Towards Mobile Document Image Retrieval for Digital Library

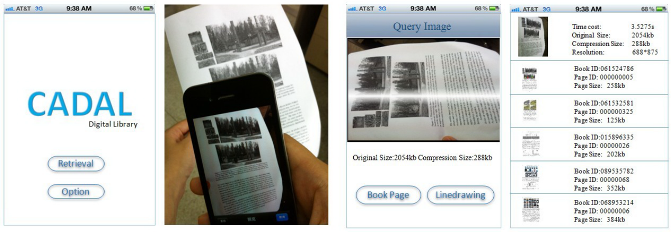
Ling-Yu Duan, *Member, IEEE*, Rongrong Ji, *Member, IEEE*, Zhang Chen, Tiejun Huang, *Member, IEEE*,
Wen Gao, *Fellow, IEEE*

***Abstract*—With the proliferation of mobile devices, recent years have witnessed an emerging potential to integrate mobile visual search techniques into digital library . Such a mobile application scenario in digital library has posed significant and unique challenges in document image search. The mobile photograph makes it tough to extract discriminative features from the landmark regions of documents, like line drawings, as well as text layouts. In addition, both search scalability and query delivery latency remain challenging issues in mobile document search. The former relies on an effective yet memory-light indexing structure to accomplish fast online search, while the latter puts a bit budget constraint of query images over the wireless link. In this paper, we propose a novel mobile document image retrieval framework, consisting of a robust Local Inner-distance Shape Context (LISC) descriptor of line drawings, a Hamming distance KD-Tree for scalable and memory-light document indexing, as well as a JBIG2 based query compression scheme, together with a Retinex based enhancement and an OTSU based binarization, to reduce the latency of delivering query while maintaining query quality in terms of search performance. We have extensively validated the key techniques in this framework by quantitative comparison to alternative approaches.**

***Index Terms*—Digital library, mobile visual search, line drawing retrieval, shape context, inner-distance, K-D tree, Hamming space, JBIG2 compression**

I. Introduction

In the past decades, digital library has played an important role in accessing the corpus of massive scanned documents stored in the digital image format. Content based retrieval could be a promising solution to facilitate pervasive and efficient access of the document images. In a typical scenario, a query is formulated as a photo that captures the visual objects of user interest, for example, a book cover, a document page, a figure, or even a line drawing. The visual query is sent to the server end, where the visually similar documents are matched and returned. To improve the image matching efficiency, the extracted visual signatures of database images have to be indexed, typically by an inverted indexing table. Comparing to typing query keywords, a snapped photo based query undoubtedly simplifies the input of a user query. Furthermore, in some specialized domains like searching ancient hieroglyph, content based queries retain as the effective approach.



Original Size:2054kb Compression Size:288kb

Time cost: 3.5275s
Original Size: 2054kb
Compression Size: 288kb
Resolution: 688*875

Book ID:061524786
Page ID: 000000005
Page Size: 258kb

Book ID:061532581
Page ID: 000000325
Page Size: 125kb

Book ID:015896335
Page ID: 00000026
Page Size: 202kb

Book ID:089535782
Page ID: 00000068
Page Size: 352kb

Book ID:068953214
Page ID: 00000006
Page Size: 384kb

Fig. 1. The exemplar scenario of mobile document image retrieval in digital library. The digital resources can be readily accessed by snapping a document image to search.

The recent proliferation of mobile devices has witnessed emerging approaches to improve the user experience of digital library browsing and search,, with various applications such as education, augmented reality, location search and product retrieval. User queries can be formed in a ubiquitous way, for instance, from the posters in a subway, or from the hieroglyph manuscripts in a museum. Figure 1 shows a typical scenario of mobile visual search in digital library.

The Problem. To deploy a mobile document image retrieval system in digital library, there are three challenges to be addressed from the perspective of mobile visual search. The first challenge comes from the photograph distortion of the embedded camera on a mobile device. Different from regularly scanned document images, the distortion of mobile captured images would seriously degenerate the performance of visual search. The landmark regions like line drawings are typically taken by mobile users to form a visual query. So a robust visual descriptor is required to characterize these line drawing regions. To retrieve document images, the relative layout of a shape or shapes, rather than their absolute scales and positions, plays an important role, as shown in Figure 2. Moreover, different from a simple contour shape, line drawings are characterized by much richer details (See Figure 2). Beyond existing shape descriptors, a new descriptor of line drawings has to be developed for characterizing the inner shape details.

The second challenge is on how to properly describe the textual regions in document images. In addition to line drawings, a significant proportion of document images is textual regions, which, in many cases, presents visually similar or even identical paragraph layouts and fonts. Based on existing local appearance descriptors, textual regions would produce fairly similar feature responses, thereby yielding high false

This work was supported by the Chinese Natural Science Foundation under Contract No. 61271311 and No. 61121002, and in part by the Research Fund of ZTE Corporation, and the CADAL Project Program.

Ling-Yu Duan, Rongrong Ji, Zhang Chen, Tiejun Huang and Wen Gao are with the Institute of Digital Media, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China. (e-mail: {lingyu,rrji,cjie,tjhuang,wgao}@pku.edu.cn). Rongrong Ji is also affiliated with the School of Information Science and Technology, Xiamen University. Rongrong Ji is the corresponding author.

Manuscript received on February 15th and accepted on October 15th, 2013.

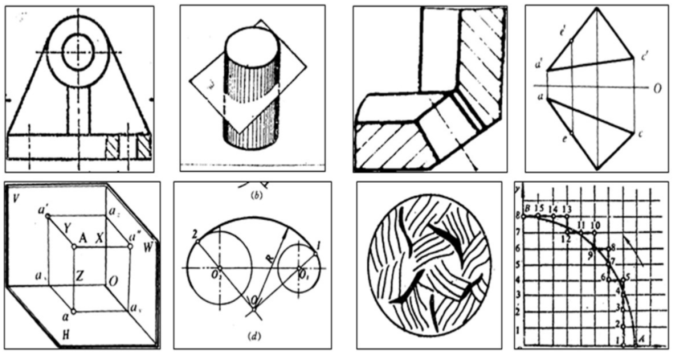


Fig. 2. Examples of line drawing in document images.

positive rate in retrieval. In addition, to improve the scalability, a memory-light descriptor indexing structure like visual vocabulary or hashing [30][31] [1] is important, which aims to remove or accelerate the time consuming online linear scanning process in retrieving document images.

Last but not least, to reduce the query delivery latency in mobile search scenarios is a challenging issue. Especially when wireless network connections are subject to bandwidth limit, a mobile query of small size is preferred. Different from existing works in extracting compact descriptors at the terminal for visual search [11][12][16], image compression of low complexity is more preferred because a variety of descriptors have to be extracted at the server end to fulfill document image retrieval. In particular, line drawings and textual regions retain sufficiently discriminative even after binarization, which may contribute to effective image compression.

The Proposed Framework. The framework includes three key components, as shown in Figure 3:

- Local Inner-distance Shape Context (LISC) to describe line drawings, which is robust against the mobile photographing distortion.
- Hamming Distance KD-Tree to seek the tradeoff between document retrieval performance and memory complexity of indexing structure in searching textual regions.
- JBIG2 based query image compression, together with a Retinex based image enhancement and an OTSU based binarization, to fulfill the low bit rate query.

In the following we present the key components in details.

Local Inner-distance Shape Context: A significant proportion of landmark figures in document images is in the form of line drawing, consisting of a group of lines with a smooth background. An effective descriptor are needed to represent the inner shape details of line drawings. However, existing shape descriptors [22][3][7][2][15], for instance the Edge Histogram Descriptor (EHD) [22], Shape Context Descriptor [2], and Stroke feature [15], produce quite a few weaknesses. First, the rotation invariance may not be met, for example, EHD [22] is not rotation invariant. Second, inner details cannot be properly captured by Shape Context [2], which are more suitable for describing the shape outlines. Third, the spatial Euclidean distances in [32][2] will significantly degenerate when mobile queries undergo the perspective distortions.

Hence we propose Local Inner-distance Shape Context (LISC) to describe line drawings robustly, especially for mobile search scenarios. LISC starts with detecting the interest points from the so-called maximal curves, upon which multiple shape context descriptors are produced at local maximal points. These shape context descriptors, together with their spatial layout, are then concatenated to form a LISC histogram.

Hamming Distance KD-Tree: To seek an optimal tradeoff between maximizing the search scalability and minimizing the descriptor quantization loss, a Hamming Embedding KD-Tree is proposed for indexing. First, instead of quantization based feature indexing such as hashing [36] or visual vocabularies [30][31], scalable feature matching is proposed based on the KD-Tree approximate nearest neighbor search, which is empirically shown to be suitable for textual regions. Second, distinct from the traditional KD-Tree, we propose to perform an orthogonal feature space transform, and quantize individual (orthogonal) feature dimension in one bit, in building up the KD-Tree. Accordingly, hamming signature is stored so that the memory cost of indexing structure is significantly reduced at the server end.

JBIG2 based Query Compression: Coming with the ever growing computation power of mobile devices, recent works have proposed to transfer computing workload from the server to the client. Typical works focus on directly extracting and delivering compact descriptors from the mobile terminal [11][12][16]. However, in document image retrieval, we propose an image compression and binarization scheme of low complexity to reduce the bit rate of query images. Most documents are scanned in a black and white image format. We may binarize a query image followed by image compression at a high compression rate; meanwhile, the quality of compressed queries would not degenerate much in terms of search performance. In details, the proposed scheme starts from a Retinex based image quality enhancement to filter out noises and remove the shadow. Then, an OTSU [33] binarization is employed. Finally, a JBIG2 [28] based binary image compression is deployed.

The rest of this paper is organized as follows: Section II reviews the related work. Section III introduces our line drawing descriptor LISC Section IV details our Hamming embedding KD-Tree based indexing and search. Section V introduces the image query compression scheme. We present our quantitative comparisons to the state-of-the-arts in Section VI and conclude this paper in Section VII.

II. Related Work

Digital Library. As a major trend in library operation and accessing, digital library is dedicated to collecting and managing massive documents in the digital format[2], which enables users to browse and search either locally or remotely. For instance, the World Digital Library (WDL) project[3] has collected huge amounts of books from handwritten, maps,

[1] While parallel indexing is a straightforward solution, given a database scale, an essential task is to minimize the memory complexity as much as possible for the optimized usage of computing resources.

[2] http://en.wikipedia.org/wiki/Digital_library

[3] www.wdl.org/

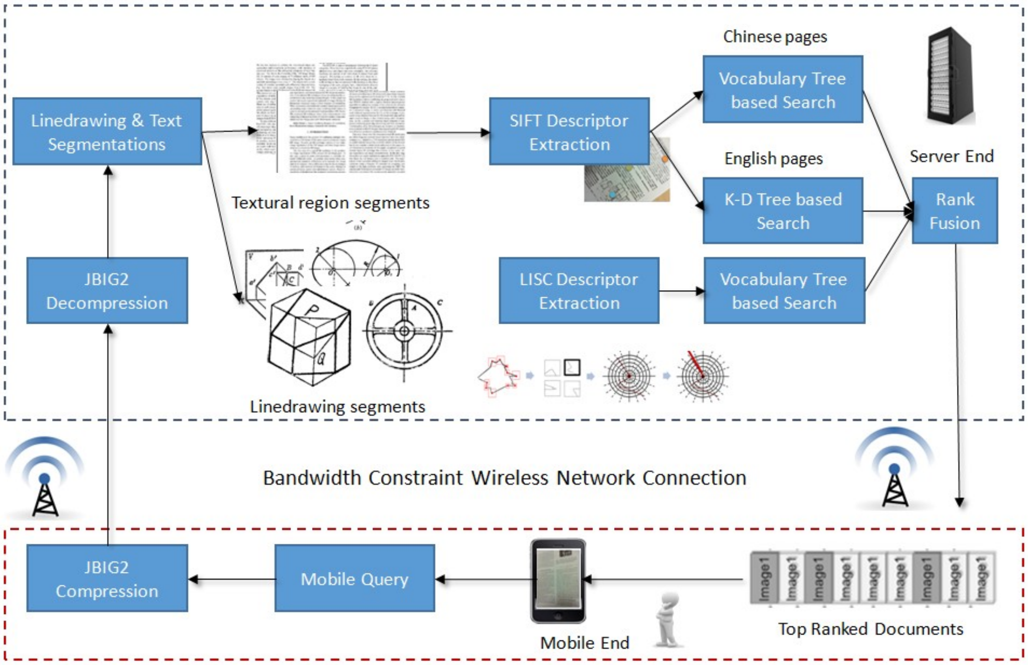


Fig. 3. The proposed mobile document retrieval framework.

printed document, newspapers etc. worldwide. The China-US Million Book Digital Library (CADAL) project[4] has covered over 400 million scanned books. The Google Books project[5] aims to search scanned books, which is linked with charged services for easy browsing worldwide. With its ever growing large scale of documents, one key technical challenge lies in dealing the scalability issue, i. e., how to develop robust techniques to efficiently and effectively. In particular, mobile document retrieval is becoming an emerging topic to broaden the functionality and reachability of digital library.

Document Image Retrieval. Since 1990s, the problem of document image retrieval has been widely studied, due to a wide variety of applications in digital library and beyond. The main target is to find the exact or similar documents by querying a printed or scanned document image over a large document corpus. Previous works typically rely on Optical Character Recognition (OCR) techniques to tackle the problem of document retrieval. More recently, visual matching is becoming a promising alternative to solve the limitation of poor OCR performance in scanned documents retrieval. For example, Mao et al. [29] proposed a paragraph structure analysis approach, which attempts to segment a document image into paragraphs, and then search the line drawings and textual regions, respectively. Nakai et al. [25] proposed to characterize visual statistics of the word density, in which the centroid of each word to the other words ia assumed to follow an affine invariant cross-ratio. With the development of visual search algorithms, local descriptors plus bag-of-features models, such as SIFT plus Visual Vocabulary [31][30], are supposed to benefit document image search in a sense of generic use.

However, none of existing techniques is dedicated to sort out the challenging issues in the emerging task of mobile document image retrieval. One one hand, an image query from a mobile camera is negatively effected by photograph distortion more or less, such as rotations, lighting changes, and de-focusing, etc. Such kinds of distortion have posed a big challenge from the viewpoint of state-of-the-art OCR techniques. On the other hand, existing visual descriptors in terms of words and paragraph layouts, basically work well over pictographs like Chinese or Japanese documents, which are, unfortunately, much less effective for the documents in Latino languages such as English or French. In particular, for the latter, the visual characteristics of words and paragraph layouts are quite similar [25], which is less likely to produce patterns from layout or other global characteristics. Furthermore, the local descriptors like SIFT or CHOG cannot work well for document image retrieval in a fine granularity.

Shape Description. Shape description is a long standing research topic in computer vision. One representative work is the Shape Context descriptor proposed by Belongie et al. [2]. Shape Context attempts to model the shape characteristics by calculating the shape related pixel distribution in a polar coordinate system, in which the distribution, embedded with context, has shown advantages in improving the robustness of shape descriptors. Beyond the delineated line shape, some descriptors like Stroke [15] work on a set of different types of lines such as curves and right angles, etc, aiming to describe the spatial layout of these lines to represent a shape. The other group of methods focus on yielding a descriptor in a "generative" manner. For example, works in [4][5] proposed to build up a part based shape model based on a set of training images. Alternatively, works in [1][2][6][7] proposed to model the shape structures generatively from one image only. For

[4] archive.org/details/cadal/
[5] books.google.com/

generative shape models, the shape statistics of each image are modeled separatively, and the matching of image pairs is computed over the corresponding model components. In 3D shape retrieval, works in [3][22] proposed to transform a 3D drawing to 2D space,and then employ the 2D shape histogram [10] or Edge Histogram Descriptor (EHD) [22] to represent 3D shape based on the gradient histogram distribution of shapes. To deal with the scalability issue, recent work in [32] further proposed a new shape descriptor based on the oriented chamfer matching (OCM) distance.

In mobile scenarios, the above mentioned works cannot be well applied to search landmark regions in document images, e.g., the line drawings, due to more or less distortion caused by mobile photographing. For instance, the EHD approach [22] cannot handle the rotation invariance. Shape Context is suitable for describing the outline of figures, rather than the inner details, which would probably yield ambiguous histogram representations for the figures with similar outlines but different inner appearances. Especially, in previous works the absolute coordinates of shape points are not invariant with respect to the perspective transform, so that identical or similar resolution settings between two line drawings are necessary for exact matching, which, unfortunately, cannot be guaranteed in mobile visual queries. Instead, the proposed LISC descriptor comes up with a sort of local inner-distance [3] to extend the shape context [2], which is in spirit motivated by [8] that combines global and local features to characterize connected points.

Mobile Visual Search. More recently, much more exciting research and applications have been pursued in mobile visual search. For example, Zhang et al. [17] adopted sequential matching of more than one reference views to estimate the pose and motion direction for better location recognition. Schindler et al. [18] presented an approach to large-scale location recognition based on geo-tagged video streams, which works on multi-path search over the vocabulary tree. Eade et al. [19] adopted a vocabulary tree based approach to real-time loop closing. Yeh et al. [20] proposed a hybrid color histogram to compensate its original ranking results in location recognition by using mobile devices. Furthermore, Crandall et al. [21] presented a system to identify landmark buildings based on image data, meta data and other photos taken within a 15-minute time window.

Compact Descriptors for Visual Search. With the fast growing computation power of mobile devices, recent work [11][12][16] proposed to directly extract compact descriptors at the terminal, and send the compact descriptors instead of the query images towards low bit rate visual search. However, most local descriptors reported in literatures like SIFT [13], SURF [14] or PCA-SIFT [15] are "over size", e.g. sending hundreds of such descriptors typically costs more data throughput comparing to sending the original image directly. One pioneer work goes to the Compressed Histogram of Gradients (CHoG) proposed by Chandrasekhar et al. [12]. CHoG adopts Haffman tree coding to compress the initial descriptor of gradient histograms into approximately 60 bits. Suppose that $\sim 1,000$ interest points are detected per image, the overall feature data is approximately 8KB, which has been less than the query image size (typically over 20KB). Recent papers in [11][16] stepped forward to compress the quantized bag-of-words histogram instead of compressing the local descriptors.

III. Towards Line Drawing Descriptors

A. Overview

A dominant proportion of figures in document images are in the form of line drawings, i.e., a group of lines located at a smooth background. For document search, we propose to deal with textual regions and line drawing regions separately. A Gabor transform is applied to the document image. The low frequency part is classified as the line drawing region, while the high frequency part as the textual region. We need to solve two problems in describing the figures of line drawings:

- The texture of line drawings is sparse, which would degenerate the performance of local descriptors like SIFT.
- Shape descriptors like Shape Context [2] only focus on the object outline but ignore the internal texture. However, internal texture or line details are important to describe the object of line drawings.

Taking Shape Context for instance,

$$h_i(k) = \sharp\{\hat{x_j} \neq \hat{x_i} : \ (\hat{x_j} - \hat{x_i}) \in bin(k)\}, \tag{1}$$

in which h_i denote the Shape Context descriptor at a reference point x_i, $\hat{x_i}$ the center of the polar coordinate system. The polar coordinate system is subdivided into 60 regions based on the quantization of angle as well as the distance scales, with $k = 1, 2, ..., 60$. For line drawings, since the lines are densely distributed in each region, the 60-bin histogram are very likely to be uniformly distributed, which would be less discriminative[6] based on the statistics of *every* lines.

Hence we propose to identify the most discriminative points *inside* the line drawing, and then aggregate the feature statistics from these discriminative ones. Such a selective treatment behaves just like the functional module of feature selection to generate compact descriptors for visual search, aiming to produce more discriminative histogram response. The proposed LISC descriptor starts with detecting interest points within a line drawing, and then generates their Shape Context histograms using the local inner-distances for each interest point, respectively. Such histograms of interest points are finally aggregated to form an averaged histogram as the shape descriptor, as shown in Figure 4. In summary, the proposed LISC descriptor has improved the original shape context descriptor in two aspects:

- LISC investigates the features of interest points inside a line drawing, rather than every shape point. So the uniform distribution from dense feature responses from traditional Shape Context [2] can be alleviated significantly.
- LISC employs a polar coordinate system with a novel inner-distance along the lines, which is robust against

[6] Note that such a distribution of dense features is with other shape descriptors like EHD [22].

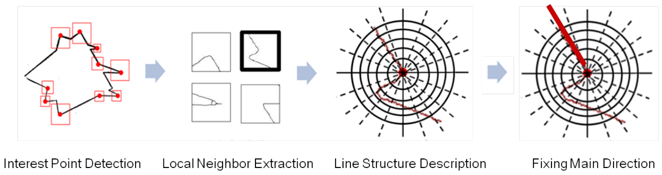


Fig. 4. The pipeline of the Local Inner-Distance Shape Context descriptor. Note that detectors are operated in the same scale.

a variety of photographing variances, comparing to the Euclidean distance.

In dealing with mobile document image search, LISC instead of Shape Context brings about two main advantages: (1) Local features are suitable to capture the "salient" shape details. The negative effects from background or occlusions can be alleviated, which may greatly contribute to the descriptor robustness. (2) The adopted inner distance helps reduce the negative effects of perspective transform as well.

B. LISC Descriptor Extraction

Interest Point Detection. Given a line drawing, we attempt to detect the curve corners, in which the points of maximal response values are used to locate interest points, The underlying assumption is that the discriminative features of a line drawing basically come from the corners in the curve structure. Interest point detection is to allocate the contribution of meaningful shape statistics to the corners only.

There exist many corner detectors in literatures, such as SUSAN, COP, Harris and CSS [9]. We adopt the CSS [9] detector, because of its computational efficiency. CSS detects the local maxima curvature to extract the initial corner candidates, which are subsequently verified by using angle tangent with adaptive thresholds.

For each detected corner, its square neighbor region with length λ is chopped out, from which we form the part based primitives of the line drawing. In our implementation, λ is empirically set to 100.

Given an interest point, the Shape Context descriptor [2] is extracted as $I = \{F_1, F_2, ..., F_N\}$, where F_i is the i-th interest points. Only the shape features from interest points are aggregated [7].

Choosing Initial Descriptors. Comparing to the global shape descriptors [4][5], the "local" shape descriptor renders the Shape Context features more discriminative, since the lines within a local neighborhood are sparse, thereby avoiding the uniform distribution of dense features. As shown in Figure 4, interest points (red points) are detected from the corner detector, the neighborhood sub-figures (red rectangles) are isolated to extract the shape context feature. Since different line drawings may exhibit various line constitutions, the line structure is supposed to be reflected in local neighborhoods. Similar line drawings produce similar line structure of local neighborhoods, and vice versa.

Inner-Distance. In the original Shape Context descriptor, the distance from a reference point to other anchor points is measured by the Euclidean distance, which cannot suffice for non-convex polygons, as demonstrated in [3]. Thus "inner-distance" is adopted to replace the Euclidean distance. The inner-distance is defined as the shortest path along the lines to measure the distance between two interest points. Comparing to [3] working on direct inner-distance, our proposed drawing-based inner-distance is more like the "earth mover distance" based on the patches along the lines. The absolute distance between two points along the line would change in the Euclidean space, whereas their relative distance of sub-segments along the line retains almost invariant. Therefore, the improved inner-distance is more robust against the photographing variance with respect to the perspective.

Given a line drawing D, the drawing-based inner-distance can be calculated as:

$$\forall x, y \in D \quad d_{drawing}(x, y; D) = |\gamma(x, y; D)|, \tag{2}$$

where $\gamma(x, y; D)$ is the shortest path from x to y along the curves in D. It consists of two steps:

- First, let G denote a graph built with the points from a part P of the line drawing, $V(G) = \{v_i \in P | i = 1, 2, \ldots, N\}$ the set of vertices of G where N is the number of points in P, $E(G) = \{e_{ij} | v_i\}$ the adjacent edges with v_i in P, where $i, j = \{1, 2, \ldots, N\}$, and $dist$ a matrix that stores the inner distance between each pair of points in $V(G)$. Initially, we have

$$dist[i][j] = \begin{cases} EuclideanDis(v_i, v_j), & e_{ij} \in E(G) \\ \infty, & \text{otherwise.} \end{cases} \tag{3}$$

- Then, Floyd-Warshall shortest path algorithm is applied to G to update $dist$ with the time complexity $O(N^3)$ [8]. Finally, we get the drawing-based inner-distance from x_i to x_j as:

$$d_{drawing}(x_i, y_j; D) = dist[x][y]. \tag{4}$$

where $i = 1....N$ and $j = 1...M$.

Explanation. Given an affine transform (a sort of variation), each individual distance between any two points on the line might change. However, since we propose to find out the shortest path between two points along a line, the shortest path more likely retains the same. For instance, if a line drawing is stretched, the relative distance ratios of these shortest pathes along a line are supposed to remain unchanged. With the improved "earth mover distance" inner-distance, the proposed LISC descriptor contribute to more robustness in matching line drawings between a mobile query image and the reference document images.

Descriptor Extraction. Considering the vectors from a reference point as origin to all the other points on the shape, the Shape Context at the reference point is defined as the feature histogram of the other points in the relative polar coordinates, i.e.,

$$h_i(k) = \#\{x_j \neq x_i : (x_j - x_i) \in bin(k)\}, \tag{5}$$

[7] Such a selective treatment can improve the descriptor, since more repetitive points are more robust against mobile photograph variances.

[8] Other shortest path algorithms can be employed to replace the Floyd-Warshall Algorithm, and better performance or efficiency may be expected

where the bins uniformly divide the log-polar space. x_i denote the reference point, x_j any other point on the shape, and $x_j - x_i$ the vector from x_i to x_j

In the original Shape Context, every point in the contour is considered as a reference point to describe the shape, while the proposed LISC descriptor selects the detected corners only as reference points to generate the shape descriptor.

Moreover, the Euclidean distance in the original shape context [2] is replaced with the inner-distance. Therefore, the LISC descriptor of a neighborhood part centered at x_i can be defined as a histogram of relative polar coordinates of all other points x_j:

$$h_i'(k) = \#\{x_j \neq x_i : (x_j' - x_i') \in bin(k)\}, \tag{6}$$

where x_i is the reference point, and x_j in a line drawing D is mapped to x_j' in the log-polar coordinate system, and $j = 1, 2, \ldots, N$. Let $x_j' = (\rho, \theta)$, let $\rho = |x_j'| = \log d_{drawing}(x_i, x_j; D)$ be the relative distance. The relative angle θ is defined as:

$$\theta = \arctan(\frac{\Delta Y}{\Delta X}), \tag{7}$$

where ΔX and ΔY denote the differences of the projections of x_i and x_j over x and y axis, respectively.

C. LISC Descriptor Matching

Since the LISC descriptor is represented as a distribution histogram, x^2 statistics test is a straightforward way to do matching between LISC descriptors:

$$C_{ij} \equiv C(x_i, x_j) = \frac{1}{2}\sum_{k=1}^{K} \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)}, \tag{8}$$

where $C(x_i, x_j)$ denotes the matching cost of two points, $h_i(k)$ and $h_j(k)$ the K-bin normalized histogram at x_i and x_j, respectively.

Given the set of matching costs C_{ij} between all pairs of corners, x_i from the first line drawing and y_j from the second line drawing, following [7], the matching process is to minimize $\sum_i C(x_i, y_{\pi(i)})$, where π is a permutation.

Speedup: Over a large scale document corpus, the LISC descriptor matching procedure can be speedup by using the Bag-of-Features model [31]. The LISC descriptors of line drawings are quantized and indexed by an inverted file. Thus, very fast similarity computing between pairs of line drawings can be achieved by *tf-idf*.

D. Rotation and Scale Invariance

Scale Invariance. To fulfill scale invariance, we further propose an adaptive division of subregions, in addition to the assignment of principle orientation:

- First, to assign the main orientation, we project the line pixels in a local neighborhood to the polar coordinate space, and then generate the distribution with respect to the bins of different angles. The angle with the maximal bin value is accordingly selected as the main orientation to rotate the region for normalized description.
- Second, to fulfill adaptive region division, we calculate the averaged distance between all pairs of interest points in a line drawing, then normalize the distances to $[0, 1]$, which ensures that the inner distances of query and reference line drawings are with a uniform scale.

Rotation Invariance. We further present rotation invariance by introducing the relative angle. As proposed by Belongie et al. [2], one choice is to rotate the coordinate system at each reference point so that the positive x-axis is aligned with the tangent vector. However, this method does not work for LISC, since the detected corners are usually located at the intersection of a few curves, and there is a lack of well-defined tangents.

To solve this problem, we propose to find out the top two nearest corners x_j and x_k where $i \neq j \neq k$, for each corner $x_i, i = 1, 2, \ldots, N$. Let l_{ij} be the line linking x_i and x_j, and l_{ik} linking x_i and x_k, respectively. x_i is then the intersection of l_{ij} and l_{ik}. We denote the angle between l_{ij} and l_{ik} by α. Then, when we deal with the bisector of α as the y-axis of the coordinate space centered at corner x_i, the content of a neighborhood part at corner x_i is supposed to be rotation invariant.

IV. Towards Scalable Search

Motivation. To accomplish effective and efficient retrieval, we resort to the K-D Tree based approximate nearest neighborhood search, with a Hamming distance embedding scheme [23] to reduce memory cost [9]. The motivation is to introduce a compact binary code to reduce the memory cost from storing original local descriptors for backtracking. In the proposed Hamming Distance (HD) KD-Tree, we replace the Euclidean feature space with a Hamming space. The Hamming distance KD-Tree enables very fast similarity matching, while maintaining matching performance. Below we give the formulation in details.

Offline Learning. The HD KD-Tree structure is learnt offline. To generate the Hamming signature of LISC or SIFT features, the feature space reduction, followed by feature space binarizing, is applied, which involves four basic stages:

- *Generating Random Orthogonal Matrix*: Following [37][38], we first perform a QR decomposition on a random Gaussian matrix for the subsequent feature transform. From the resulting matrix, the top d_b rows are formed as a $d_b \times d$ orthogonal projection matrix P. In our implementation, we set $d_b = 64$ and $d = 128$ for SIFT, as well as $d_b = 32$ and $d = 60$ for Shape Context.
- *Projecting Features*: The matrix P is used to transform each individual feature x_i in KD-Tree leaves, yielding the projection feature vector $z_i = Px_i = (z_{i1}, z_{i2}, ..., z_{id_b})$.
- *Representing Leaf Nodes*: For each leaf l, we calculate the mean vector $\phi_{i,h}$ of features $\{z_{ih} | q(x_i) = l\}$ over the projection vectors $h = 1, 2, ..., d_b$, where $q(x_i) = l$ denotes the feature x_i belonging to l.

[9] One alternative is to reduce the descriptor number by data sampling or to apply vector quantization to raw descriptors , which would suffer from degenerated performance seriously.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2013.2293063, IEEE Transactions on Multimedia

Algorithm 1: HD KD-Tree based document image search.

```
Input: Query image $q$, descriptors $\{x_i\}(q)$, max recursive number $n_{max}$.
Output: The similar images in the reference set.
Initialize a ranking image list $I$.
for each $x_q$ in $\{x_i\}(q)$ do
    recursive number $n_r = 0$
    while $n_r < n_{max}$ do
        Initialize the minimal heap $h$.
        Walking through the KD-Tree over $x_q$ to find out the nearest leaf node.
        Generate the Hamming signature $b(x_q)$.
        Find out the nearest feature points $x_i$ to $b(x_q)$, store $h\big(b(x_q), b(x_i)\big)$ in $h$.
        $n_r + +$
    end
    Find out the most similar point in $h$ to $x_q$, and put its containing images into $I$.
end
Pick out the top ranking images in $I$ as the returning.
```

- *Generating Hamming Signature*: For the feature set z_i in each leaf node l, we generate the Hamming signature $b(z_i) = \big(b_1(z_i), b_2(z_i), ..., b_{d_b}(z_i)\big)$ as

$$b_h(z_i) = \begin{cases} 1 & \textit{if } z_{ih} > \Phi_{ih}, \\ 0 & \textit{otherwise}. \end{cases} \tag{9}$$

Online Search. In online search, for every local descriptor extracted from a given document image involving line drawing regions or textual regions, or both, the descriptor is transformed from the original feature space, i.e. the 128 dimensional SIFT feature space or the Shape Context feature space, to the Hamming space, with dimension d. Subsequently, KD-Tree based search is performed to retrieve the most similar descriptors in the Hamming space. The online search process is outlined in Algorithm 1.

It is noted that, the rank fusion is carried out via merging the ranking lists from querying both text regions and landmark linedrawings. Given two ranking lists with weighted scores to indicate the ranks of respective documents, we directly fuse the weights from both textual regions and landmark linedrawings. Subsequently, we re-rank all candidate documents based on their merged weights.

Discussion. The reasons why we incorporate Hamming distance computing into the K-D Tree search are two folds. First, as shown in previous study as well as the experiments in Tables 2 and 3, SVT is actually affected by feature quantization errors. However, such effects can be well alleviated by the K-D Tree based search, due to the approximated nearest neighborhood search of descriptors with the functionality of back tracking, which avoids the phase of quantizing descriptors to code words. Second, while Hamming distance works on the binarization of feature space, much less quantization errors are incurred comparing to vocabulary based quantization methods, thanks to the backtracking functionality of K-D Tree.

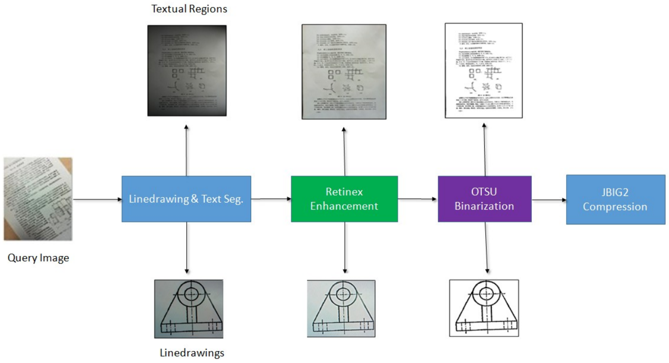


Fig. 5. Query image enhancement prior to binarization.

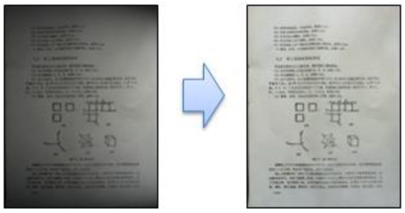


Fig. 6. Examples of Retinex based image enhancement.

V. Towards Low Bit Rate Query

In this section, we further discuss the query compression scheme towards low bit rate mobile visual search in the field of document images. Rather than extracting compact descriptors as in [11][12][16], we propose to compress query images in JBIG2 at low complexity.

Before JBIG2 compression, image enhancement is carried out, followed by binarization, which aims to remove the photographing variations from shadow, illumination, etc. The effects of enhancement and binarization has been shown in Figure 5. The implementation have been deployed to the mobile end.

Image Enhancement. We adopt a Retinex [24] based image enhancement to remove the shadow effects. As pointed out in [24], the imaging mechanism of human eyes is determined by two main factors, e.g., the incidence light and the object reflection. Both factors are formulated in the Retinex theory as:

$$S(x, y) = R(x, y)L(x, y), \tag{10}$$

where $L(x, y)$ denotes the incidence light, and $R(x, y)$ the object reflection, (x, y) denotes the pixel position. The enhancement is to recover the reflection $R(x, y)$ from an image $S(x, y)$ by using a Gaussian kernel to compute L from S, such that $L(x, y) = S(x, y)G(x, y)$, where $G(x, y)$ is the Gaussian kernel. This has derived the following operation:

$$\begin{aligned} log\big(R(x, y)\big) &= log\Big(\frac{S(x, y)}{L(x, y)}\Big) \\ &= log\big(S(x, y)\big) - log\big(S(x, y)G(x, y)\big). \end{aligned} \tag{11}$$

Figure 6 shows the effects of Retinex based enhancement.

Binarization. We adopt the OTSU [33] technique to bina-

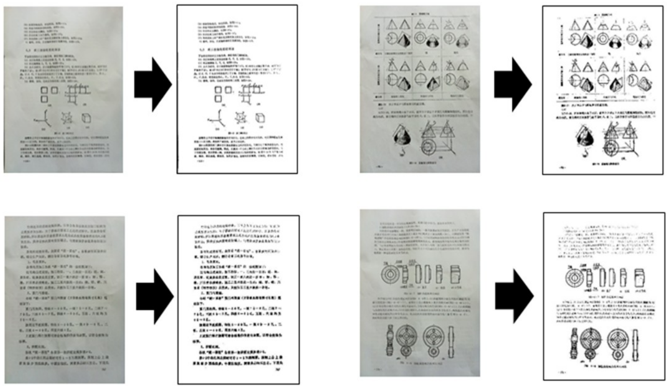


Fig. 7. Examples of OTSU based binarization.

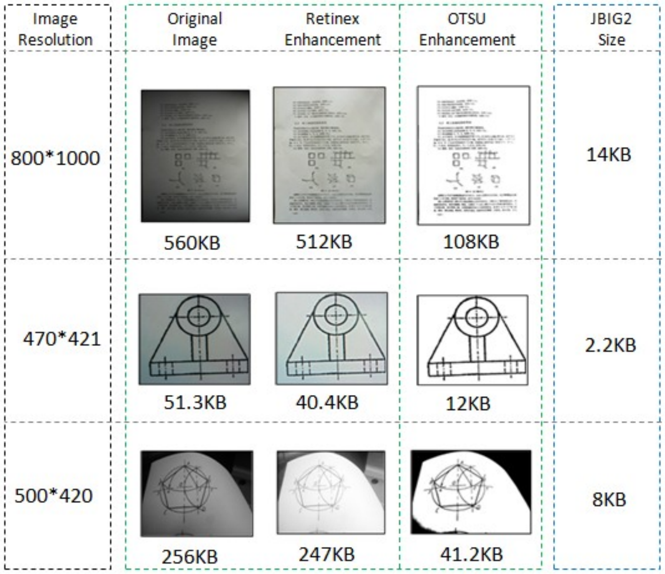


Fig. 8. Examples of JBIG2 based compression including Retinex and OTSU effects, as well as compression rates.

rize the query image after enhancement [10]. Figure 7 shows several examples of OTSU based binarization.

JBIG2 Image Compression. JBIG2 [28] is a standard approach to compress a binarized image. JBIG2 first tries to segment a document image into three types of regions, namely, text regions, halftone regions, and regular regions. For text regions, the symbol compression is used. For halftone regions, the grid coding is applied. And the arithmetic coding is applied to regular regions. Figure 8 shows the JBIG2 based compression effects of the entire document pages, as well as the closeup of specific line drawing regions.

VI. Experimental Validation

A. On LISC Descriptor

We first report the quantitative performance of LISC based line drawing descriptors, with comparisons to the alternatives or state-of-the-arts in document image retrieval benchmarks.

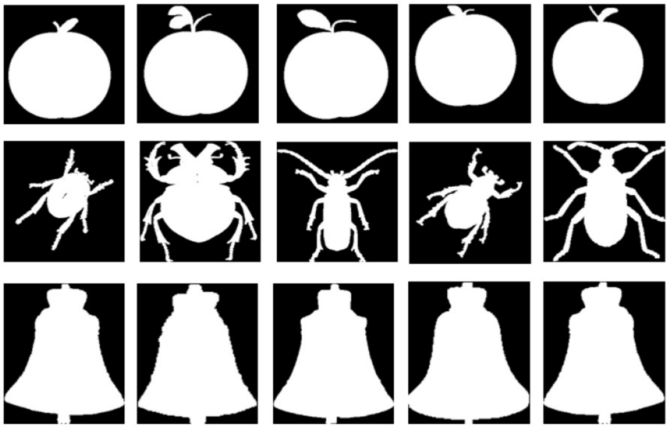


Fig. 9. Examples of the *MPEG-7 CE Shape 1 Part B* benchmark.

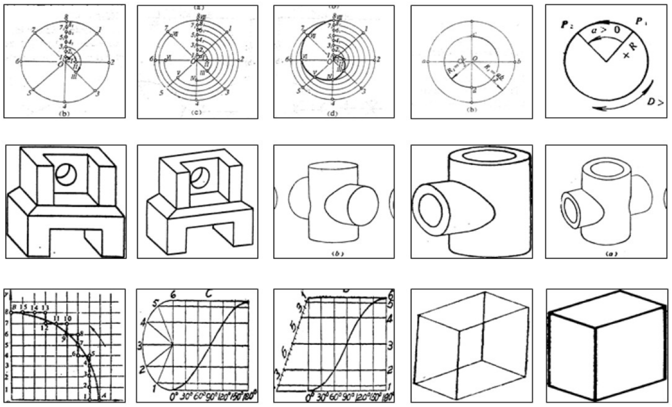


Fig. 10. Examples of the *CADAL Linedrawing* benchmark.

Benchmarks and Evaluation Protocol. The Precision-Recall curve is used as the evaluation protocol. We quantize the performance over two benchmarks with comparisons to the state-of-the-arts.

- *MPEG-7 CE Shape 1 Part B [35]:* This benchmark contains over 1,400 line drawings. Most of line drawings produce the contour like shape. There are in total 70 categories, each of which contains 20 images captured from different perspectives. Figure 9 shows some examples. Over this benchmark, 500 images are selected to generate visual queries while the rest are used as the reference images.
- *CADAL Line drawing:* This benchmark contains 13,000 line drawings collected from the scanned book pages in the CADAL project. Unlike *MPEG-7 CE Shape 1 Part B*, this benchmark provides a real-world line drawing collection, in which each line drawing contains rich inside details. Figure 10 shows some examples of CADAL line drawings. Over this benchmark, we select 883 images to form visual queries, each of which has 2 to 15 reference images. To further validate the scale and rotation invariance of LISC descriptors, for each query, we generate two more query images by random image rotation or scaling. Finally, we have generated a query set of in total 2,649 images.

Baselines and Parameter Settings. We compare the pro-

[10] Nevertheless, other techniques like Niblack [34] can be employed as well, since the shadows and noises have been removed by enhancement.

TABLE I
THE MAP WITH RADIUS BIN NUMBER = 12 AND DIAMETER BIN NUMBER = 2, 5, 7.

Diameter Bin	2	5	7
mAP	0.81	0.83	0.78

TABLE II
THE MAP WITH DIAMETER BIN NUMBER = 5 AND RADIUS BIN NUMBER = 8, 12 ,24.

rADIUS Bin	8	12	24
mAP	0.77	0.83	0.72

posed LISC descriptor with SIFT [13], Shape Context [2], Edgel [32], and Stroke [15] in both benchmarks. Following the parameter settings of Shape Context [2], we subdivide the radius of each regional circle into 5, and the angle into 12, thereby yielding 60 dimensions in total. For the Edgel feature, we set the tolerance rate as 20 pixels, the same setting as [32].

Quantitative Performance on the *MPEG 7 CE Shape-1 Part B*. Figure 11 shows the Precision-Recall curves on the *MPEG 7 CE Shape-1 Part B* benchmark with comparisons to [2][13][15][32]. Both Shape Context and LISC perform better. For the shape simplicity over this benchmark, Shape Context suffices for the description of outline like shapes. However, Edgel yields worse performance, due to the lack of invariance regarding to rotation and scale, which in turn commonly exist in the *MPEG 7 CE Shape-1 Part B* benchmark. The SIFT descriptor, originally designed to describe local interest points, is poor in shape representation.

Regarding the parameters setting of Shape Context which we apply to generate the LISC, we have cross validated the performance of shape context with different settings in the radius division and the angle division as below: (1) fix $\alpha = 12$, with radius beta division from 3×12, 5×12, 7×12, to 9×12. (2) fix $\beta = 5$, with α degree division from 5×8, 5×12, to 5×24. With different parameter settings, we generate the evaluation results listed in Table I and Table II, from which we finally select $\alpha = 12$ and $\beta = 5$ as the shape context setting to generate the LISC descriptors.

Figure 12 shows mobile retrieval examples of line drawings, in which, for each row, the left is the query image, and the rest list the top 5 returned line drawings images from the *MPEG 7 CE Shape-1 Part B* benchmark.

Quantitative Performance on the *CADAL Line drawing*. Figure 13 shows the Precision-Recall curves on the CADAL Line drawing benchmark with comparisons to [2][13][15][32]. Clearly, LISC has significantly outperformed other methods, since LISC is well designed to characterize the inner details of line drawings. This has been validated by LISC superior performance over Shape Context, while Shape Context performs fairly well in the *MPEG-7 CE Shape 1 Part B* benchmark.

Figure 14 further shows the retrieval examples from the CADAL Benchmark, in which, for each row, the left is the query image, and the rest list the top 5 returned line drawings images.

In summary, the proposed LISC descriptor works well in retrieving line drawings, which is able to discriminate those challenging line drawings with rich and complicated inner details.

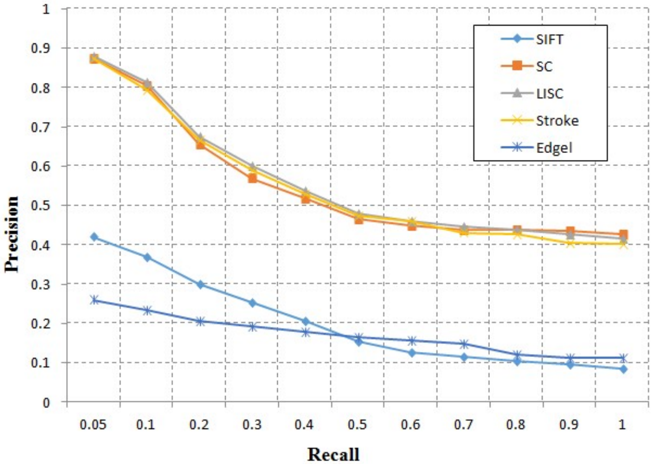


Fig. 11. Precision-Recall curves on the *MPEG-7 CE Shape 1 Part B* benchmark.

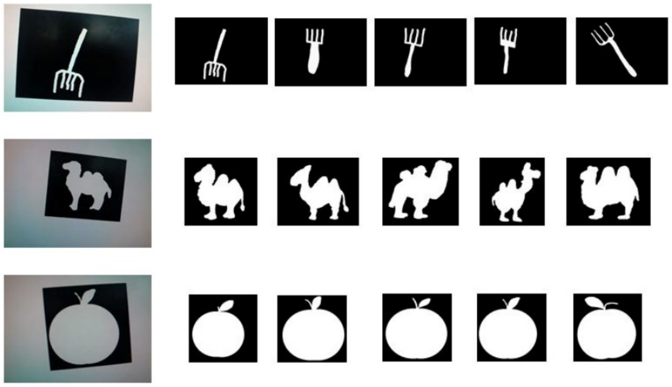


Fig. 12. Exemplar mobile queries as well as the top returned line drawing images on the *MPEG-7 CE Shape 1 Part B* benchmark.

B. *On Hamming distance KD-Tree*

Memory Cost. To evaluate the advantage of HD KD-Tree, we first study the memory cost, which is a crucial issue for scalable visual search. By using Hamming signature, the memory consumption from storing local descriptor is largely reduced. For instance, the size of a SIFT descriptor can be greatly reduced from 128 bytes to 64 bits, which has achieved a large memory reduction of about 16:1. Suppose we store the descriptors of 10,000 English document images, the online memory cost can be reduced from 1.9 GB to 255 MB. As a result, together with the KD-Tree indexing structure, the overall memory cost will be less than 400 MB.

Time Complexity. Based on HD KD-Tree, the online search mainly involves two phases, i.e. the nearest neighbor search in each leaf node, and the recursive leaf search within the tree. As shown in Table III, there exist a tradeoff between the average feature number in each leaf, and the recursion number. Adding more features would slow down the search within a node, but reduce the possibility of recursive scrolling as well. In our work, we empirically set the maximum number of features in each leaf $\alpha = 60$ with the maximum recursive number 10.

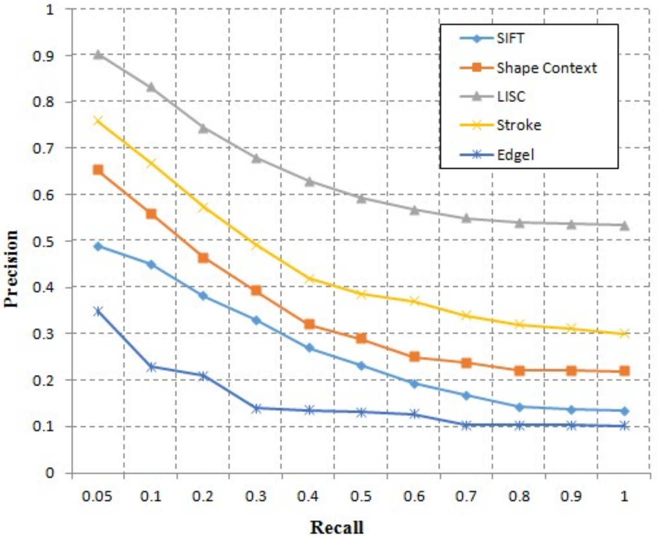


Fig. 13. Precision-Recall curves on the *CADAL Line drawing* benchmark.

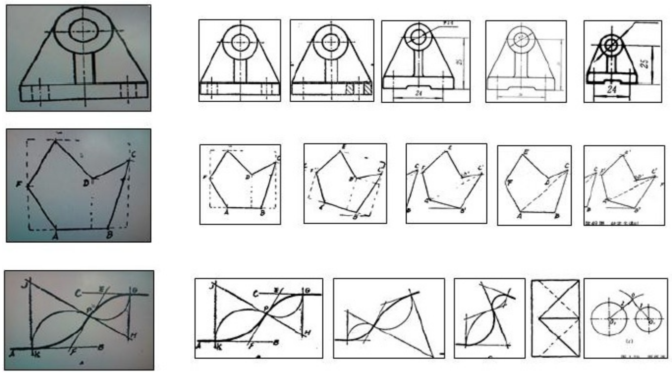


Fig. 14. Exemplar queries as well as their top five returned images on the CADAL benchmark.

Baselines and Evaluation Protocol. We compare the HD KD-Tree with the most representative Scalable Vocabulary Tree (SVT), which is built upon the hierarchical k-means clustering. We use mean Average Precision (mAP) to evaluate retrieval performance, which is defined by:

$$mAP = \frac{1}{K} \sum_{k=1}^{K} \frac{\sum_{r=1}^{N^k_{relevant}} P(r)}{N^k_{relevant}}. \qquad (12)$$

where $k = 1$ to K denote the query for evaluation. $N^k_{relevant}$ is the number of relevant documents to the kth query; r is the rth relevant document; $P(r)$ is the precision at the cut-off rank of document r.

Evaluation Benchmarks. We collect both Chinese and English document images to evaluate the performance of HD KD-Tree. For the *Chinese Document Image* benchmark, we collect 10,000 scanned book pages from the CADAL project. In this benchmark, around 70% document images are with line drawings, and the rest 30% are with only textual paragraphs. In total 6,184,436 features are extracted from these Chinese document images. For the *English Document Image* benchmark, we collect the papers from ICIP 2011 and ICASSP 2011 electronic proceedings. Over 8,000 document images are collected. In total 12,626,492 features are extracted from these English document images.

TABLE III
INFLUENCE OF THE MAXIMUM FEATURE NUMBER PER NODE IN HD KD-TREE.

Max N. in each leaf	1	10	20	30	40	50	60
Recursive N. (mAP=0.85)	450	45	27	15	12	9	7

TABLE IV
MEMORY, MAP, AND TIME ON CHINESE DOCUMENT IMAGES.

Approach	mAP	Memory (Fea.)	Memory (Ind.)	Time(Fea.)
SVT	80.83%	256 MB	42 MB	0.45ms
KD-Tree	94.89%	796 MB	247 MB	0.31ms
HD KD-Tree	89.02%	120 MB	4 MB	0.10ms

To form the test query set, we randomly select 100 images from the *Chinese Document Image* collection. The printed out documents are captured by using a mobile phone camera to generate mobile image queries. For each document, five images are snapped with camera rotation and scale changes, yielding in total 500 image queries. Likewise, we construct the mobile query set for the *English Document Image* collection.

Parameter Settings. Regarding the HD KD-Tree, we set the maximum number of features in each leaf as 60, and the maximum recursive as 10. Traditional KD Tree is used as a baseline approach, which incurs at most recursive 700 . For the vocabulary tree based on hierarchical k-means clustering, we have 5 hierarchical layers, each layer containing 10 clusters. In clustering, when a node contains less than 60 points, the clustering within this node terminates.

Memory and Time Cost on the Chinese Document Benchmark. By vocabulary tree, we train a dictionary with $8,357$ nodes (code words), where the hierarchical structure costs about 256 MB. For both KD-Tree and HD KD-Tree, the complexity comes from, and as linear to, the number of features to be indexed. Table IV lists the comparison in terms of mAP, memory, and time cost. For the *Chinese Document Image* benchmark, both HD KD-Tree and KD-Tree outperform the vocabulary tree in terms of mAP. Comparing to KD-Tree, our HD KD-Tree is much more memory efficient.

Memory and Time Cost on the English document benchmark. For vocabulary tree, we train a dictionary with $9,726$ nodes (words), where the hierarchical structure costs about 256 MB . Table V shows the comparison in terms of mAP, memory, and time cost. For the *English Document Image* benchmark, both HD KD-Tree and KD-Tree significantly outperform the vocabulary tree in terms of mAP. Likewise, our HD KD-Tree is much more memory efficient than KD-Tree.

HD KD-Tree vs. KD-Tree. Figure 15 compares the performance between HD KD-Tree and KD-Tree. It can be observed that increasing recursive number tends to bring about more mAP gap between two approaches. This phenomenon can be attributed to the "relative" characteristics of the Hamming distance between two features within one leaf node versus cross two different leaf nodes. Given a leaf node, Hamming distance does replace the traditional Euclidean distance; however, the absolute distance of two features from different leaf

TABLE V
MEMORY, MAP, AND TIME ON ENGLISH DOCUMENT IMAGES.

Approach	mAP	Memory(Fea.)	Memory(Ind.)	Time(Fea)
SVT	10.53%	247 MB	42 MB	0.35ms
KD-Tree	90.89%	1920 MB	505 MB	0.39ms
HD KD-Tree	85.65%	120 MB	8 MB	0.14ms

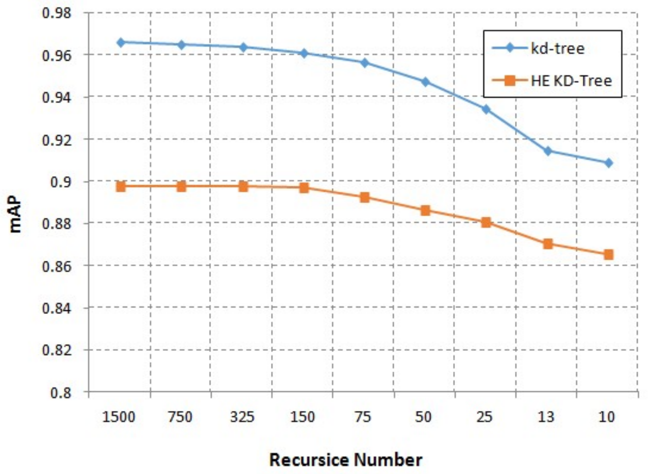


Fig. 15. On the comparisons between KD-Tree and HD KD-Tree.

nodes are undetermined, which is impossible to be recovered from their corresponding Hamming signatures. When more recursive operations occur in nearest neighbor searching, more features from different leaf nodes would enter the ranking queue . Unfortunately, it is more likely to bring about wrong nearest neighbors because of the improper feature ranking based on the node dependent relative feature distance, rather than the absolute distance.

SVT is an alternative approach to descriptor quantization, which is indeed problematic for retrieving English documents. Different from K-D Tree or HD K-D Tree based nearest neighborhood search, SVT introduces a feature space quantization to speed up the search process. However, SVT may be subject to serious quantization errors, i.e. assigning neighbor descriptors into distinct codewords. Such negative effects are dominant in quantizing local features from English documents, since English characters is very limited, thereby yielding too many visually similar patterns of different granularity to be quantized properly. But for Chinese documents, since the Chinese characters are pictograph, which are more diverse and discriminative in terms of visual patterns, SVT is basically workable for scalable visual search by feature quantization. This findings also indicate that SVT is sensitive in terms of quantization errors, which relies on the image databases to learn the visual dictionary however.

SVT vs. K-D and HD K-D Trees. The direct use of SVT to search English documents is problematic, because of considerable quantization errors in feature space. To solve the quantization issue, K-D Tree is an alternative, which works as a nearest neighbor search with the mechanism of backtracking to ensure the accuracy of nearest neighbor search. However, K-D tree is weak in terms of huge memory cost. Hence Hamming distance and binary feature space are introduced to replace the original Euclidean space. Finally, the proposed HD K-D Tree attempts to figure out an optimal tradeoff between higher memory consumption and lower quantization errors.

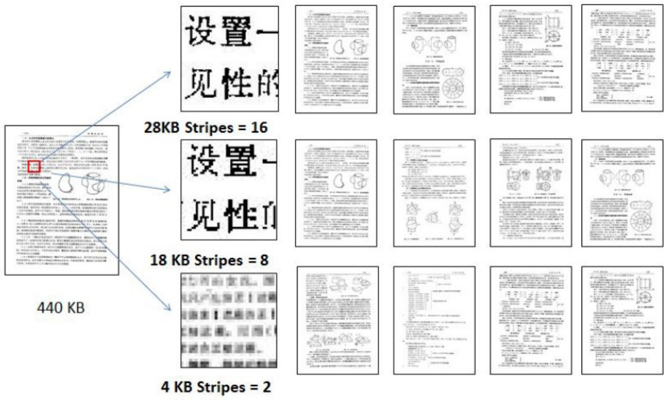


Fig. 16. The effects of stripe values on compression rates , image quality, as well as search accuracy.

C. On Query Compression

Benchmarks and Baselines. We compare the JBIG2 based compression scheme with other alternatives in the following four benchmarks: *CADAL Chinese Document Image*, *English Document Image*, *MPEG-7 CE Shape 1 Part B* and *CADAL Linedrawing*, all of which have been introduced in the previous subsections. We compare the JBIG2 compression scheme to both Product Quantization (PQ) based [26] and JPEG + VSQT [27] based compression schemes. PQ based scheme is to compress the descriptors of local patches, while JPEG + VSQT is to compress the query image by using JPEG with the optimized quantization table. To make a fair comparison, we setup different compression rates and compare the retrieval performance at different rates with different approaches.

Parameters Tuning. For the JBIG2 compression, the most important parameter is the stripes value, i.e., the larger the stripes value is, the lower the compression rate is, and vice versa. In this work, we test the stripes values of 2, 8,and 16 to evaluate the performance at different compression rates. Figure 16 shows image examples on how stripes values effect the compression rates as well as the retrieval performance.

For the PQ based scheme, the compression rate is subject to both feature subdivision and the cluster number k in each subdivision. For SIFT, we have PQ subdivision as 16, 8, and 4 respectively, and k as 10. For LISC, we have PQ subdivision as 15, 10, and 5 respectively, and k as 10. And for VSQT [27], the same JPEG quantization table as in [27] is applied.

Generally speaking, for document images at the resolution 800 × 1000, JBIG2 can reduce the bit rate by 94% (from 500KB down to 30KB), without any noticeable retrieval performance degradation.

In addition, as shown in Figure 17, the first row of images indicate that by directly binariazing images without Retinex, the result is indeed unsatisfactory, while the second row of images show that by using Retinex based image enhancement, the quality of interest points can be improved.

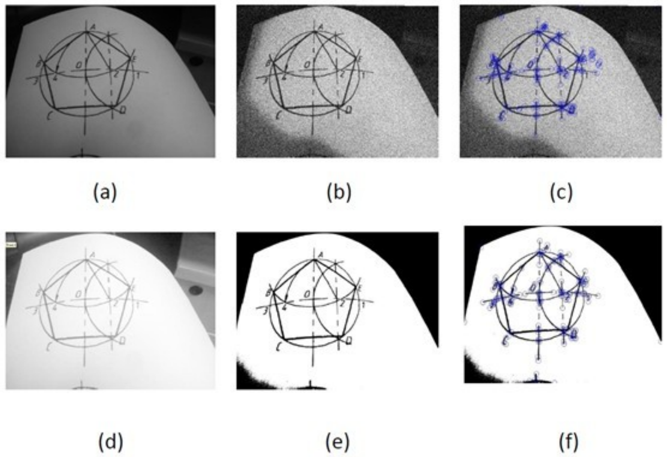


Fig. 17. (a) Original query image. (b) Directly OTSU binarizing from a. (c) Interest points extracted from b. (d) Enhanced image by Retinex. (e) Binarizing from d. and (f) Interest point extracted from (e).

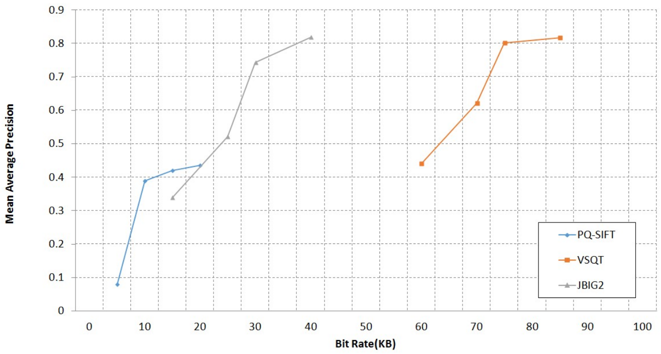


Fig. 18. Rate distortion comparison on the *Chinese Document Image* benchmark.

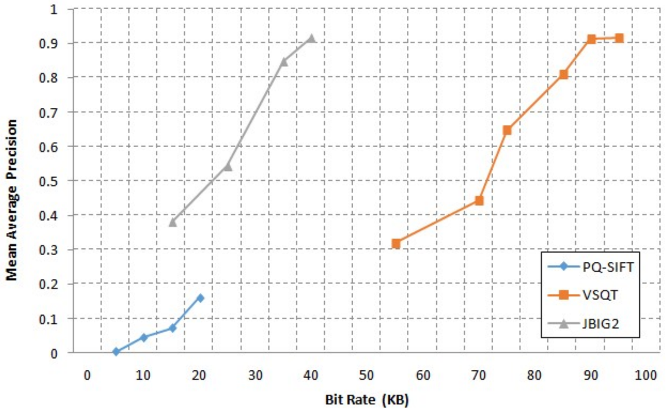


Fig. 19. Rate distortion comparison on the *English Document Image* benchmark.

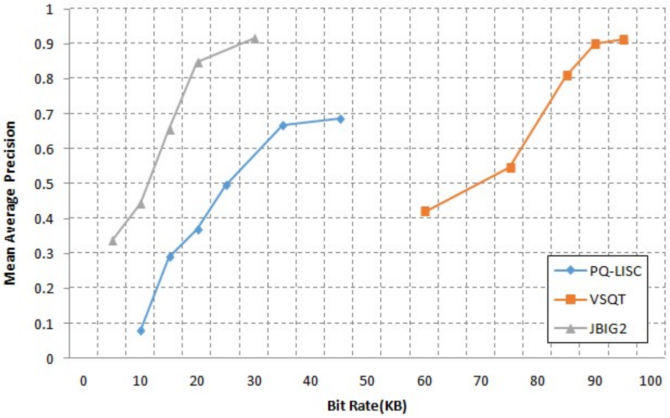


Fig. 20. Rate distortion comparison on the *MPEG 7 Shape 1 Part B* benchmark.

Rate Distortion Analysis for Document Images. Figure 18 and Figure 19 show the rate distortion analysis in both Chinese and English documents, with comparisons to PQ-SIFT [26], VSQT [27], and JBIG2 [28]. For Chinese document images, the original resolution is 1000×800, and image size is $500-700$ KB in JPEG format. As shown in Figure 18, the proposed JBIG2 compression has achieved a promising mAP of up to 82% at lower bit rates of less than $30-40$ KB over the *Chinese Document Image* benchmark.

Similar findings can be observed in the *English Document Image* benchmark, as shown in Figure 19. Note that the performance of PQ-SIFT features is very poor. This is due to the weakness of Production Quantization [26]; that is, when quantizing the feature subspace of SIFT, the quantization errors largely degenerate the search accuracy in English document images. With the proposed JBIG2 compression, we maintain the performance of about 90% mAP at the image size of less than 40 KB, which has greatly outperformed the JPEG based compression [27] in terms of mAP versus compression rate.

Rate Distortion Analysis for line drawings. For line drawings, since SIFT is poor in describing the shapes, we setup a comparison baseline PQ-LISC by applying product quantization to the proposed LISC descriptors. Although PQ-LISC reduces the size of LISC descriptors, PQ quantization would cause the loss of discriminative power, thereby yielding serious drop of retrieval mAP. On the other hand, while VSQT [27] can maintain comparable retrieval performance as JBIG2, the compression rate of VSQT is much worse than JBIG2, as shown in Figure 20 and Figure 21.

VII. Conclusion and Future Work

With the proliferation of mobile devices, there is an emerging need to retrieve document images by mobile visual search. However, there exist fundamental challenges to fulfill the functionality of mobile visual search in digital library, i.e., lacking a descriptor to characterize line drawings, lacking an effective yet memory-light indexing and online matching pipeline to fast search document images, as well as the challenge of reducing mobile query delivery latency, etc. To tackle these challenges, we have proposed a general framework towards mobile document image retrieval. First, Local Inner-distance Shape Context descriptor is designed to represent line drawings, widely available in document images, which has been empirically shown to be robust against a variety of mobile query distortions. Second, Hamming Distance KD-Tree is proposed to address the issue of high memory cost in building up the indexing structure towards scalable search.

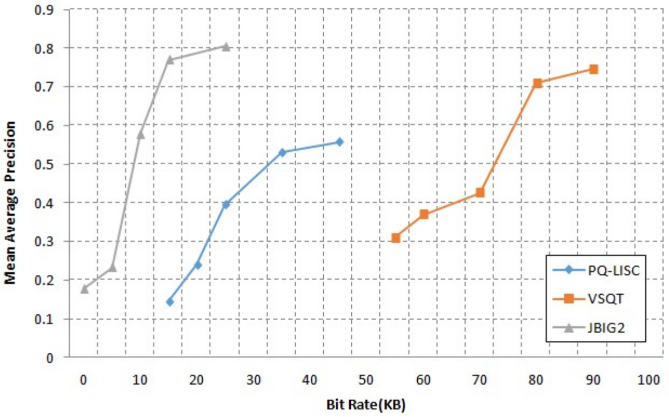


Fig. 21. Rate distortion comparison on the *CADAL Linedrawing* benchmark.

Third, JBIG2 based compression scheme of low complexity is introduced to reduce the query delivery latency while maintaining comparable search accuracy. These innovative components have been successfully integrated and demonstrated in a mobile document retrieval prototype, which provides rich search functionality for book covers, book pages, scanned documents, as well as line drawings. Extensive quantitative comparisons to a variety of alternative approaches have been carried out, which has proved the merits of the proposed framework.

However, to implement a user friendly document image retrieval, a synergic collaboration of textual regions, line drawings, and multilingual page representation is important. In this work, we simply apply heuristic rules to rank the documents with respect to separate document elements. A full-fledged ranking fusion is included in our future work. In addition, geometric constraints have not been used in document retrieval, as the current emphasis is on the essential representation of document elements towards mobile document retrieval. Undoubtedly, to further improve the retrieval performance, reranking is a meaningful approach. For example, geometric verification, or other cues based re-ranking. In addition to effective re-ranking algorithms, the efficiency of re-ranking is an important issue as well. How to develop effective and efficient re-ranking strategies towards mobile document image retrieval is also one of our future work.

REFERENCES

[1] B. Kimia, A. Tannenbaum, and S. Zucker. Shapes, shocks, and deformations, I: The components of shape and the reaction-diffusion space. *IJCV*, 1995. 3

[2] S. Belongie, J. Malik, and J. Puzicha. Shape context: A new descriptor for shape matching and object recognition. *PAMI*, 2002. 2, 3, 4, 5, 6, 9

[3] H. Ling and D. Jacobs. Shape classification using the inner-distance. *PAMI*, 2007. 2, 4, 5

[4] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation *PAMI*, 2004. 3, 5

[5] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 2005. 3, 5

[6] R. Basri, L. Costa, D. Geiger, and D. Jacobs. Determining the similarity of deformable shapes. *Vision Research*, 1998. 3

[7] A. Ratarangsi and R. Chin. Scale-based detection of corners of planar curves. *PAMI*, 1992. 2, 3, 6

[8] Z. Tu and A. Yuille. Shape matching and recognition-using generative models and informative features. *ECCV*, 2004. 4

[9] X. Chen, H. Nelson, and H. Yung. Corner detector based on global and local curvature properties. *Optical Engineering*, 2008. 5

[10] J. Pu and K. Ramani. On visual similarity based 2D drawing retrieval. *Computer Aided Design*, 2005. 4

[11] D. Chen, S. Tsai, V. Chandrasekhar, G. Takacs, J. Singh, and B. Girod. Tree histogram coding for mobile image matching. *DCC*. 2009. 2, 4, 7

[12] V. Chandrasekhar, G. Takacs, D. Chen, S, Tsai, R. Grzeszczuk, and B. Girod. CHoG: Compressed histogram of gradients a low bit-rate feature descriptor. *CVPR*. 2009. 2, 4, 7

[13] D. Lowe. Distinctive image features form scale-invariant keypoints. *IJCV*, 2004. 4, 9

[14] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. *ECCV*. 2006. 4

[15] T. K. Bhowmik, U. Bhattacharya, and S.. Parui. Recognition of bangla handwritten characters using an MLP classifier based on stroke features. Neural Information Processing. Springer Berlin Heidelberg, 2004. 2, 3, 4, 9

[16] R. Ji, L.-Y. Duan, J. Chen, H. Yao, J. Yuan, Y. Rui, and W. Gao. Location discriminative vocabulary coding for mobile landmark search. *IJCV*. 2011. 2, 4, 7

[17] W. Zhang and J. Kosecka. Image based localization in urban environments. *3DPVT*, 2006. 4

[18] G. Schindler and M. Brown. City-scale location recognition. *CVPR*, 2007. 4

[19] E. Eade and T.Drummond. Unified loop closing and recovery for real time monocular SLAM. *BMVC*, 2008. 4

[20] T. Yeh, K. Tollmar, and T. Darrell Searching the web with mobile images for location recognition. *CVPR*, 2004. 4

[21] D. Crandall, L. Backstrom, and D. Huttenlocher. Mapping the world's photos. *WWW*, 2009. 4

[22] Efficient use of MPEG-7 edge histogram descriptor. *MPEG CDVS Proposal*, 2011. 2, 4

[23] H. Jegou, M. Douze, and C. Schmid. Hamming distance and weak geometric consistency for large scale image search. *ECCV*, 2008. 6

[24] Z. Rahman, D. Jobson, and A. Woodell. Retinex processing for automatic image enhancement. *Journal of Electronic Imaging*, 2004. 7

[25] T. Nakai and K. Kise. Use of affine invariants in locally likely arrangement hashing for camera-based document image retrieval. *Document Analysis System*, 2006. 3

[26] H. Jegou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *PAMI*, 2010. 11, 12

[27] L.-Y. Duan, X. Liu, T. Huang, and W. Gao. Optimizing JPEG quantization table for low bit rate mobile visual search. *VCIR*, 2012. 11, 12
.

[28] ISO/IEC JTC1/SC29/WG1 N1359. JBIG2 Final Committee Draft, July 1999. 2, 8, 12

[29] S. Mao and A. Rosenfeld. Document structure analysis algorithms: A literature survey. *SPIE Electronic*, 2003. 3

[30] D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. *CVPR*, 2006. 2, 3

[31] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. *ICCV*, 2003. 2, 3, 6

[32] Y. Cao, C. Wang, L. Zhang, and L. Zhang. Edgel index for large-scale sketch-based image search. *CVPR*, 2012. 2, 4, 9

[33] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 1975. 2, 7

[34] K. Khurshid. Comparison of Niblack inspired binarization methods for ancient documents. *SPIE Electronic Imaging*, 2009. 8

[35] S. Jeannin and M. Bober. Description of core experiments for MPEG-7 motion/shape. *MPEG-7,ISO/IEC/JTC1/SC29/WG11/MPEG99/N2690*, 1999. 8

[36] W. Liu, J. Wang, R. Ji, Y.-G. Jiang, and S.-F. Chang. Supervised hashing with kernels. *CVPR*, 2012. 2

[37] Y. Freund, S. Dasgupta, M. Kabra, and N. Verma. Learning the structure of manifolds using random projections. *NIPS*. 2007. 6

[38] V. Santosh. Randomly-oriented k-d Trees Adapt to Intrinsic Dimension. *ARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*. 2012. 6

Ling-Yu Duan (M06) received the Ph.D. degree in Information Technology from The University of Newcastle, Australia, in 2007, the M.Sc. degree in Computer Science from the National University of Singapore, Singapore, and the M.Sc. degree in Automation from the University of Science and Technology of China, Hefei, China, in 2002 and 1999, respectively. Since 2008, he has been with Peking University, Beijing, China, where he is currently an Associate Professor with the School of Electrical Engineering and Computer Science. Dr. Duan is leading the group of visual search in the Institute of Digital Media, Peking University. Since 2012, Dr. Duan is the deputy director of the Rapid-Rich Object Search (ROSE) Lab, a joint lab between Nanyang Technological University, Singapore and Peking University, China, with a vision to create the largest collection of structured domain object database in Asia and to develop rapid and rich object mobile search. Before that, he was a Research Scientist in the Institute for Infocomm Research, Singapore, from 2003 to 2008. His interests are in the areas of visual search and augmented reality, multimedia content analysis, and mobile media computing. He has authored more than 80 publications in these areas. He is a member of the ACM.

Rongrong Ji (M'11) is currectly wokring at Xiamen University since 2013, where he directs the Intelligent Multimedia Technology Laboratory and serves as a Dean Assistant in the School of Information Science and Technology. Before that, he used to be a Postdoc research fellow in the Department of Electrical Engineering, Columbia University from 2010 to 2013, worked with Professor Shih-Fu Chang. He obtained his Ph.D. degree in computer science from Harbin Institute of Technology, under supervision of Professor Hongxun Yao. He had been a visiting student at University of Texas of San Antonio worked with Professor Qi Tian, and a research assistant at Peking University worked with Professor Wen Gao in 2010, a research intern at Microsoft Research Asia, worked with Dr. Xing Xie from 2007 to 2008.

Zhang Chen received his M.S. degree from the school of EE & CS of Peking University, under the supervision of Prof. Duan Lingyu . He is currently working with Tencent Corp. His research interest is in visual search and digital document retrieval.

Tiejun Huang (M01) received the B.S. and M.S. degrees from the Department of Automation,Wuhan University of Technology, Wuhan, China, in 1992 and the Ph.D. degree from the School of Information Technology and Engineering, Huazhong University of Science and Technology, Wuhan, in 1999. He was a Postdoctoral Researcher from 1999 to 2001 and a Research Faculty Member with the Institute of Computing Technology, Chinese Academy of Sciences. He was also the Associated Director (from 2001 to 2003) and the Director (from 2003 to 2006) of the Research Center for Digital Media in Graduate School at the Chinese Academy of Sciences. He is currently a Professor with the National Engineering Laboratory for Video Technology, School of Electronics Engineering and Computer Science, Peking University, Beijing, China. His research interests include digital media technology, digital library, and digital rights management. Dr. Huang is a member of the Association for Computing Machinery.

Wen Gao (M92CSM05CF09) received the Ph.D. degree in electronics engineering from the University of Tokyo, Tokyo, Japan, in 1991. He is currently a Professor of computer science with the Institute of Digital Media, School of Electronic Engineering and Computer Science, Peking University, Beijing, China. Before joining Peking University, he was a Professor of computer science with the Harbin Institute of Technology, Harbin, China, from 1991 to 1995, and a Professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing. Dr. Gao served or serves on the editorial boards for several journals, such as the IEEE Transactions on Circuits and Systems for Video Technology, IEEE Transactions on Multimedia, IEEE Transactions on Autonomous Mental Development, EURASIP Journal of Image Communications, and Journal of Visual Communication and Image Representation. He has chaired a number of prestigious international conferences on multimedia and video signal processing, such as IEEE ICME and ACM Multimedia, and also served on the advisory and technical committees of numerous professional organizations.