I J C T A, 9(2) 2016, pp. 781-795

New Harris Corner and Edge Descriptor for Partial-duplicate Mobile Search

[1*]K. Roselin Mary Clare and [2]M. Hemalatha

ABSTRACT

With large-scale duplication on visual search over mobile devices, local descriptor of image is supposed to be distinguished, well-organised and compacted. The modern image retrieval algorithms use local invariant to speed up matching and quantization which relies on visual codebook. The codebook typically consumes huge quantity of memory space during online recovery stage and demands considerable amount of offline resources, since it contains millions of visual words. The current study on Edge-SIFT shows, it is important to study more efficient interest point detectors that identify stable regions with rich edge clues. Edge-SIFT produce more reliable position, scale and orientation clues, which largely decrease the registration error. But significant points to detect the spatial clues are not spot lighted, which improves the matching duplicate image patches and discriminative power. To address this issue, propose a new enhanced Harris corner and Edge SIFT algorithm (HC-ESIFT) for large partial duplicate mobile image search. In the proposed method first employ a novel Edge SIFT scheme to extract reliable edges of local image patches. Afterwards, enhance the Edge SIFT by Harris corner point detection with the binary descriptor of these local patches. Scaling and orientation of normalized image patches using binary edge maps are derived from HC-ESIFT. To project the strongly distinguished power with compressed representation, the use of chi square kernel learning function and HC-ESIFT provides compressed binary edge map and preserves both locations and orientation. This work promotes flexible online verification, using indexed framework and similarity measurement. Hybrid Hierarchical K-Centres (HHKC) clustering generates enormous visual vocabulary tree with HC-ESIFT method to achieve precise and quick matching of visual tree. The proposed HC-ESIFT method is independent on quantizing and verifying the training images and generalizing the image datasets.

Index Terms: Image local descriptor, large-scale image search, Hybrid Hierarchical K-centres (HHKC) Clustering and, mobile vision.

1. INTRODUCTION

In view of the fact that tablets and smart phones percolates over millions of people to make use of mobile as primary internet provider since 2012 with the increasing popularity. The highly equipped camera phones on mobile devices provide convenient and natural portals to the digital world space. Compared with traditional computer, the advantages of mobile device, such as flexible and freshens user experience, convenience, easy access, etc., are appealing to users. With the increased popularity, faster growth in development and huge advantages mobile phones provides platform for research and applications of multimedia applications.

Content Based Image Retrieval (CBIR) shows gradual advancement in large scale image database over the last few decades. The methods proposed in paper [1][3][4] exploits the use of invariant local features to signify the bag-of-visual- words (BoW) model. BoW shows excellent scalability and retrieval precision while indexing image datasets in large scale. The post-processing techniques like Query expansion [6] and spatial verification [2][4][5] enhances the accuracy. The problem to proceed with content-based image is, performing visual matching effectively and reliably over images.

[1] Research Scholar, Research and Development Centre, Bharathiar University, Coimbatore, India.
[2] Dean sciences, Dr. SNS Rajalakshmi College of Arts and Science, Coimbatore, India.
[*]Corresponding Author: *E-mail: roselin.scholar@gmail.com*

The descriptor quantifies the visual word in the codebook, by matching the hash values (i.e., the visual word ID) of the local feature with the use of conventional BoW. Furthermore, the complexity of matching the features from quadratic to linear is lowered drastically. To decrease the multifaceted hashing and quantization from linear to logarithm, a hierarchical codebook with millions of leaf nodes and huge vocabulary tree [7] is adopted. The scalability of local feature matching and quantization has been increased using this approach. There are two main issues encountered with the use of visual codebook. Initially, codebook gets through huge amount of memory while in online and necessitates considerable amount of resources to provide offline training. Secondly, the complexity in controlling the feature quantizing error with vector quantization is not that much easy. For instance, 128D SIFT descriptors [8] require hundreds of megabytes to amass runtime with millions of training descriptors.

However, as illustrated in many works [8-9], to increase the accuracy, the visual vocabulary shows extra attempts and differentiated ability. Specifically, traditional visual vocabulary shows shortcomings such as containing lots of noisy visual words, losing spatial clues, large quantization errors, etc. To overcome these issues, existing works are mostly focused on three aspects, i.e., visual vocabulary generation [10], indexing [11-12], and post verification in online retrieval [13-14]. For visual vocabulary generation, different algorithms have been proposed to depress noisy visual words and to improve the discriminative power. For image indexing, Jegou *et al*. [11] propose to compress local descriptors into compact codes for efficient image similarity computation. For post verification in online retrieval stage, the mismatched visual words are identified and removed between the images to improve the accuracy on computation. But many of current works on large-scale partial duplicate image search are not fitted for image search in computation sensitive scenarios.

Traditional BoWs representation and depress the mismatched visual words with post spatial verification. Hence, conclusion that histogram based descriptors perform well in visual classification and recognition tasks , which are generally built on statistical features and statistical models of the whole image, but might still not be the most optimal feature for large-scale partial-duplicate image search, which is based on the near-duplicate local image patch matching. There are three properties in local descriptors to represent effective BoWs in partial-duplicate image search over mobile podium. They are 1) high discriminative power, i.e., preserve spatial and visual contexts in image patches; 2) high efficiency, i.e., extraction, similarity measurement, and matching should be efficient to compute; and 3) compactness, i.e., descriptor should be compact to store and transmit.

To improve the efficiency of Edge-SIFT furthermore [15] the study on more efficient interest point detectors that identify stable regions with rich edge clues. The desirable detector is also expected to extract scale and orientation clues and be able to estimate the affine transformations. Hence, the final Edge-SIFT could be adjusted to be more robust to affine changes. One possible solution is to detect stable corner points from the binary edge maps which are focused in this research work.

Based on the scales and orientations of interest points, normalize the image patches into fixed scale and orientation for edge map extraction. Based on local auto-correlation function, the Harris corner detector is executed. Here, the changes in the signal with patch shift are measured using local auto-correlation function in different directions. Then decompose the edge map into different sub-edge maps, according to the directions of edges. To make Edge-SIFT more robust to registration errors caused by inaccurate interest point localization, affine transformations, etc., expand the edges in each sub-edge map to make edges in nearby locations can be considered for similarity computation. The conjunction of resulting sub-edge maps is hence taken as the initial Edge-SIFT. The extracted initial edge shift *sparse*, *i.e.*, lots of bins (bits) are 0-value is observed. To make the final Edge-SIFT more compact and hence improve the efficiency of similarity computation, propose to compress it. Here use a kernel function to select the discriminative bins. Similar to the feature s*election strategy collect a dataset set, where the relevance kernel value between images are labelled. Edge-SIFT are compact, i.e., 384 bit, and are efficient for similarity computation, expect to get a

compact and efficient vocabulary tree suitable for mobile applications. Visual vocabulary tree can be generated through Hybrid Hierarchical K-Centers (HHKC) clustering with the defined similarity measurement. A novel indexing framework with fast online verification is proposed. Edge-SIFT are binary, compact and allows for fast similaritycomputation. Experimental results verify the validity of proposed feature and indexing algorithms. Conclude that this work is more suitable for large-scale partial-duplicate image retrieval task on mobile platform than SIFT.

2. BACKGROUND KNOWLEDGE

With the fast development of RISC (Reduced Instruction Set Computer) processor, mobile camera, displaying technology, wireless network, mobile devices have become more powerful, ubiquitous and important to users. Currently, mobile visual search has become a popular research topic for both the academic and industrial communities. So mobile visual search methods have been uses an image processing methods for retrieval. To exploit classic inverted index and visual word models [16], recent researches concentrates on invariant local features for scalable image search. Feature quantization, feature extraction, image ranking and image indexing are the four important key modules for image searching framework.

SIFT [9] extracted from MSER [17], Difference of Gaussian (DoG) [9], and Hessian affine detector [18] is a most effective and popular feature extraction. To provide a robust match in 3D viewpoint, change in illumination and addition of noise to attain a sustainable range of affine distortion using a invariant rotation and image scaling [9]. The single feature has high probability over large database which is highly distinctive for huge images.

Designing a local descriptors shows huge effort in providing efficiency and discriminability as such in the SURF [19] and edge-SIFT [15]. Coined SURF (Speeded-Up Robust Features) presents a invariant descriptor and scalable rotation detector to provide distinctiveness, repeatability, and robustness thus making computation faster comparatively. This can be achieved based on image convolutions by developing a strengthen descriptors and detectors. Thus the detection, description, and matching steps are combined to measure detectors and descriptors. This kind of descriptors is not suitable for image searching in mobile devices. Hence, the descriptors are added with feature quantization to improve the results and to be applied in mobile environment.

Visual words are represented by local descriptors by mapping or hashing one or more visual words at the feature quantization level [16]. The matching between the images plays an important role in identifying the code book using the clustering techniques. The techniques are listed as approximate –K means (AKM) [20], k–means [16] and hierarchical k means [8]. The quantization of feature is a hash function of the local feature.

A larger and discriminative vocabulary is used in vocabulary tree [8] which experimentally shows a drastic change in the quality improvement. The most noteworthy property of this scheme is the tree straightforwardly indicates quantization. The feature quantization and indexing is integrated to form a common idea. To project the power of vocabulary tree, the quality is evaluated by retrieving the ground database. The SIFT descriptor represents the ID code in the visual word by searching the vocabulary tree hierarchically.

To build a quantization method and a vocabulary tree [20] a scalable method is used to randomize the tree to attain the ground truth. Quantization shows the major impact on quality retrieval. An efficient spatial verification is added to improve the query performance and to show consistent improvement in search quality by re-ranking the results experimentally. The visual vocabulary is larger with less margin.

A hard-decision strategy causes some errors in quantization which directs to miss match while vector quantization. The mobile devices must be loaded with large visual vocabulary in the memory and it is

highly unfeasible to load if the data is too large, which leads to expensive cost while computing vector quantization algorithms.

He *et al.* [2] proposed geometric verification and boundary re-ranking to improve retrieval accuracy and to reduce vector quantization error using hash based search framework. The proposed work is based on ”Bag of Hash Bits” (BoHB), here the local feature is encoded to small number of hash bits to represent the bag of hash bits instead of quantizing visual words. The proposed method benefits cheap memory, low transmission cost and computation on the mobile side etc. to solve the associated challenges in mobile search.

For searching the images in a large scale databases, an inverted index structure are adopted directly [16]. The query image and the database images can be identified easily with the help of inverted index list. Using weighted formulation [16] the database image similarity with query can be evaluated. The disadvantage of this approach is the storage of 224-bit per feature in the inverted indexes which leads to huge memory usage. Hence, kernel function is used to measure the efficiency of the similar images. Thus, it achieves the high precision without affecting retrieval accuracy.

Calonder *et al.* [21] proposed Binary Robust Independent Elementary Features (BRIEF) descriptor for computing intensity difference tests between image patches of paired samples. BRIEF shows higher discrimination in computing simple intensity test using few bits. With hamming distance, descriptor similarity is evaluated, which is effective in computing L_2 norm. Finally, fast build and match has be done. The major advantage in BRIEF is speed, reliability, robustness with boundary tolerance to transformations and distortions.

A novel query-sensitive ranking algorithm [22] proposed by Zhang et al to search the –neighbours for retrieval of images, to rank PCA- in which the precision feature matching is effectively improved but the risk is increased by some missing matches.

From the literature compared with SIFT, despite of the clear advantage in speed, these compact descriptors show limitations in the aspects of descriptive power, robustness, corner point detection and generality. Moreover, similar to SIFT, many of the existing compact descriptorsare also based on the statistic clues in the local image patches,hence also loose spatial clues that are important for discriminativepower. Therefore, discriminative, efficient, and compactlocal descriptors are still high desired for mobile applications

3. PROPOSED EDGE AND CORNER POINT BASED SIFT EXTRACTION

The framework for proposed Harris Corner and Edge-SIFT extraction is illustrated inFig. 1. As illustrated in the Fig. 1, first extract imagepatches surrounding the interest points. Then, based on thescales and orientations of interest points, normalize theimage patches into fixed scale and orientation for edge mapextraction. Based on local auto-correlation function, the Harris corner detector is executed. Here, the changes in the signal with patch shift are measured using local auto-correlation function in different directions. Then decompose the edge map into differentsub-edge maps, according to the directions of edges. Tomake Edge-SIFT more robust to registration errors caused byinaccurate interest point localization, affine transformations etc., expand the edges in each sub-edge map to makeedges in nearby locations can be considered for similaritycomputation. The conjunction of resulting sub-edge maps ishence taken as the initial Edge-SIFT. It can be observed that the extracted initial Edge-SIFT are *sparse, i.e.,* lots of bins (bits) are 0-value. To make the final Edge-SIFT more compact and hence improve the efficiency of similarity computation, propose to compress it. Here use a kernel function to select the discriminative bins. Similar to the feature selection strategy collect a dataset set, where the relevance kernel value between images are labelled. Edge-SIFT are compact, i.e., 384 bit, and are efficient for similarity computation, expect to get a compact and efficient vocabulary tree suitable for mobile applications. Visual vocabulary tree can be generated through clustering with the defined similarity measurement.

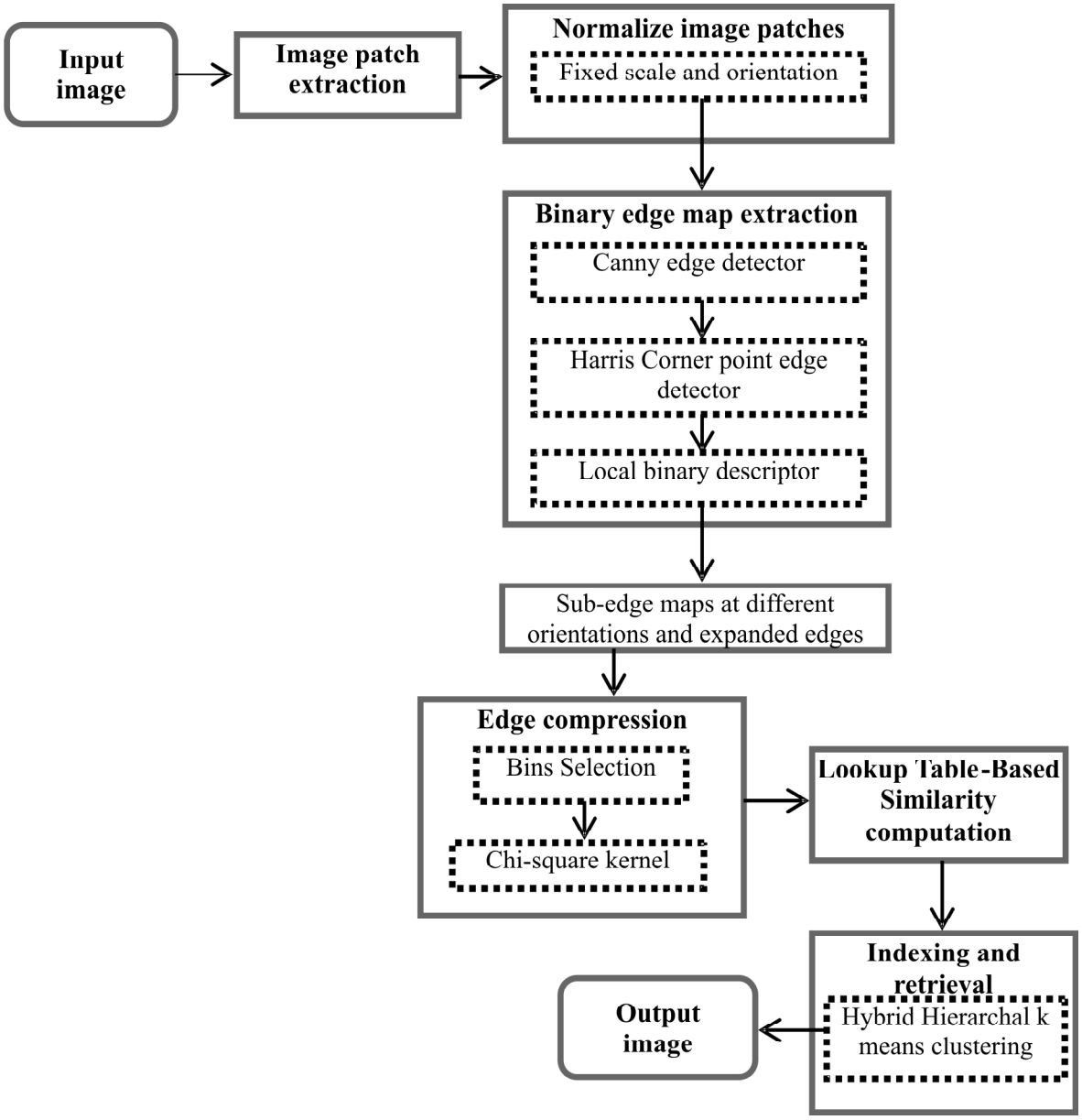

Figure 1: Proposed Edge and corner point based SIFT extraction

Interest Point Detection: Leverage the approach of SIFT for interest point detection. As illustrated by Lowe [9], there are three steps in this approach

1) Scale-space extrema detection: a series of DoG images with different scales are computed. The sample points in DoG images that are larger or smaller than all of their 26 neighbours in both the current image and two adjacent images in scale space are identified as candidate interest points. In this step, candidate interest points and their scale clues can be extracted.

2) Keypoint localization: more accurate interest point locations are assigned. Meanwhile, unstable points are discarded.

3) Orientation assignment: dominant orientations of interest points are computed by summarizing the pixel gradients in their corresponding image patches and selecting the most dominant directions. After interest point detection, get the location, scale, and orientation of each interest point.

In the following steps, introduce how extract image patches and achieve scale invariance and rotation invariance.

Image Patch Extraction and Normalization

Based on the detected interest points, first extract image patches around interest points. The size of extracted image patch corresponding to interest point i is defined as:

$$R_i = r \cdot scale_i \tag{1}$$

where scale denotes the scale of an interest point. It can be contingent that, larger r = 2.5 be in contact to larger image patches, which contain spatial clues and edges, thus it helps to improve the distinguished power of the edge descriptor. However, larger r also increases the computational cost.

Edge Descriptor Computation: From the D × D sized image patch, first utilize canny detector [23] for edge map extraction for its high efficiency and reasonably good performance.

Harris Corner Point Detection Computation: Corners in images represent a lot of important information. Extracting corners accurately is significant to image processing, which can reduce much of the calculations. In a variety of image features, corners are not affected by illumination and have the property of rotational invariance. They are only about 0.05% in the whole pixels. Without losing image data information, extracting corners can minimize the processing data. Therefore, corner detection has practical value and it plays an important role in scale space theory, motion tracking [24], image matching [25]. This detector is based on the local auto-correlation function of a signal, which measures the local changes of the signal with patches shifted by a small amount in different directions. Given a shift ("x, "y) and a point (x, y), the auto-correlation function is defined as [26]:

$$E(x,y) = \sum_{\Omega(x,y)} G_{x,y}(x_i, y_i)[I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta x)]^2 \tag{2}$$

where $I(\cdot,\cdot)$ → image function,

(x_i, y_i) → points in the window $\Omega(x, y)$ which is centered on (x, y),

$G(x,y)(\cdot,\cdot)$→ Gaussian kernel function also centered on (x, y). If displacement is small, the Taylor expansion is truncated by first order series with approximated shift image.

Thus get the formula below:

$$I(x_i + \Delta x, y_i + \Delta y) \approx I(x_i, y_i) + \left[I_x(x_i, y_i) I_y(x_i, y_i)\right] \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \tag{3}$$

where $I_x(\cdot,\cdot)$ and $I_y(\cdot,\cdot)$ denote respectively the partial derivatives in the x and y direction. Then, by substituting Eq. 3 into Eq. 2, get:

$$E(x,y) = \sum_{\Omega(x,y)} G_{x,y}(x_i, y_i)\left(\left[I_x(x_i, y_i) I_y(x_i, y_i)\right]\right)^2 = [\Delta x\ \Delta y] M(x,y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \tag{4}$$

where M(x, y) is as

$$M(x,y) = \begin{bmatrix} \sum_{\Omega(x,y)} G_{x,y}(x_i, y_i) I_x^2(x_i, y_i) & \sum_{\Omega(x,y)} G_{x,y}(x_i, y_i) I_x(x_i, y_i) I_y(x_i, y_i) \\ \sum_{\Omega(x,y)} G_{x,y}(x_i, y_i) I_x(x_i, y_i) I_y(x_i, y_i) & \sum_{\Omega(x,y)} G_{x,y}(x_i, y_i) I_y^2(x_i, y_i) \end{bmatrix} \tag{5}$$

The local neighbourhood intensity structure is captured by the matrix M(x, y). Assume, the matrix M(x, y) has the eigen values λ_1, λ_2 respectively. The eigenvalues form an invariant description with respect to rotation. For each pixel (x, y) , there are three principal cases depending on the eigenvalues:

1. The local auto-correlation function disappears in any direction, if λ_1, λ_2 are small. The pixel (x, y) has constant intensity approximately around the image region.

2. The local auto-correlation function appears in ridge shape, when there is a change in the orthogonal direction and in the local shift of the matrix M(x, y) which means one of the eigen value is higher and the other value is lower. This indicates that an edge passes (x, y);
3. A shift in any directions, significantly increases the matrix M (x, y) which means that both the eigen values are higher and local auto-correlation function hits sharp peak. This indicates that there is a corner on (x, y).

In order to avoid the calculation of the two eigenvalues, use instead the response function R defined in formula 6 to distinguish the three cases above.

$$R(x,y) = \det\big(M(x,y)\big) - k\ \Big(trace\big(M(x,y)\big)\Big)^2 \tag{6}$$

Note that k = 0.04 is an empirical value. Therefore, a point is considered as a corner if R ≥ s where s > 0 is a certain fixed threshold predefined by the user. After edge extraction and corner point extraction, image patches become $D \times D$ bit binary edge maps, where values of edge pixels are 1 or 0, otherwise. The edge map can be regarded as a binary local descriptor containing D^2 bits. Note that, extracting edge maps from scale normalized patches makes the response of canny detector more stable and robust to image blur and scale changes than direct edge extraction from original images. According to the edge pixel and corner point pixel matching criteria, the similarity measure can be formalized as

$$Sim(A,B) = 2.\frac{\sum_{i=1}^{D^2} Hit(a_i,b_i) * R(a_i,b_i)}{(N_A + N_B)} \tag{7}$$

$$Hit(a_i,b_i) = \begin{cases} 1 & if\ a_i.b_i = 1\ , \left|\theta_a^{(i)} - \theta_b^{(i)}\right| \le \varepsilon \\ 0 & \end{cases} \tag{8}$$

$$R(a_i,b_i) = \begin{cases} 1 & if\ a_i.b_i = 1\ , R \ge s \\ 0 & \end{cases} \tag{9}$$

where A and B are two binary descriptors, a_i, b_i are the values of the i th bit in these two descriptors. N is the no. of edge pixels i.e., the nonzero bits, in a descriptor. θ is the orientation of the edge pixel, and ε is a threshold. Intuitively, edge pixels in the same location with orientation difference smaller than ε would be considered as a match. In Eq. (7), the orientation of each edge pixel needs to be computed online, making it expensive to compute. One possible solution to speed up without losing orientation constraint is to quantize the edge pixels in sub-vectors representing different orientations. Improve robustness to registration error is to loose thelocation constraint in similarity measurement, *i.e.*, edges fromnearby locations could be also matched. The correspondingsimilarity measurement can be reformed as:

$$Sim(A,B) = 2.\frac{\sum_{i=1}^{4*D^2} Hit(a_i,b_i) * R(a_i,b_i)}{(N_A + N_B)} \tag{10}$$

$$Hit(a_i,b_i) = \begin{cases} 1 & if\ a_i.b_i = 1\ , \left|l_a^{(i)} - l_b^{(i)}\right| \le w \\ 0 & \end{cases} \tag{11}$$

$$R(a_i,b_i) = \begin{cases} 1 & if\ a_i.b_i = 1\ , R \ge s \\ 0 & \end{cases} \tag{12}$$

where denotes the location of an edge pixel, and w is a threshold controlling the strictness of location constraint. However, (10) is also more expensive so propose an edge expansion strategy to acquire an improved edge descriptor. Intuitively, a binary vector can be compressed by removing the sparse bins, which consistently shows 0-value [15]. Define the compactness of the kthbin in the initial Edge-SIFT descriptor with Eq. (13), i.e.,

$$\chi_k = \sum_{i=1}^{N} v_k^{(i)} / N \tag{13}$$

where N denotes the total number of collected edge descriptors from a dataset, and $v_k^{(i)}$ is the value of the kth bin in the ith descriptor. Therefore, set a threshold for descriptor compression. Specifically, bins with compactness below the threshold will be discarded.

Discriminative Bins Selection: According to the above mentioned strategy, to compress initial Edge-SIFT as well as to preserve its discriminative power need to choose an ideal threshold. However, such threshold is hard to decide. To conquer this issue, first select several initial bins with high compactness from initial Edge- SIFT, and then identify and add discriminative bins to get the final compressed Edge-SIFT.The matching value is determined between two images A and B is computed according to the number of matched descriptors between them. Specifically, assume that there are two images $A = \{d_A^{(k)} \in \mathbb{R}^d, k = 1, 2, \ldots, N_A\}, B = \{d_B^{(k)} \in \mathbb{R}^d, k = 1, 2, \ldots, N_B\}$, where, $\rightarrow$ d-dimensional local edge descriptor, in which the values are extracted from the images.

A measurement of the generic image-level is defined as,

$$S(A, B) = f\left(\left[k\left(d_A^{(k)}, d_B^{(k)}\right)\right], \forall d_A^{(k)} \in A, d_B^{(k)} \in B\right) \tag{14}$$

Where N$\rightarrow$ total number of local descriptors, i.e., 'd', in an image. where$\left[k\left(d_A^{(k)}, d_B^{(k)}\right)\right] \rightarrow$ local kernel matrix of a feature pair combinations of A,B respectively. $f(\cdot) \rightarrow$ mapping function from local to set-level kernel matrix. Here use χ^2 kernel [27]

$$\chi^2 = K(A, B) = \frac{AB}{A + B} \tag{15}$$

select 'n' initial compact bins, by running the iteration for 'm' times, we obtain a m+n bit compressed Edge- SIFT. The number of m can be flexibly adjusted to seek a trade-off between compactness and discriminative power.

Lookup Table-Based Similarity Computation: As mentioned above, each compressed binary Edge-SIFT can be represented as a list of basic units, i.e., bytes. Each byte can be represented as an integer code with value ranges between [0, 255]. Therefore the similarity computation between two Edge-SIFT descriptors is transformed as the similarity computation between two lists of integer codes. Hence formulate the similarity computation [15] as, i.e.,

$$FastSim(A, B) = \frac{2.\sum_{i=1}^{U} MEPN\left(C_A^{(i)}, C_B^{(i)}\right)}{\sum_{i=1}^{U} TEPN\left(C_A^{(i)}, C_B^{(i)}\right)} \tag{15}$$

where U is the number of integer codes, i.e., C in Edge-SIFT. Suppose the size of Edge-SIFT is 384 bit, its U would be 48. MEPN(·,·) and TEPN(·,·) return the Matched Edge Pixel Number and Total Edge Pixel Number in two codes, respectively.

Indexing and retrieval: To generate BoWs representation, first quantize Edge- SIFT into code words. SIFT into code words. Visual vocabulary tree can be generated through clustering with the defined similarity measurement. As a popular clustering algorithm, hierarchical K-means is generally efficient for visual word generation. The visual codes in the initial stage of the algorithm will not be changed once it is initiated, this is the main problem encountered in the hierarchical clustering. To address this problem, Hybrid Hierarchical K-Centers(HHKC) algorithm [28], has been proposed which uses bottom-up (Unweighted Pair Group Method with Arithmetic Mean (UPGMA) and top-down (K-centers) hierarchical clustering algorithms. The average database image and pair wise similarity queries are measured using the Group Average Algorithm (unweighted pair group method with arithmetic mean (UPGMA)).

$$FastSim\left(VC_i^Q, VC_i^D\right) = \frac{1}{N_A . N_B} \sum_{A \in VC_i^Q, B \in VC_i^D} FastSim(A, B) \tag{16}$$

Where,

FastSim(*A*,*B*) → shows the similarity between two images or clusters, which is represented as vectors. If both the centroids ends in same cluster, the visual code word belongs to the same cluster, when UPGMA finds K clusters in centroids. Therefore, use K-centers clustering instead. Different form K-means, the cluster center of K-centers is simply updated as the data point having the maximum similarities with the other data points in the same cluster. BoWs representation is computed by quantizing local features into visual words. Hence, quantization error is inevitable and may degrade the retrieval performance. To decrease quantization error, divide Edge-SIFT after discriminative bins selection into two parts: the former selected β bins are called as Quantization Code (QC) and the latter selected ² bins are called as Verification Code (VC). QC is utilized for visual vocabulary tree generation and Edge-SIFT quantization, i.e., BoWs representation computation. VC is kept in the index file for online verification. The indexing strategy is based on the standard inverted file indexing framework. Differently, each term in the index list contains extra verification code for online verification. The corresponding online image similarity computation can be represented as:

$$S(Q, D) = \sum_{i=1}^{4 \times D^2} IDF_i FastSim\left(VC_i^Q, VC_i^D\right) \tag{17}$$

where Q and D are query and one of the database images respectively. i denotes one of their matched visual words. IDF_i means the Inverse Document Frequency of visual word i in the image index. VCQ i is the auxiliary code of the Edge-SIFT descriptor in image Q, whose main code is quantized as visual word i. Suppose the mobile image retrieval is implemented based on a client-server architecture, where the server maintains an image index and the mobile device uploads queries and receives retrieval results. With the proposed retrieval framework, two kinds of information should be sent for query from mobile devices, i.e., visual word ID and VC of each Edge- SIFT. Hence, keeping a larger VC would potentially improve the retrieval performance, but produces more transmission cost. In addition, larger VC may make QC too sparse to generate valid visual vocabulary and BoWs representation. Hence, parameters α and β can be flexibly adjusted to chase a reasonable trade-off between retrieval accuracy and transmission cost.

4. EXPERIMENTS AND EVALUATIONS RESULTS

In this section use Oxford Building [29] for testing the effects of different parameters and evaluating the validity of Edge- SIFT compression. There are 5062 images in the oxford building dataset, collected from a particular landmark by searching flickr. The collections are made manually from 11 different landmarks representing 5 potential queries to produce a comprehensive truth. Thereby it gives 55 set of queries to evaluate which the retrieval system. The relevance degrees between queries and dataset images have been manually annotated. Four possible labels has been generated for all the landmarks and the images in the systems

Good - A good, apparent picture of the object/building.

OK - More than 25% of the thing is clearly able to be seen.

Bad - The object is not nearby.

Junk - Less than 25% of the object is observable, or there are high levels of occlusion or deformation.

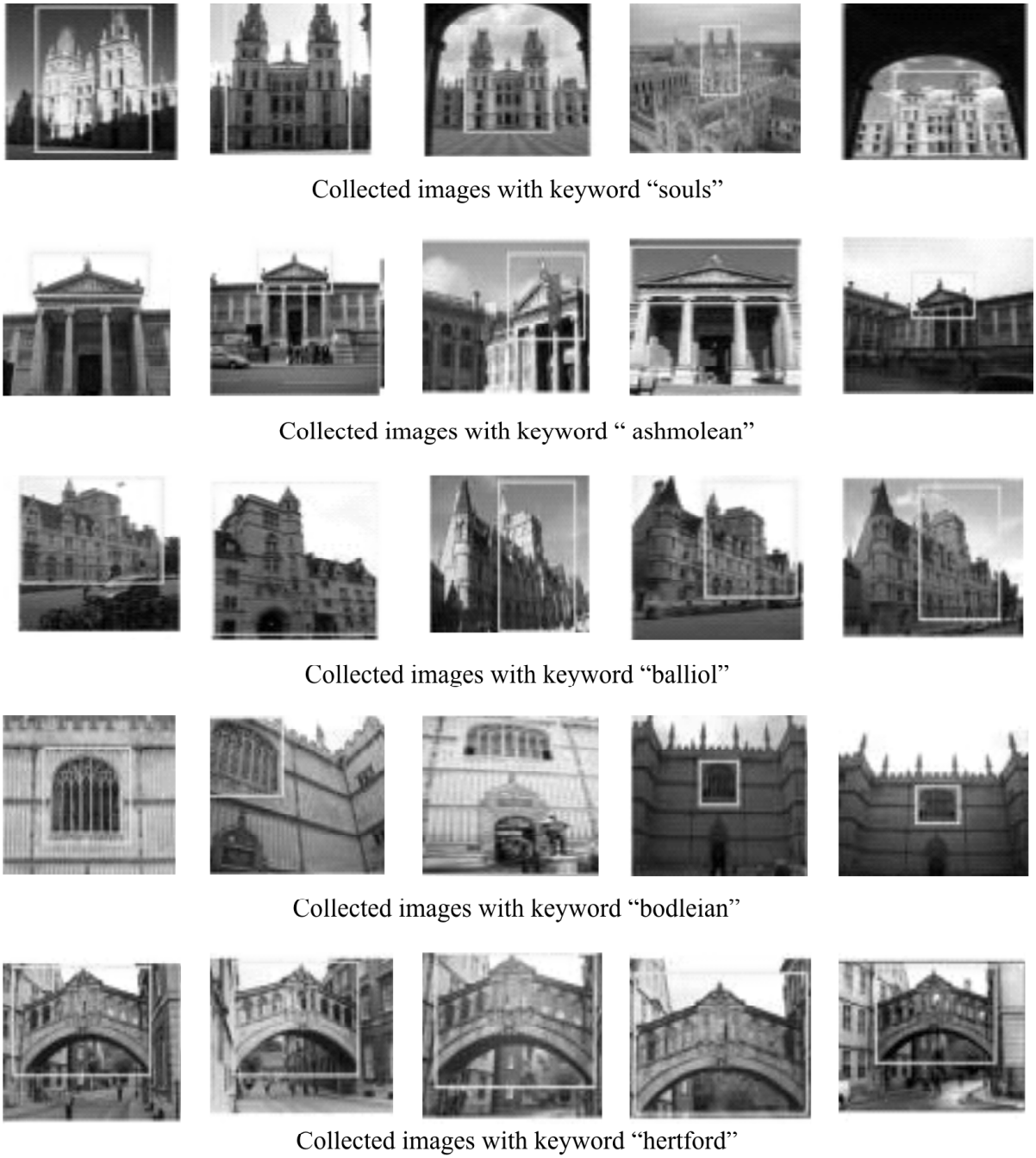

Figure 2: Illustration of the collected landmark dataset

To train discriminative bins selection for Edge-SIFT compression, use the Paris dataset [30] as a training set. This dataset contains 6412 images. Similar to the Oxford Building, the queries and corresponding ground truth are also available in the Pairs dataset. To test the performance of Edge-SIFT in large-scale image retrieval, collect a dataset containing 1 million images. During the retrieval process, the landmark images are adopted as queries, and are considered to evaluate the retrieval performance. Examples of the landmark dataset are illustrated in Fig. 2.

Parameter Selection: The initial Edge-SIFT and proposed enhanced Harris corner and Edge SIFT (HC-ESIFT) are related to the three parameters: r , which controls the size of the extracted image patch; D, which decides the size of the edge map; and w which controls the edge expansion. Test the effects of these parameters in image retrieval tasks

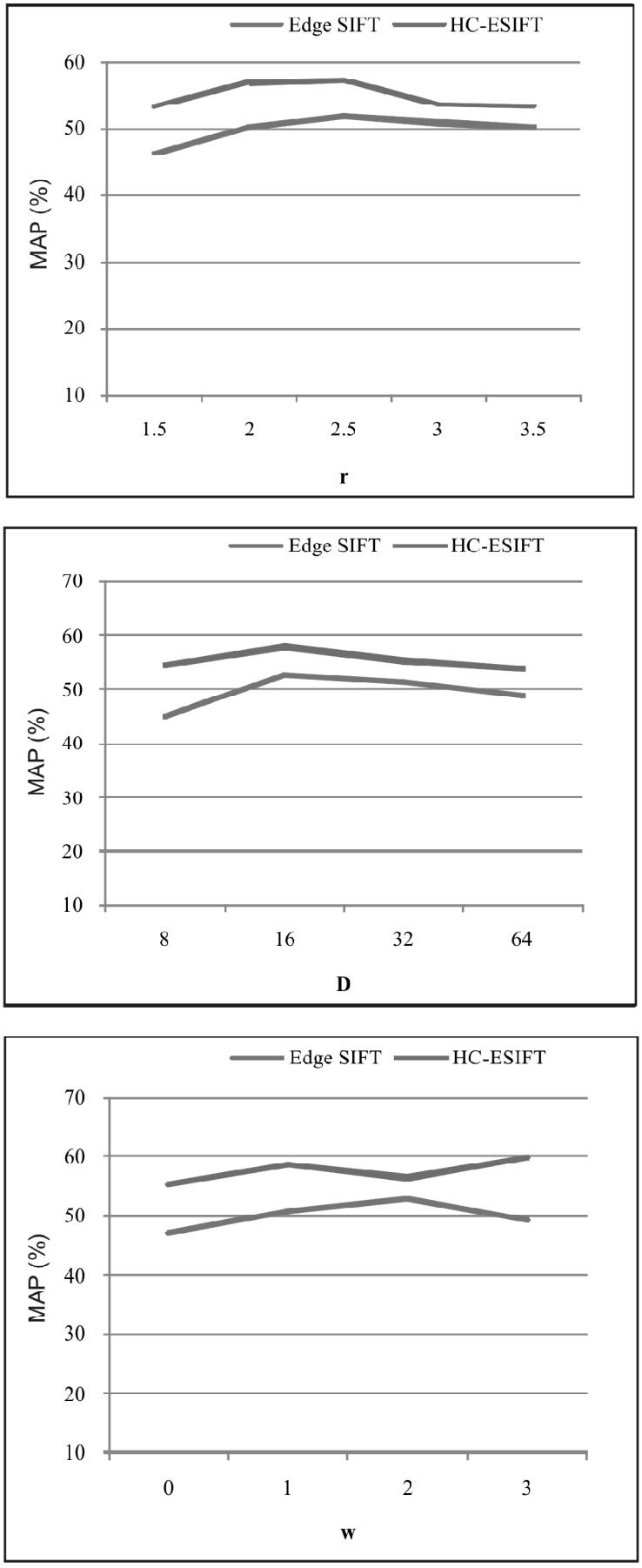

Figure 3: (a) (b) and (c) Illustration of the effects of r, D, and w

The effects of r betweenedge detection methods such as Edge SIFT and Edge SIFT (HC-ESIFT) are illustrated in Figure 3(a). From the figure, itis observes that larger r is helpful for performance improvement; this is because larger image patches contains richer edge clues, which make edge descriptor more discriminative. However, increasing r does not consistently improve the performancebetween edge detection methods such as Edge SIFT and Edge SIFT (HC-ESIFT). From the Figure 3(a) set the parameter r as 2.5, this shows reasonably good performance for HC-ESIFT method since the proposed work Harris Corner Point Detection is also performed to increase the efficiency.

The effects of D between edge detection methods such as Edge SIFT and Edge SIFT (HC-ESIFT) are illustrated in Figure 3(b). It is clear in the figure that, the retrieval performance degrades, if D is too large or too small. Intuitively, small D results in compact descriptor, however it doesn't loss any information of the imagepatches, since exact corner points also considered during edge detection process. From the Figure 3(b) set the parameter D as 16, this shows reasonably good performance for HC-ESIFT method.

The effects of w between edge detection methods such as Edge SIFT and Edge SIFT (HC-ESIFT) are illustrated in Figure 3(c). It can be observed that edge expansion is helpful to improve the performance. However, when w =3, the adjacent edges and their corner points were exactly detected by proposed method, which increase the accuracy of the system . Meanwhile, the validity of edge expansion is closely related to the edge map size. In the following experiments, for$16 \times 16 \times 4$ bit initial descriptor, set the value of w as 1; while for $32 \times 32 \times 4$ bit initial descriptor, set the value of 'w' as 2.

Validity of Edge-SIFT and HC-ESIFT Compression: After selecting the parameters, hence compress the initialEdge-SIFT and select the discriminative bins. Test twotypes of initial Edge-SIFT descriptors: a 1024 bit one whoser , D, and w are 2.5, 16, and 1, and another 4096 bit onewhose r , D, and w are 2.5, 32, and 2, respectively .First compress the two descriptors to 32 bit and 64 bit by selecting compact bins. In the Edge-SIFT descriptors compact bins are selected using Rankboost method and the proposed work HC-ESIFT compact bins are selected using kernel function for exact matching.

Clearly from the Figure 4, as more bins are optimally selected to the compressed descriptors, their retrieval performances are improved remarkably. This proves the validity of discriminative bins selection strategy using the kernel similarity function. The compressed HC-ESIFT from the 4096 bit descriptor

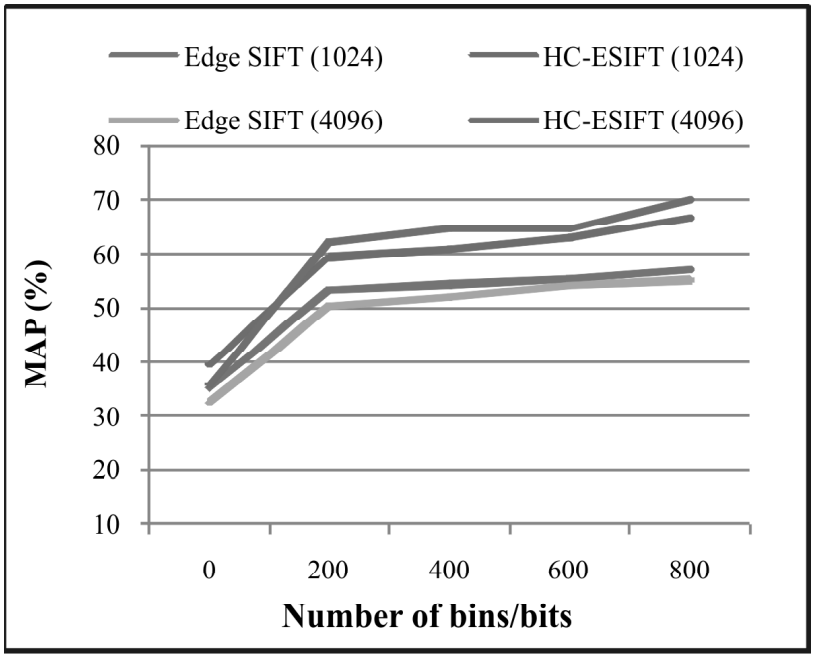

Figure 4: Illustration of the validity of the descriptor compression

finally outperforms the one from the 1024 bit descriptor. Since the proposed work optimal bins are selected using kernel function. This shows that larger descriptor contains richer clues, thus more discriminative bins can be selected. Using kernel function exact finds the matching between two different images. It can be also observed that, two compressed descriptors finally outperform their initial descriptors with more compact sizes.

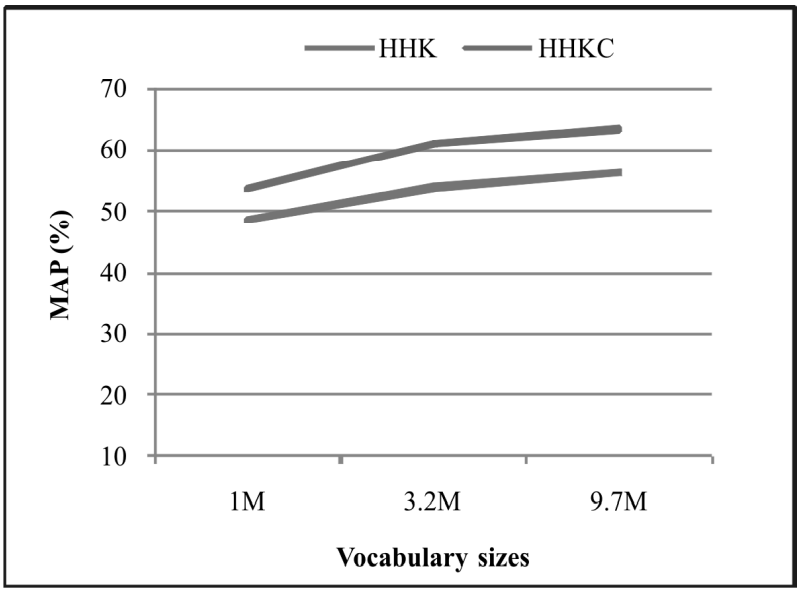

Figure 5: Comparisons of MAP and efficiency with different vocabulary sizes

Generate visual vocabularies by hierarchically clustering and hybrid hierarchical clustering methods to the descriptors which split the tree into 5-level vocabulary trees. Set the branch numbers as 16, 20, and 25, corresponding to three vocabulary sizes: 1M, 3.2M and 9.7 M, respectively. Comparisons among the 1024 bit SIFT descriptor are illustrated in Figure 5. For comparison is done between Hierarchical k center means (HCK) and Hybrid Hierarchical K-Centers (HHKC). It shows that the proposed HHKC produces high MAP to three vocabulary sizes. The proposed HHKC algorithm, uses the top-down (K-centers) and bottom-up (Unweighted Pair Group Method to measure the database image and the similarity queries with Arithmetic Mean (UPGMA) agglomerative hierarchical clustering algorithms for each clusters.

5. CONCLUSION AND FUTURE WORK

In this paper, propose a novel Harris corner points detection and edge based local descriptor called HC-ESIFT. This proposed HC-ESIFT is varied from traditional SIFT descriptor since it considers and updates four key modules, including feature extraction, feature quantization, image matching and retrieval.In feature extraction stage, Edge-SIFT are built upon edge maps of local image patch and keep both locations and orientations of edges. Then decompose the edge map into different sub-edge maps, according to the directions of edges. In order to make Edge-SIFT more robust and more compact, further study corner points detection and discriminative bins selection using kernel function. In feature quantization step, visual vocabulary tree can be generated through clustering with the defined similarity measurement. Here Hybrid Hierarchical K-Centers (HHKC) algorithm uses both the K-centers and Unweighted Pair Group Method with Arithmetic Mean (UPGMA) agglomerative hierarchical clustering to generate visual tree. To utilize Edge-SIFT in large-scale partial-duplicate mobile search, further propose an inverted file based indexing framework, which allows for flexible online verification. The indexing strategy is based on the standard inverted file indexing framework. Differently, each term in the index list contains extra verification code for online verification.Hence, conclude that, HC-ESIFT is compact,efficient, discriminative; retrieval system is

accurateand efficient for large-scale mobile partial-duplicate imageretrieval.Conducting experiments on mobile platforms requires lots of engineering implementation and optimization, which is beyond the scope of this paper. Consequently, use standard PC to simulate the mobile platform to compare Edge-SIFT with SIFT and ORB in the aspects of retrieval accuracy, efficiency, and data transmission. Because transporting these descriptors to the mobile platform does not change the computation or memory complexity, we could reasonable draw the conclusion that Edge-SIFT are superior to the other two descriptors in mobile visual search. However, it is still desirable to test Edge-SIFT on real mobile platforms.

REFERENCES

[1] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian, "Spatial coding for large scale partial-duplicate web image search.' in Proc. ACM Int. Conf. Multimedia, , pp. 511–520, 2010.

[2] J.He, J. Feng, X. Liu, T. Cheng, T. Lin, H. Chung, and S. Chang, "Mobile product search with bag of hash bits and boundary reranking.' in Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 3005–3012, 2012.

[3] Tian Q., S. Zhang, W. Zhou, R. Ji, B. Ni, and N. Sebe, "Building descriptive and discriminative visual codebook for large-scale image applications," Multimedia Tools Applicat. Vol. 51. No. 2. pp. 441–477, 2011.

[4] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in Proc. Eur. Conf. Computer Vision. pp. 404–417, 2006.

[5] W. Zhou, H. Li, Y. Lu, and Q. Tian, "Large scale image search with geometric coding.' in Proc. ACM Int. Conf. Multimedia, pp. 1349–1352, 2011.

[6] W. Zhou, H. Li, Y. Lu, and Q. Tian, "SIFT match verification by geometric coding for large-scale partial-duplicate web image search," ACM Trans. Multimedia Comput., Commun.,Applicat. (TOMCCAP).Vol. 9. No. 1.pp. 4, 2013.

[7] Y.Kuo, K. Chen, C. Chiang, and W. Hsu, "Query expansion for hashbased image object retrieval.' in Proc. ACM Int. Conf. Multimedia, pp. 65–74, 2009.

[8] D. Nister, and H. Stewenius, "Scalable recognition with a vocabulary tree.' in Proc. IEEE Conf. Computer Vision and Pattern Recognition. Vol. 2. pp. 2161–2168, 2006.

[9] D. G. Lowe, "Distinctive image features from scale invariant keypoints," Int. J. Comput. Vision. Vol. 60. No. 2. pp. 91–110, 2004.

[10] H.Jégou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large-scale image search.' in Proc. 10th Eur. Conf. Comput. Vis., Part 1, pp. 304–317, 2008.

[11] J.Yuan, Y. Wu, and M. Yang, "Discovery of collocation patterns: From visual words to visual phrases.' in Proc. IEEE Conf. Comput. Vis. Pattern Recognition, pp. 1–8, 2007.

[12] L.Paulevé, H. Jégou, and L. Amsaleg, "Locality sensitive hashing: A comparison of hash function types and querying mechanisms," Pattern Recognit. Letter. Vol. 31. No. 11. pp. 1348–1357, 2010.

[13] Y. Mu, J. Sun, T. Han, L. Cheong, and S. Yan, "Randomized locality sensitive vocabularies for bag-of-features model.' in Proc. 11th Eur. Conf. Comput. Vis. Conf. Comput. Vision, pp. 748–761, 2010.

[14] Z. Wu, Q. F. Ke, and J. Sun, "Bundling features for large-scale partial duplicate web image search.' in Proc. IEEE Conference. Computer Visison Pattern Recognition, pp. 25–32, 2009.

[15] S.Zhang, Q.Tian, K.Lu, Q.Huang, and W.Gao, "Edge-SIFT: Discriminative binary descriptor for scalable partial-duplicate mobile search," IEEE Transactions on Image Processing. Vol. 22, No. 7.pp. 2889-2902, 2013.

[16] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos.' in Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 1470–1477, 2003.

[17] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," Image Vision Computers.Vol. 22. No. 10, pp. 761–767, 2004.

[18] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," Interantional Journal of Computer Vision. Vol. 60. No. 1. pp. 63–86, 2004.

[19] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," Computer vision and image understanding. Vol.110. No.3. pp.346-359, 2008.

[20] J.Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching.' in Proceedings IEEE Conference Computer Vision Pattern Recognition, pp. 1–8, 2007.

[21] M.Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features.' in Proc. 11th Eur. Conf. Comput. Vis., pp. 778–792, 2010.

[22] X. Zhang, L. Zhang, and H. Shum, "QsRank: Query-sensitive hash code ranking for efficient -neighbor search.' in Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 2058–2065, 2012.

[23] Y. M. Luo, and R. Duraiswami, "Canny edge detection on NVIDIA CUDA.' IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 1-8, 2008.

[24] W. Yang, L.Dou, J.Zhang, J.Lu, "Automatic Moving Object Detection and Tracking in Video Sequences," SPIE Fifth International Symposium on Multispectral Image Processing and Pattern Recognition, pp.676-712, 2007.

[25] E. Vincent, and R.Laganiere, "Detecting and matching feature points," Journal of Visual Communication and Image Representation. Vol. 16. No. 1. pp.38-54, 2005.

[26] J.B. Ryu, C.G. Lee, and H. H. Park, "Formula for Harris corner detector", Electronics letters. Vol.47. No. 3. pp.180, 2011.

[27] Y. Song, I. V. McLoughlin, and L. R. Dai, "Local Coding Based Matching Kernel Method for Image Classification", PloS one. Vol.9. No.8. pp.103575, 2014.

[28] K. Murugesan, and J. Zhang, "Hybrid hierarchical clustering: an experimental analysis", University of Kentucky, Lexington, Technical Report: CMIDA-HiPSCCS, pp. 001-11, 2011.

[29] *http://www.robots.ox.ac.uk/~vgg/data/oxbuildings*

[30] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases.' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 1–8, 2008.