ldots, m-1$, where m denotes the rotation times. For example, if $m = 8$, the query image will be rotated in 8 angles: $\frac{0\cdot\pi}{8}$, $\frac{1\cdot\pi}{8}$, $\ldots$, $\frac{7\cdot\pi}{8}$ and therefore generate 7 more new query images for further

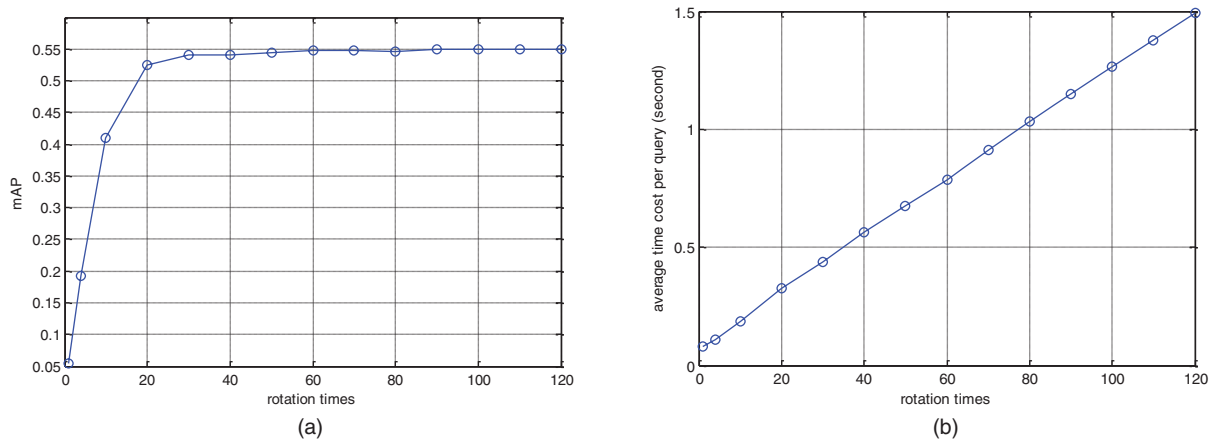

Fig. 8. Performance of spatial coding with different rotation times: (a) mean average precision; (b) average time cost per query.

duplicate image search. We test spatial coding on the one-million image dataset with a 1M visual codebook on different values of m. As shown in Figure 8, when the rotation times increase, its mAP first increases sharply and then keeps stable, while the time cost increases proportionally. When $m = 60$, it reaches the peak MAP. Considering both accuracy and time cost of spatial coding, in the comparison, we select the rotation times of 8, 30, and 60, and denote the corresponding spatial coding approach as "SC(8)", "SC(30)", and "SC(60)", respectively. SC(8) has similar time cost to our Geometric Coding (GC) approach while SC(30) achieves similar mAP performance to our approach. "SC(60)" achieves the best mAP performance with the least rotation times. It can be noted that there is a small gap in mAP between "SC(60)" and GC. "SC(60)" performs slightly better than GC, which is mainly due to the fact that the orientation detection suffers from trivial errors in SIFT extraction. In our geometric coding approach, the orientation detection error will be propagated when local features' coordinates are adjusted by the reference feature's orientation value with Eqs. (1) and (7). Nevertheless, in the spatial coding approach, this error is attenuated through soft quantization in orientation space [Zhou et al. 2010].

We perform the experiments on a server with 2.4 GHz CPU and 8GB memory. Figure 9 illustrates the mAP performance of the comparison algorithms and our Geometric Coding (GC) approach. Figure 10 shows the average time cost per query of all six approaches. The time cost of SIFT feature extraction is not included in all algorithms. Compared with the baseline, our approach is more time consuming, since it is involved with geometric coding and verification. It takes the baseline 0.095 second to perform one image query on average, while for our approach the average query time cost is 0.155 second, 0.06 second more than the baseline. However, our approach increases the mAP from 0.37 to 0.54, an 46% improvement over the baseline.

Our approach is more efficient than SC(8), SC(30), SC(60), and reranking with RANSAC. SC(8) takes comparable time cost with GC, but its mAP is much worse than our approach and even worse than the baseline. Our approach also achieves comparable mAP to spatial coding with 30 rotation times (SC(30)), However, the average time cost per query of SC(30) is 2.8 times that of our approach. Since SC(30) is involved with 29 more rotated new queries to gain similar mAP to GC, more time cost is introduced. When the rotation times increase to 60, the mAP of SC(60) reaches 0.548, with a slight improvement over our GC approach (0.540) but with much higher time cost. The computational time of SC(60) is 0.785 second per query, which is five times that of GC. The reason that GC achieves a little

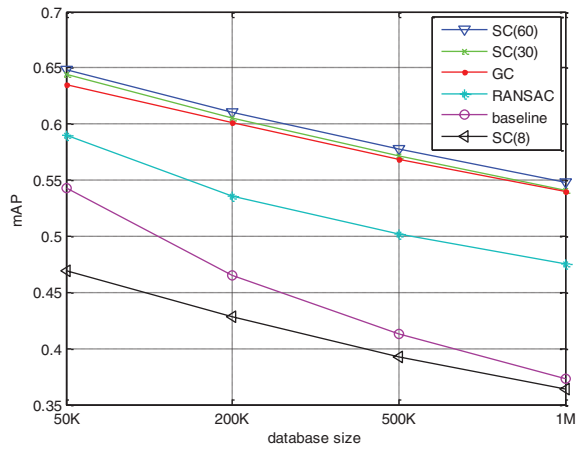

Fig. 9. Comparison of mAP for different methods on the 1M database. (Best viewed in color PDF)

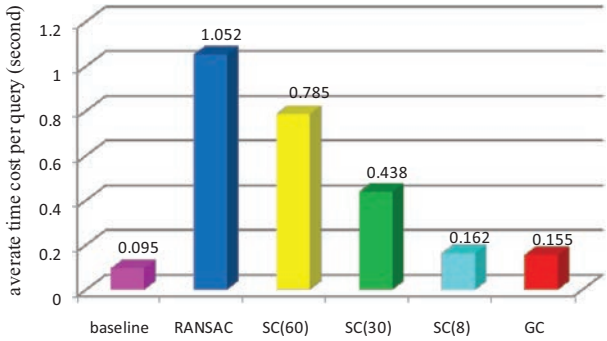

Fig. 10. The average time cost of the comparison methods and our Geometric Coding (GC) approach.

lower mAP than SC(60) is that the orientation detection error from SIFT extraction is propagated to the coordinate adjustment in Eq. (1) and Eq. (7). RANSAC is the most time-consuming approach, due to the affine estimation from many random samplings. It costs 1.052 seconds on average per query, which is 6.7 times more than our approach. However, it is notable that our approach achieves even better mAP performance than the "RANSAC" method. This is because that "RANSAC" only makes use of the coordinate information, while our approach fully exploits geometric clues including the scale, orientation, and spatial position. Therefore, our approach is more powerful in distinguishing those geometric inconsistent matches.

Figure 11 illustrates some sample results using our geometric coding approach on the one-million image dataset. It can be observed that the retrieved results are not only diverse but also contain large changes in color, scale, rotation, significant occlusion, etc.

4.4 Evaluation on Copydays Dataset

With the parameters selected in Section 4.2, we also compare our approaches with the other algorithms on the Copydays dataset with four different sizes of distracting images, that is, 100K, 200K, 500K, and 1M. The mAP results are given in Table II. Since there are four categories each with different number of images, we list the mAP respectively. As can be seen, this dataset is relatively easier than

Fig. 11. Sample results on the one-million image dataset. Query images are shown on the left of the arrow in each row, and highly ranked returned images (selected from those before the first false positive) are shown on the right.

Table II. Comparison of All Methods in Accuracy of Queries from Four Parts of The Copydays Dataset on Four Different Sizes of Database (100K, 200K, 500K, and 1M)

		Baseline	RANSAC	SC(60)	SC(30)	SC(8)	GC
Original	100K	0.9588	0.9588	0.9636	0.9626	0.7992	0.9594
	200K	0.9565	0.9578	0.9626	0.9617	0.7912	0.9586
	500K	0.9533	0.9560	0.9614	0.9603	0.7819	0.9574
	1M	0.9505	0.9545	0.9602	0.9592	0.7755	0.9566
JPEG	100K	0.8894	0.8949	0.9154	0.9141	0.7445	0.9103
	200K	0.8844	0.8918	0.9132	0.9116	0.7360	0.9078
	500K	0.8767	0.8875	0.9099	0.9081	0.7260	0.9042
	1M	0.8705	0.8842	0.9072	0.9051	0.7187	0.9015
Cropping	100K	0.9184	0.9244	0.9352	0.9343	0.7543	0.9308
	200K	0.9132	0.9218	0.9334	0.9325	0.7461	0.9292
	500K	0.9060	0.9179	0.9309	0.9298	0.7363	0.9266
	1M	0.9005	0.9152	0.9287	0.9277	0.7295	0.9246
Strong	100K	0.7575	0.7858	0.8078	0.8056	0.6323	0.7926
	200K	0.7470	0.7803	0.8033	0.8008	0.6240	0.7876
	500K	0.7332	0.7729	0.7976	0.7947	0.6140	0.7817
	1M	0.7236	0.7677	0.7932	0.7899	0.6074	0.7769

Table III. Comparison of All Methods in Average Time Cost on Copydays Images on an One-Million Image Dataset

Approach	Baseline	RANSAC	SC(60)	SC(30)	SC(8)	GC
Average time cost (second)	0.139	6.693	1.286	0.605	0.324	0.283

the aforesaid DupImage dataset and the mAP values of all approaches are relatively higher. Among the four categories, the best results are from queries with original image, while the worst results are obtained with queries from "strong" attacked images. The average time cost per query of all approaches is shown in Table III. The time cost of SIFT feature extraction is not included.

Compared with the baseline, our approach is more time consuming. It takes the baseline 0.139 second to perform one image query on average, while for our approach the average query time cost is 0.283 second, about twice the time cost of the baseline. However, for each category of the ground-truth images, our approach achieves better mAP performance over the baseline.

Similar to the results in Section 4.3, our approach is more efficient than SC(8), SC(30), SC(60), and reranking with RANSAC. SC(8) takes even more time cost than GC, but its mAP is much worse than our approach and even worse than the baseline. Our approach also achieves comparable mAP to SC(30), However, the average time cost per query of SC(30) is about twice of our approach. The mAP of SC(60) is higher on all four categories of images over our GC approach, but with much higher time cost. RANSAC is the most time-consuming approach. It costs 6.693 seconds on average per query, which is 22.6 times more than our approach. Also, our approach achieves better mAP performance than the "RANSAC" method.

5. CONCLUSION

In this article, we propose a novel geometric coding scheme for SIFT match verification in large-scale partial-duplicate image search. The geometric coding consists of geometric square coding and geometric fan coding. It efficiently encodes the global geometric context of local features in an image and effectively discovers false feature matches between images. Our approach can effectively detect duplicate images with rotation, scale change, occlusion, and background clutter with low computational cost. Experiments on two million-scale datasets reveal that our approach outperforms the baseline even following a RANSAC verification. Besides, our approach also attains comparable performance with the spatial coding scheme, but takes much less time.

Our geometric fan coding is inspired by the spatial coding scheme [Zhou et al. 2010]. The key difference is that, key point locations are first adjusted by SIFT orientation and the generated coding maps are invariant to rotation changes. Our GFC can adaptively achieve rotation invariance, which cannot be well addressed by spatial coding. Besides, the spatial verification is performed in a soft manner to tolerate drifting error of relative positions of local features.

Our geometric coding scheme aims to discover Web images sharing duplicated patches. With high accuracy and efficiency, our approach is effective for large-scale partial-duplicate image retrieval. However, when searching general objects where distinctive SIFT features are not repeatable, it may not perform well.

REFERENCES

BAY, H., TUYTELAARS, T., GOOL, L. V. 2006. SURF: Speeded up robust features. In *Proceedings of the 9th European Conference on Computer Vision (ECCV'06)*. 404–417.

BELONGIE, S., MALIK, J., AND PUZICHA, J. 2002. Shape matching and object recognition using shape context. *IEEE Trans. Pattern Anal. Mach. Intell. 24,* 4, 509–522.

CHANG, S.-K., SHI, Q. Y., AND YAN, C. Y. 1987. Iconic indexing by 2-D strings. *IEEE Trans. Pattern Anal. Mach. Intell. 9,* 3, 413–428.

CHUM, O., PHILBIN, J., SIVIC, J., ISARD, M., AND ZISSERMAN, A. 2007a. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *Proceedings of the IEEE 11th International Conference on Computer Vision.* 1–8.

CHUM, O., PHILBIN, J., ISARD, M., AND ZISSERMAN, A. 2007b. Scalable near identical image and shot detection. In *Proceedings of the 6th ACM international Conference on Image and Video Retrieval.* ACM, 1–8.

CHUM, O., PERDOCH, M., AND MATAS, J. 2009. Geometric minhashing: Finding a (thick) needle in a haystack. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 17–24.

CHUM, O., MATAS, J., AND OBDRZALEK, S. 2004. Enhancing RANSAC by generalized model optimization. In *Proceedings of the Asian Conference on Computer Vision.* 812–817.

COPYDAYS, 2008. http://lear.inrialpes.fr/~jegou/data.php

DUPIMAGE, 2011. http://www.cs.utsa.edu/~wzhou/data/DupGroundTruthDataset.tgz

FISCHLER, M. A. AND BOLLES, R. C. 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. ACM. 24*, 6, 381–395.

GAO, Y., WANG, C., LI, Z., ZHANG, L., AND ZHANG, L. 2010. Spatial-Bag-of-Features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 3352–3359.

GOOGLE SIMILAR IMAGE SEARCH, 2009. http://similar-images.googlelabs.com/

HOANG, N. V., GOUET-BRUNET, V., RUKOZ, M., AND MANOURIER, M. 2010. Embedding spatial information into image content description for scene retrieval. *Pattern Recogn. 43*, 9, 3003–3012.

JEGOU, H., DOUZE, M., AND SCHMID, C. 2008. Hamming embedding and weak geometric consistency for large scale image search. In *Proceedings of the 10th European Conference on Computer Vision.* 304–317.

JEGOU, H., HARZALLAH, H., AND SCHMID, C. 2007. A contextual dissimilarity measure for accurate and efficient image search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 1–8.

LOWE, D. 2004. Distinctive image features from scale-invariant key points. *Int. J. Comput. Vis. 60,* 2, 91–110.

MATAS, J., CHUM, O., URBAN, M., AND PAJDLA, T. 2002. Robust wide baseline stereo from maximally stable extremal regions. In *Proceedings of the British Machine Vision Conference.* 384–393.

NISTER, D. AND STEWENIUS, H. 2006. Scalable recognition with a vocabulary tree. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2161–2168.

OLIVA, A., AND TORRALBA, A. 2001. Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int. J. Comput. Vision 42,* 3, 145–175.

PHILBIN, J., CHUM, O., ISARD, M., SIVIC, J., AND ZISSERMAN, A. 2007. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 1–8.

PHILBIN, J., CHUM, O., ISARD, M., SIVIC, J., AND ZISSERMAN, A. 2008. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 1–8.

SIVIC, J. AND ZISSERMAN, A. 2003. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision.* 1470–1477.

SMITH, J. R. AND CHANG S.-F. 1996. VisualSEEk: A fully automated content-based image query system. In *Proceedings of the 4th ACM International Conference on Multimedia.* 87–98.

TANG, J., YAN, S., HONG, R., QI, G.-J., AND CHUA T.-S. 2009 Inferring semantic concepts from community-contributed images and noisy tags. In *Proceedings of the ACM International Conference on Multimedia.*

TANG, J., LI, H., QI, G.-J., AND CHUA T.-S. 2010. Image annotation by graph-based inference with integrated multiple/single instance representations. *IEEE Trans. Multimedia 12,* 2, 131–141.

TINEYE, 2008. http://www. Tineye.com

WANG, X., YANG, M., COUR, T., ZHU, S., YU, K., AND HAN, T. X. 2011. Contextual weighting for vocabulary tree based image retrieval. In *Proceedings of the International Conference on Computer Vision.*

WU, Z., KE, Q., ISARD, M., AND SUN, J. 2009. Bundling features for large scale partial-duplicate web image search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 25–32.

ZHANG, S., TIAN, Q., HUA, G., HUANG, Q., AND LI, S. 2009. Descriptive visual words and visual phrases for image applications. In *Proceedings of the ACM International Conference on Multimedia.* 75–84.

ZHANG, S., HUANG, Q., HUA, G., JIANG, S., GAO, W., AND TIAN, Q. 2010. Building contextual visual vocabulary for large-scale image applications. In *Proceedings of the ACM International Conference on Multimedia.* 501–510.

ZHANG, Y., JIA, Z., AND CHEN, T. 2011. Image retrieval with geometry-preserving visual phrases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 809–816.

ZHAO, W.-L., WU, X., AND NGO, C.-W. 2010. On the annotation of web videos by efficient near-duplicate search. *IEEE Trans. Multimedia 12,* 5, 448–461.

ZHOU, W., LU, Y., LI, H., SONG, Y., AND TIAN, Q. 2010. Spatial coding for large scale partial-duplicate web image search. In *Proceedings of the ACM International Conference on Multimedia.* 511–520.

ZHOU, W., LI, H., LU, Y., AND TIAN, Q. 2011. Large scale image search with geometric coding. In *Proceedings of the ACM International Conference on Multimedia.*

Received September 2011; revised January 2012; accepted March 2012