Robust Image Fingerprinting Based on Feature Point Relationship Mining

Xiushan Nie, *Member, IEEE*, Xiaoyu Li, Yane Chai, Chaoran Cui, Xiaoming Xi, and Yilong Yin

***Abstract*— Local feature points have been widely employed in robust image fingerprinting. One of their intrinsic advantages is their invariance under geometric transforms. However, their robustness against certain attacks that modify the positions of points, such as additive noising and blurring, is limited. In addition, local-feature-point-based approaches ignore the distribution of the feature points. In this paper, we harness feature point relationships, including local structures and global relevance, to overcome these limitations. In the relationship mining strategy, Delaunay triangulation is first applied to the feature points to capture their geometric structures. Subsequently, local structures are represented by searching for an independent set in the mapping graph constructed via Delaunay triangulation, whereas the global relevance is represented by the Laplacian of the graph. Finally, the local structures and global relevance are used as input to the quantization process of the image fingerprinting system. In the process of quantization, we propose an unsupervised quantization strategy called between-cluster distance-based quantization to preserve the neighborhood structure between the binary fingerprint space and the original feature space. Experimental results show that the proposed method achieves effective performance under common modifications.**

***Index Terms*— Image fingerprinting, image copy detection, relationship mining, local structure, global relevance.**

I. Introduction

With the rapid development of multimedia and internet technologies, the openness of networks has resulted in easy and inexpensive access to images. Furthermore, these images can easily be altered, tampered with, duplicated, and reshared over the network. As a result, there has been a proliferation of pirated copies or near-duplicates of images that infringe on the copyrights of the image producers. Consequently, this presence of massive numbers of copies has given rise to a strong demand for effective image copy detection methods for use in many applications, such as copyright enforcement and image database cleansing.

Traditionally, copies of images have been detected through watermarking techniques, in which imperceptible watermarks are embedded into images to prove their authenticity. Unfortunately, one limitation of such approaches is that a watermark that is directly embedded into an image distorts the image contents to some degree. By contrast, robust image fingerprinting, also called image hashing, involves the extraction of certain image features to calculate compact binary digests that enable efficient content identification without directly altering the image data. As a result, image fingerprinting has recently attracted increasing attention.

It should be noted that the application scenarios of robust image fingerprinting that are considered in this study differ from those related to image retrieval [1]–[4] or object recognition [5], [6]. In image retrieval, hashing is performed to retrieve images that contain similar objects or are from the same categories as the query images, and the major concern is how to cope with images taken from different viewpoints or containing occlusions. Image fingerprinting or hashing is also used in image forgery detection. Yan *et al.* [7], [8] proposed two efficient strategies for image tampering location and object-level tampering. By contrast, the image fingerprinting method considered in this study is mainly aimed at protecting the copyright of digital images, and to this end, a compact and secure signature is generated to represent each image. Robustness and discrimination are the major concerns which are evaluated against image distortions arising from transmission noise, lossy compression, geometric attacks, and so on. Such distortions and attacks generally do not perceptibly alter the image content or introduce viewpoint changes or large occlusions [9].

In addition, robust image fingerprinting methods typically focus on visual similarity, and they usually utilize low-level or handcrafted features, such as pixels, textures and local feature points, to generate image fingerprints, whereas methods for image retrieval or classification typically focus on semantic meanings and consequently utilize high-level features, such as features learned by deep learning frameworks. For example, two images are with the same semantic

Manuscript received August 9, 2017; revised November 14, 2017 and December 24, 2017; accepted December 26, 2017. Date of publication January 8, 2018; date of current version February 12, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61671274, Grant 61573219, Grant 61701281, Grant 61701280, in part by the Postdoctoral Science Foundation under Grant 2016M592190, in part by Shandong Provincial Natural Science Foundation under Grant ZR2017QF009, in part by the Shandong Provincial Key Research and Development Plan under Grant 2017CXGC1504, in part by the Fostering Project of Dominant Discipline and Talent Team of Shandong Province Higher Education Institutions, and in part by the Fostering Project of Dominant Discipline and the Talent Team of SDUFE. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Stefano Tubaro. *(Corresponding authors: Xiushan Nie; Yilong Yin.)*

X. Nie, X. Li, Y. Chai, C. Cui, and X. Xi are with the School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250014, China (e-mail: niexiushan@163.com).

Y. Yin is with the School of Computer Science and Technology, Shandong University, Jinan 250100, China (e-mail: ylyin@sdu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIFS.2018.2790953

meaning "wedding", but they visually look like in different scenes, which are in a church and on a beach, respectively. In these cases, they are not copied or near-duplicated images but with the same semantic meaning, and they are not the cases handled by image fingerprinting, but handled by image retrieval or classification.

In the context of image fingerprinting, feature representation and quantization are two main components. In these tasks, various types of features, such as histograms and keypoints, are extracted to represent images, and the extracted features are then quantized into binary sequences to form the final fingerprints. An image's fingerprint sequence is typically a short binary string that is adopted as a persistent fingerprint of the image [10], [11]. The fingerprint sequence of an image constitutes a digest of the image content and is robust against image modifications or adjustments that do not alter the fundamental image information, such as rotation, JPEG compression, contrast changes, and noise level adjustments [12].

A robust image fingerprinting method H that is sensitive to a private key K can be described as follows [13]:

- $H(K, I)$ is uncorrelated with $H(K, I_a)$ when two images I and I_a are dissimilar;
- $H(K, I)$ is strongly correlated with $H(K, I_a)$ when I and I_a are similar in content;
- $H(K, I)$ is uncorrelated with $H(K^a, I)$ when $K! = K^a$.

The private key is used to enhance the security of the fingerprinting system. Randomization strategy via secret keys, such as randomly selected image blocks [14], is popularly used in the fingerprinting system, and it can be definitely applied in the proposed method. These strategies and their analysis have been described in detail in the existing methods, so we do not discuss these issues in the present study.

Image fingerprinting can reduce the dimensionality and storage cost of an image by virtue of the resulting binary representation. Furthermore, the use of an efficient fingerprinting scheme can help to achieve a constant or sub-linear search time complexity in the big data application space.

Regarding feature representation, global and local features are two commonly used types of features for fingerprint generation [9], [15]–[18] [19]. Global features (such as textures) can be obtained over an entire image. For example, in [20], Li *et al.* proposed Gabor texture descriptors using the adjustability of Gabor transforms in terms of direction and scale. These descriptors are invariant to rotation and scaling, but there is a high false alarm rate for large-angle rotations. Image fingerprinting algorithms based on global features require only simple calculations and are highly efficient. However, because of their poor resistance to geometric attacks, especially cropping and rotation, researchers prefer to detect local features of images. Compared with global features, local features offer good image recognition capabilities.

Local feature points (also called keypoints or interest points), such as the Harris detector, the Scale-Invariant Feature Transform (SIFT) [9], and the Speeded-Up Robust Features (SURF) detector [21], are widely employed. In image fingerprinting algorithms, local-feature-point-based descriptors are directly or indirectly used to generate fingerprints. Other patterns of local feature descriptors, such as local binary patterns [10] and structure skeletons [22], can also be employed in image or video fingerprinting. In addition, local spatial- or frequency-domain pixels arranged according to special patterns [23], [24] have also been used to generate image fingerprints.

In general, local-feature-based methods are robust to geometric image modifications, such as rotation, shifting, and scaling. However, the feature points may be sensitive to certain image modifications, such as noise addition, blurring, and post-production manipulations (such as letterboxing or caption insertion) because the positions of local feature points may be altered as a result of such modifications. In addition, these local-feature-based methods usually ignore the local distributions of the detected feature points. To overcome these limitations, we generate image fingerprints using the relationships between different local feature points, rather than the feature points themselves, as these relationships are more stable under most modifications and can also reveal the local distributions of the feature points.

However, the relationships among feature points and the task of distribution mining for image fingerprinting are non-trivial for the following two reasons. 1) *Robustness and discrimination*. Because of the need for robustness and discrimination, it is desired that the relationships should remain stable when the feature points undergo certain content-preserving modifications, and they should also be distinct for different images. Therefore, a major challenge lies in finding a robust and unique form in order to represent these relationships. Fortunately, Delaunay triangulation, which yields a representation of the geometric structure of a point set, satisfies these requirements. 2) *Sparsity*. Because of the complexity of the interaction effects between different points, the relationships among feature points are always redundant and difficult to capture. Directly employing the relationship between every point pair would be excessively time consuming. Consequently, the sparse sampling of these relationships and the capturing of different views of these relationships are important in an image fingerprinting system. In this study, we achieve sparsity by searching for an independent set in a mapping graph.

Quantization is another key step in image fingerprinting. During the quantization process, image features are quantized into binary codes in a Hamming space, and the similarity between binary codes is evaluated based on the Hamming distance. Usually, in the quantization stage, real values are quantized into a binary string via thresholding. Single-bit quantization (SBQ) and hierarchical quantization (HQ) [25] are two primary strategies for this purpose. In SBQ, given a hash function $h_k(x)$, $h_k(x) = 1$ if $f_k(x) \geq \theta$. Otherwise, $h_k(x) = 0$. Here, θ is a threshold. In HQ, each projected dimension is divided into four regions using three thresholds, and two bits are used to encode each region. The Hamming distance is a widely used measure of the similarity after quantization; however, both SBQ and HQ will destroy the neighborhood structure in the original feature space. In SBQ, for example, if a given point is below or above the threshold, the binary fingerprint value is 0 or 1, respectively. Therefore, in the example shows in Fig. 1, points A1 and B1 will

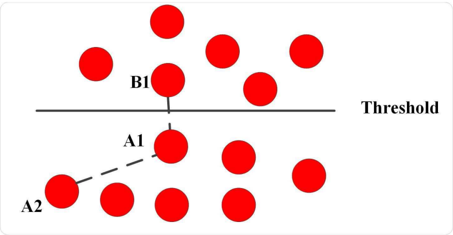


Fig. 1. An illustration of one of the limitations of quantization.

be assigned values of 0 and 1, respectively, whereas points A1 and A2 will both be assigned values of 0. As a result, the Hamming distance between A1 and B1 will be larger than that between A2 and A1, which is obviously inconsistent with the distances in the original space. Therefore, some limitations in preserving the neighborhood structure are encountered when applying these quantization schemes. In this study, we propose a new quantization strategy called between-cluster distance-based quantization (BCDQ) to overcome these limitations.

In summary, we present a novel relationship mining strategy for image fingerprinting based on Delaunay triangulation and graphs in this study. Two types of relationships, local structures and global relevance, are used in the proposed approach. Local structures represent the geometric structures between different feature points in local image regions, whereas the global relevance represents the degree of global relatedness among all feature points. Once these relationships have been obtained, we use a new quantization strategy to generate the final image fingerprint, which can effectively preserve the neighborhood structure. The main contributions of this work are as follows.

(1) Robust relationships among feature points, including local geometric structures and global relevance, are considered instead of the feature points themselves for robust image fingerprinting. These relationships are more robust to various modifications applied to images. In our strategy, the local geometric structures and global relevance are used to represent the image information from different viewpoints.

(2) A novel relationship representation and mining strategy is proposed based on Delaunay triangulation and undirected graphs. During the process of mining and representing the relationships, we first connect the feature points via Delaunay triangulation, in which the relationships among feature points are encoded as triangles (which we therefore call relationship triangles (RTs)). To avoid mutual influences between RTs, the RTs are projected to an undirected graph, and an independent set is selected from the graph to extract the local relationships. Meanwhile, the global relationship is generated based on the Laplacian matrix of this graph.

(3) An unsupervised quantization strategy called BCDQ is proposed to effectively preserve the neighborhood structure of the original features, and BCDQ is used to learn binary fingerprints. In this strategy, each feature dimension from all images is partitioned into different clusters, and cluster indices are then assigned according to the between-cluster distances. Finally the indices in each dimension are quantized into binary fingerprints, and the binary codes for all dimensions of an image are concatenated to represent that image. Compared with the existing quantization strategies, BCDQ can preserve more of the structure information of the original features.

The remainder of the paper is structured as follows. Section II reviews related works. Details about the proposed method are introduced in Section III. Experimental results and analyses are presented in Section IV. Section V concludes the study with a summary and a discussion of future work.

II. Related Works

A traditional image fingerprinting system includes two main components: feature extraction and quantization. Due to space constraints, we present a short literature review on feature extraction and quantization.

A. Feature Extraction

Global and local features are the two main types of features considered during feature extraction [15]. Global features, such as histogram and texture features, are extracted from a given image as a whole. Li *et al.* [20] proposed Gabor texture descriptors using the adjustability of Gabor transforms in terms of direction and scale to represent images. These descriptors are generally invariant to rotation and scaling, but they are sensitive to sufficiently high-angle image rotations. Generally, copy detection algorithms based on global features are simple to calculate and highly efficient. However, because of their poor resistance to geometric attacks, especially cropping and rotation, scholars tend to focus more on using local features to represent images.

Generally, Harris detector, Scale-Invariant Feature Transform (SIFT) [9], [26]–[28] and Speeded-Up Robust Features (SURF) detector [21] all yield well-known classes of local features that have commonly been used in existing works. In image fingerprinting algorithms, local feature point descriptors can be directly or indirectly used to generate fingerprints. Cao *et al.* [27] used a new tilt parameter and a geometric consistency constraint in the affine SIFT method to generate image fingerprints. Jégou *et al.* [28] proposed a VLAD (vector of locally aggregated descriptors)-based compact representation of SIFT features to generate image fingerprints. Other patterns of local feature descriptors, such as local binary patterns [10] and structure skeletons [22], have also been employed in image or video fingerprinting systems. Berrani *et al.* [29] computed local differential descriptors for each image, corresponding to the local regions of interest in the image, to construct image representations. Compared with global features, local features offer good image recognition capabilities. However, local features suffer from two limitations. The first one is that the feature points may be sensitive to certain image modifications that can alter the positions of local feature points. The second limitation is that these local-feature-based methods usually ignore the local distributions of the detected feature points. To overcome these limitations, we use the relationships between different local feature points to generate image fingerprints.

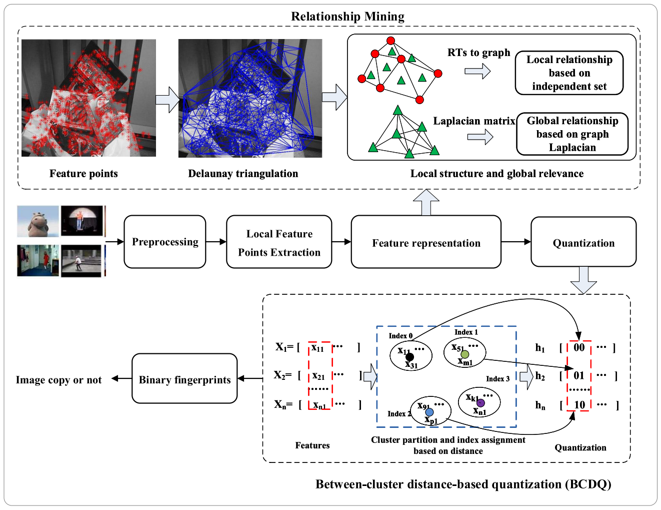


Fig. 2. Flowchart of the proposed method.

B. Quantization

Quantization is a key step in an image fingerprinting system, especially in the big data age, because quantization not only can decrease storage costs but also can accelerate the detection speed in a large-scale database. In the quantization process, real values are quantized into binary codes via thresholding. In existing works, single-bit quantization (SBQ) and double-bit quantization (DBQ) [30] are commonly used quantization strategies, in which single bits or double bits, respectively, are used to quantize each projected dimension. The hierarchical quantization (HQ) method in anchor graph hashing (AGH) [25] is another quantization strategy. Rather than using one bit, HQ uses three thresholds to divide each dimension into four regions and uses two bits to encode each region. Hence, in HQ, each projected dimension is associated with two bits. More recently, Moran *et al.* [31] proposed a variable bit quantization (VBQ) method for locality sensitive hashing, in which bits are allocated across hyperplanes. Wang *et al.* [32] presented Hamming compatible quantization (HCQ) to preserve the meaning of the similarity metric between Euclidean space and Hamming space. In general, most existing methods merely quantize the real values in each sample; they do not effectively consider the neighborhood structure of all samples in the same dimension. Intuitively, a given dimension should correspond to the same attributes in the feature vectors for all samples. Therefore, in this study, we investigate a new quantization strategy in which the structure information from all samples is considered for each dimension.

III. Proposed Method

A flowchart depicting the steps of the proposed method is presented in Fig. 2. In the proposed method, each image is first preprocessed to obtain a normalized image, without fundamentally altering the image contents. Preprocessing that does not alter the fundamental image contents is a common step for enhancing the robustness of image fingerprinting systems. We employ a bilinear interpolation method to normalize the input image size to a fixed size of 512 × 512 pixels, which is intended to increase the robustness of the algorithm with regard to image scaling operations. Following this preprocessing, various types of local feature points, such as Harris, SIFT, and SURF features, are extracted from the images. Subsequently, the local and global relationships are characterized based on Delaunay triangulation, and these relationships are adopted as the basis for the feature representation of the images. Finally, binary image fingerprints are learned using a unsupervised quantization strategy called BCDQ.

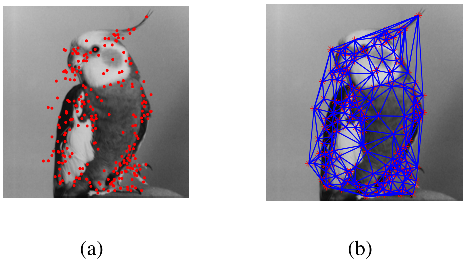


Fig. 3. Delaunay triangulation of an image: (a) local feature points and (b) results of Delaunay triangulation.

Relationship mining and BCDQ are the two main components of the proposed method. Therefore, in the following subsections, we will describe these two components in detail.

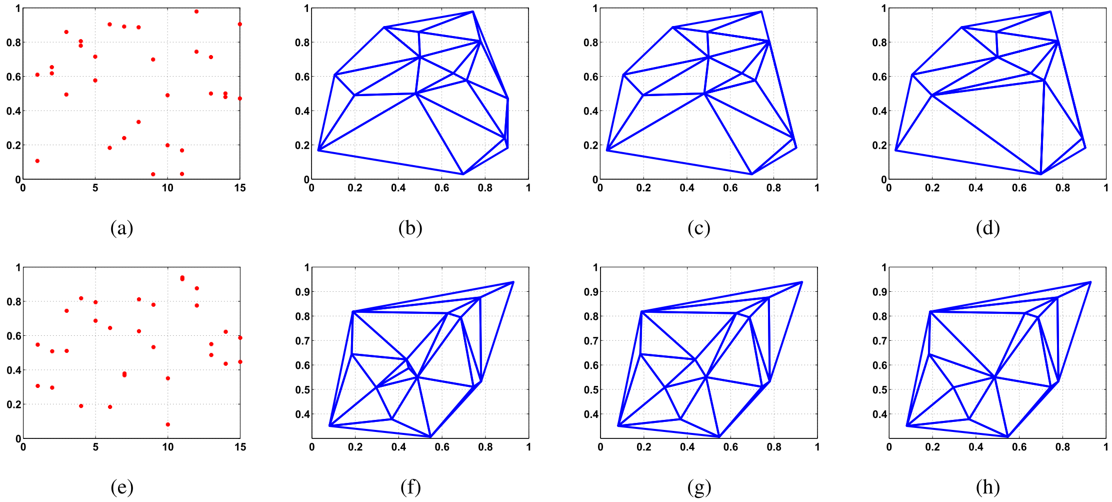
A. Relationship Mining

Local structures and global relevance are two types of feature point relationships. In the proposed method, we use Delaunay triangulation to explore the local structures and global relevance of the feature points during relationship mining. To clearly illustrate this strategy, we first present a brief introduction to Delaunay triangulation [33] and then describe how to capture these two types of relationships based on Delaunay triangulation and graph analysis.

1) Delaunay Triangulation: Consider a set P of points on a plane, and let their Voronoi diagram be denoted by $Vor(P)$. For points $p_1, p_2 \in P$, let $c(p_1)$ and $c(p_2)$ be the Voronoi cell of p_1 and p_2, respectively. The graph V of $Vor(P)$ contains a vertex for every Voronoi cell, and it contains an edge joining two vertices if the corresponding cells are adjacent. Consider the straight-line embedding of V, in which each edge joining vertices $c(p_1)$ and $c(p_2)$ is mapped to line segment $\overline{p_1 p_2}$. This embedding is the Delaunay graph of P, which we denote by D. It is the dual graph of V; it is also a planar graph, meaning that no two edges intersect.

If P is in general position, then all vertices of V are of degree three, implying that all bounded faces of D are triangles. Under the same assumption, the triangulation is unique. It has the property of being angle-optimal; i.e., it maximizes the minimum angle over all triangulations of P. Its computation can be very efficient [34]. Fig. 3 presents an example of Delaunay triangulation applied to an image, where the red stars in Fig. 3(a) indicate the SURF points extracted from the original image and Fig. 3(b) shows the results of applying Delaunay triangulation to these feature points.

To address the first challenge described in the previous section, we should find a robust and discriminative



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 4. Illustration of the robustness of Delaunay triangulation: (a) and (e) are two different original artificial data sets (each with 15 points); (b) and (f) are the Delaunay triangulations for these two original data sets; (c-d) and (g-h) are the Delaunay triangulations after the removal of one and two points from the two respective data sets.

representation of the relationships among feature points. Delaunay triangulation is a good choice for this purpose because of its desirable properties of regionality and uniqueness [35], [36]. Regionality means that manipulations such as adding, deleting, or moving a few points affect only the neighboring triangles. Uniqueness means that completely different point sets will yield different triangulations and that the same triangulation will be obtained regardless of which point is chosen as the starting point. These properties guarantee the robustness and discrimination of Delaunay triangulation.

To visually demonstrate the robustness and discrimination of Delaunay triangulation, we applied Delaunay triangulation to some artificial data points and then applied it again after removing some of the data points. The results are illustrated in Fig. 4, where Fig. 4 (a) and (e) represent two different original artificial data sets (each with 15 points). Fig. 4 (b) and (f) show the respective Delaunay triangulations for these two original data sets. We can observe that different data yield different triangulations. Fig. 4 (c-d) and Fig. 4 (g-h) show the Delaunay triangulations after the removal of one and two points from the two respective data sets. It can be seen that the changes to these points affect only their adjacent triangles, while the overall structures of the Delaunay triangulation remain the same. Therefore, the Delaunay triangulations of images are, to some extent, robust and discriminative. The process of Delaunay triangulation is described in [37]. Obviously, the relationships between the feature points of an image are effectively encoded in the corresponding Delaunay triangulation; for this reason, we refer to these triangles as relationship triangles (RTs) in this study.

As described in the previous section, the second challenge is to eliminate the redundancy and mutual influences of the feature point relationships. To achieve this, we propose a graph-based sparse sampling method to analyze the relationships from both the local and global viewpoints. As shown in Fig. 5, we first map the RTs mesh M generated via Delaunay triangulation to a graph G as follows. Each RT of M is mapped to a vertex of G. If two RTs in the mesh M share common vertices, then the corresponding vertices in G are joined by an edge, the weight of which is computed as the Euclidean distance between the two vertices. Then, the local and global relationships can be extracted based on a sparse sampling strategy by searching for an independent set in the graph G and by calculating the graph Laplacian, respectively.

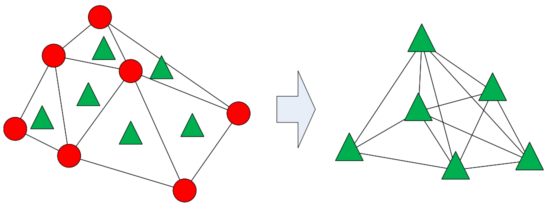


Fig. 5. Mapping from a Delaunay triangulation to a graph.

2) Local Structure Mining: Obviously, relationship pairs in local regions are captured by the RTs, so, we use the cosine of the largest angle in an RT to represent the local geometric structure. It is time-prohibitive to directly generate local relationships using all RTs for two main reasons. First, given the large number of RTs, the required computation time would be untenable. Second, each RT is, to some extent, influenced by or related to its neighbors. Therefore, a sparse sampling of RTs that do not share common points can eliminate the redundancy and mutual influences among RTs.

It is simple to conclude that finding a sparse sampling of the RTs is equivalent to finding an independent set in the graph G. It is known that finding the maximum independent set in a graph is an NP-complete problem. Therefore, in the proposed method, we simplify the problem by finding an independent set that is unique but may not be maximum. This process is described in Algorithm 1.

First, we present some definitions related to undirected graphs.

Let $G = (V, E)$ be a graph, where V and E are the vertex and edge sets, respectively.

Adjacency Matrix: A matrix $\mathbf{R} = \{r_{ij}\}$ is called the adjacency matrix of G if $r_{ij} = 1$ when $e_{ij} \in E$ and $r_{ij} = 0$ otherwise. Specially, the value of r_{ii} is 1.

Neighborhood: The neighborhood of a vertex v is defined as $N(v) = \{u \in V | e_{uv} \in E\}$.

Degree: The degree of v, denoted by $\deg(v)$, is defined as the number of neighbors of v.

Support: The support of v is defined as the sum of the degrees of all vertices that are adjacent to v. That is, $\text{support}(v) = \sum(\deg(u))$, where u denotes all vertices in $N(v)$.

Given an undirected graph $G = (V, E)$, we first construct its adjacency matrix and then find the vertex with the maximum degree, which we take as one element of the independent set. If more than one vertex has the maximum degree, then we compute the supports of these vertices and select the vertex with the maximum support as an element of the independent set. Then, we remove the rows and columns corresponding to these vertices, as well as their neighbors, from the adjacency matrix. This process is performed recursively until the adjacency matrix is empty, and thus, an independent set of the graph G is obtained. As indicated by the mapping between the graph G and the mesh of RTs, a sparse sampling of RTs is obtained, called the independent RT set. Finally, the cosine of the largest angle of each RT in the independent RT set is computed to represent the local structures.

The details of this process are shown in Algorithm 1, in which $\max(\bullet)$, $\min(\bullet)$, $\text{num}(\bullet)$, and $\text{len}(\bullet)$ denote the maximum, the minimum, the number of "$\bullet$", and the number of elements in "$\bullet$", respectively. Here, $\text{index}(f(v))$ returns the index of v, where $f(v)$ denotes the value of an operation on v, such as $\deg(v)$ or $\text{support}(v)$.

3) Global Relevance Mining: In this study, the global relevance represents the degree of global relatedness among all feature points, that is, how strongly the feature points relate to each other. As described above, each RT captures the relationship among three feature points in a local region. Thus, the global relevance can be regarded as the relatedness among all RTs. In general, if two RTs share common vertices, then they have a strong relevance. From the process of graph generation, we know that the edges of the graph indicate whether certain RTs share common vertices. Therefore, we can use the number of edges in the graph G as a measure of the global relevance.

Let $\varepsilon(G)$ be the total number of edges in the graph G. Then, we can state the following theorem:

Theorem 1: If $\varepsilon(G)$ and λ_i are the total number of edges in G and the i_{th} eigenvalue of the Laplacian matrix of G, respectively, then

$$\varepsilon(G) = \frac{1}{2}\sum_{i=1}^{n}\lambda_i \tag{1}$$

where n is the total number of eigenvalue. The Laplacian matrix is a matrix representation of G that is defined as

$$\mathbf{L} = \mathbf{D} - \mathbf{A} \tag{2}$$

Algorithm 1 Finding an Independent RT Set

```
**Input:** The adjacency matrix $\mathbf{R} = (r_{ij})$.
**Output:** A vector $\mathbf{y}$ consisting of elements that are the indices of elements in the independent RT set.
**Initialization:** $\mathbf{y} = [\ ]$.
**repeat**
    **for** $v_i \in V$ **do**
        $\deg(v_i) = \sum_j r_{ij}$
    **end for**
    $minDeg = \min(\deg(v_i))$
    **if** $\text{num}(\text{index}(minDeg)) > 1$ **then**
        **for** $v_k \in S$, where $S$ is a set, the degrees of the elements of which are equal to $minDeg$, **do**
            $\text{support}(v_k) = \sum \deg(u),\ u \in N(v_k)$
        **end for**
        $ms = \max(\text{support}(v_k))$, $minDeg\_maxsup = \text{index}(ms)$
    **else**
        $minDeg\_maxsup = \text{index}(minDeg)$
    **end if**
    $m \leftarrow minDeg\_maxsup$, $\mathbf{y} = \mathbf{y} \cup \{m\}$, $\mathbf{R}(m,:) = [\ ]$, $\mathbf{R}(:,m) = [\ ]$, $V = V - y$
**until** $\mathbf{R} = [\,]$
```

where $\mathbf{D} = diag(d_1, d_2, \cdots, d_n)$ is the degree matrix of the graph G, which is a diagonal matrix. The symbol d_i represents the number of edges connected to the i_{th} vertex of G, and the matrix $\mathbf{A}$ is the adjacency matrix of G.

We now prove Theorem 1. Given a matrix $\mathbf{C}$, if δ_i is an eigenvalue of the matrix, then according to the properties of matrices, we can write the following:

$$\sum_{i=1}^{n}\delta_i = tr(\mathbf{C}) \tag{3}$$

where $tr(\mathbf{C})$ is the trace of the matrix $\mathbf{C}$.

Recall that λ_i is an eigenvalue of the Laplacian matrix $\mathbf{L}$. Therefore,

$$\sum_{i=1}^{n}\lambda_i = tr(\mathbf{L}) = tr(\mathbf{D} - \mathbf{A}) = \sum_{i=1}^{n} d_i \tag{4}$$

Because each edge is used twice during the degree computation, from Eq. (4), we can conclude that

$$\varepsilon(G) = \frac{1}{2}\sum_{i=1}^{n} d_i = \frac{1}{2}\sum_{i=1}^{n}\lambda_i \tag{5}$$

Finally, we use $\varepsilon(G)$ to represent the global relevance. Obviously, the $\varepsilon(G)$ is always bigger than 1. To make the global relevance have a magnitude equal to the local structure, we normalize all the global relevance values of training data into the range (0, 1). The local structure and global relevance vectors are then concatenated to obtain the final image feature representation.

B. Between-Cluster Distance-Based Quantization

After obtaining the feature representation, we next use an unsupervised quantization strategy to learn binary fingerprint codes. As stated in Section I, existing quantization methods suffer from some limitations when using Hamming distances. The Manhattan quantization scheme presented in [38] overcomes these limitations to some extent. In Manhattan quantization, each feature dimension is first divided into K clusters, and q bits of natural binary code (NBC) are then used to encode the index of each cluster. The indices of the regions are randomly assigned by the clustering algorithm. However, the pairwise distances between different clusters also reflect the neighborhood structure among different samples in the corresponding dimension. Hence, the between-cluster distances should also be considered in the quantization. Therefore, in this study, we propose a strategy called BCDQ for learning binary codes.

The main difference between BCDQ and traditional Manhattan quantization is that we adopt a rule for index assignment to ensure that the most distant clusters are assigned by indices that are separated by the largest gap. The process of BCDQ is described as follows:

(1) Cluster center learning. Each dimension of all samples is divided into different clusters using the K-means clustering algorithm. If we wish to encode each dimension using q bits, then K is set to 2^q. Thus, K cluster centers are obtained. For a new sample, we compute the distances between the value of the new sample in each dimension and the cluster centers in the corresponding dimension, and we then assign the new sample to its corresponding cluster in each dimension in accordance with these distances.

(2) Cluster index assignment. We calculate the pairwise distances between clusters and identify the maximum distance and the two corresponding clusters. An index of 0 is assigned to the cluster that contains more samples between these two corresponding clusters. The index assignment rule for the remaining clusters is as follows: For the ith $(0 \leq i \leq K-1)$ cluster, find the nearest cluster to the i_{th} cluster that has not yet been assigned an index and set its index to $i+1$. The index assignment process in BCDQ is illustrated in Fig. 6.

(3) Quantization. In each dimension, the index of each cluster is encoded by using q binary bits, and these q bits are taken as the fingerprint in this dimension. The binary codes of each sample in all dimensions are concatenated to obtain the final fingerprint of the corresponding sample.

Here, we consider $q = 2$ as an example to demonstrate the process of BCDQ. In step 1, all samples are divided into four clusters in each dimension using the K-means method (or another clustering algorithm). Then, we apply the index assignment strategy described in step 2 to assign the corresponding indices to all clusters. For cluster indices of $\{0,1,2,3\}$, the corresponding binary codes are $\{00,01,10,11\}$. Finally, we concatenate the index codes of a sample in all dimensions to obtain the fingerprint of that sample.

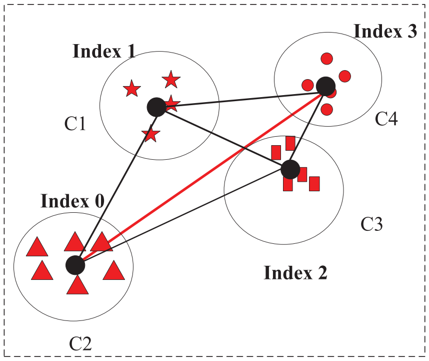


Fig. 6. Illustration of index assignment in BCDQ: The red line represents the longest pairwise distance between clusters; consequently, cluster C2, which contains more samples than cluster C4, is assigned an index of 0, and C1 is assigned an index of 1 because it is the nearest neighbor of C2. Indices are assigned to the other clusters following a similar rule.

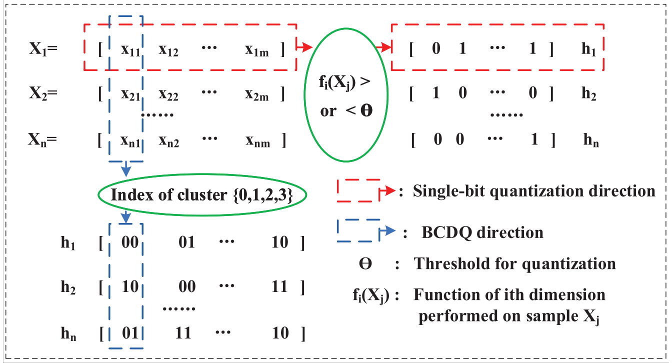


Fig. 7. Illustration of BCDQ and SQ: in BCDQ, each dimension is quantized for all samples at once, whereas in SQ, each sample is quantized for all dimensions at once.

Generally, a given dimension corresponds to the same attribute for all samples. Because the quantization of each dimension is performed on all of the training samples in that dimension, BCDQ fully considers the neighborhood structure among all samples in each dimension.

Figure 7 shows a comparison of BCDQ and SQ which is a commonly quantization strategy. In Fig. 7, $\mathbf{X} = (x_{ij})_{n\times m}$ is a feature matrix, where n and m are the number of samples and the number of feature dimensions, respectively. During the SQ process, as highlighted by the red rectangles, all of the feature dimensions of one sample are first projected by using a function f, and the values obtained via this projection are then compared against a threshold θ. The binary fingerprint value of x_{ij} is 1 if the value $f_i(x_{ij})$ is larger than θ; otherwise, the binary fingerprint value is 0. By contrast, BCDQ, as highlighted by the blue rectangles, is performed separately on each dimension for all samples. All samples are first divided into different clusters in each dimension, and the indices of the clusters are then encoded to obtain binary fingerprint values, as described above.

The advantage of BCDQ is that we cluster the samples in each dimension instead of a fixed threshold to generate binary fingerprint codes, and this clustering can preserve more neighbourhood structures. Furthermore, different dimensions

Algorithm 2 The Proposed Approach

Input: Image data set: training images and query image

Output: Binary image fingerprints of the images in the database and image copy results for the query image.

Offline Learning

(1) *Preprocessing*: Each image is normalized to a standard form;

(2) *Feature point extraction*: Various types of feature points, such as SIFT, SURF detectors and salient regions, are extracted;

(3) *Delaunay triangulation*: Delaunay triangulation is applied to each extracted feature point set;

(4) *Relationship mining*: The triangle mesh obtained via Delaunay triangulation for each image is first mapped to an undirected graph, and an independent set is then found from the graph using Algorithm 1. A sparse sampling of the RTs is obtained in accordance with this independent set. The local structures and global relevance are extracted from the RTs and graph Laplacian, and they are concatenated to obtain the final relationship vector, which is taken as the feature representation of the corresponding image;

(5) *BCDQ*: The final fingerprint vector of each image is generated via BCDQ.

Online Detection

(1) Preprocessing is first applied to the query image, Delaunay triangulation and relationship mining are then implemented on the query image to obtain its feature representation;

(2) In each feature dimension, the distances between the query image and the cluster centers of the database images in the corresponding dimension are calculated, and the query image is assigned to the correct cluster in each dimension based on these distances. Subsequently, the fingerprint of the query image is obtained via BCDQ;

(3) The distances between the fingerprint codes of query image and the database images are calculated to get image copies or near-duplicate images.

in the feature vector indicate different attributes. So the BCDQ strategy that clusters the samples in each dimension is more reasonable.

In summary, the proposed approach is summarized in Algorithm 2.

IV. Experiments

A. Experimental Setting

To illustrate the performance of the proposed algorithm, a combined database containing almost 100,000 images (including original images and eight distorted versions for each original image) is employed to evaluate the robustness and discrimination performance. The original images are collected from the UCID image database [39], ImageNet [40] and the Internet [18]. For each original image, a total of eight modifications/attacks are applied: (1) rotation, (2) Additive Gaussian White Noise (AGWN), (3) blurring, (4) contrast enhancement, (5) letterboxing, (6) logo/caption insertion, (7) picture in picture, and (8) JPEG compression. Some of the modifications are shown in Fig. 8. The descriptions and parameters of the various modifications are provided in Table I.

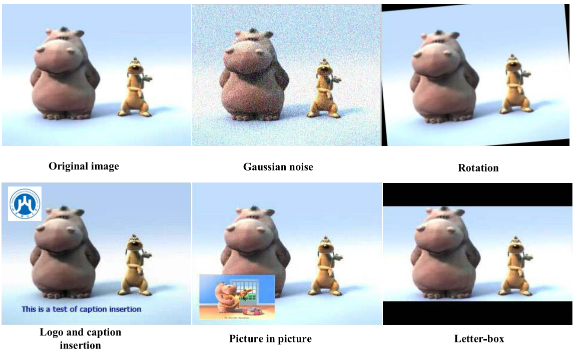


Fig. 8. The modifications to the original image.

TABLE I

Descriptions of Modifications

Attack	Parameter Settings
Rotation	3 or 5 deg counterclockwise
AGWN	noise variance of 0.004 or 0.006
Salt-and-pepper noise	noise density of 0.01 or 0.03
Blurring	motion blur, 10 pixels
JPEG compression	quality factor of 50 or 70
Letterboxing	10% of pixels are replaced with black boxes
Caption insertion	insert a line of text at the bottom
Picture in picture	insert a different picture, size 50 x 50

In these experiments, we use SURF points as the feature points in the proposed approach and compare the proposed approach to several state-of-the-art approaches. They include

- Discrete fourier transform (DFT)-based method [23]: This method first regularizes the input image based filtering, and then the secondary image is obtained by the rotation projection. The robust frequency feature is extracted from the secondary image after discrete Fourier transform to generate image fingerprints.
- Gabor filtering and dithered lattice vector quantization (GF-DLVQ)-based method [41]: This method generates image fingerprints based on random Gabor filtering and dithered lattice vector quantization (LVQ).
- Nonnegative matrix factorization (NMF)-based method [42]: This method considers the image as a matrix, and uses a randomized dimensionality reduction that retains the essence of the original image matrix to generate image fingerprints.
- Ring partition and NMF (Ring-NMF)-based method [24]: This method constructs an rotation-invariant secondary image using ring partition and NMF to generate image fingerprints.
- Ring partition and invariant vector distance (Ring-IVD)-based method [12]: This method generates image fingerprints using ring partition and invariant vector distance.

We also compare the proposed method with its one variant:

- SURF: We only use feature point descriptors SURF with BCDQ to generate image fingerprints.

To demonstrate the advantages and scalability of the proposed relationship-based strategy, we also generated image fingerprints using other local feature point, i.e., the SIFT points, and show the comparison result between the proposed relationship-based strategy and the feature point-based method under different modifications.

For binary fingerprint algorithms, such as the proposed approach and the DFT-based method, we use the normalized Hamming distance to measure similarity, which is defined as

$$D(\mathbf{h}_1, \mathbf{h}_2) = \frac{1}{N}\sum_{\mathrm{i}=1}^{N} |\mathbf{h}_1(i) - \mathbf{h}_2(i)| \qquad (6)$$

where $\mathbf{h}_1$ and $\mathbf{h}_2$ are the fingerprint vectors of two images and N is the fingerprint length. Usually, the value of the normalized Hamming distance between two fingerprint vectors is inversely proportional to the similarity of the images.

For the methods that output real-value fingerprints, we use the correlation-coefficient-based distance metrics, as suggested in [42] and [24], which is

$$S(\mathbf{h}_1, \mathbf{h}_2) = \frac{\sum_{i=1}^{N} (\mathbf{h}_1(i) - \mu_1)(\mathbf{h}_2(i) - \mu_2)}{\sqrt{\sum_{i=1}^{N} (\mathbf{h}_1(i) - \mu_1)^2}\sqrt{\sum_{i=1}^{N} (\mathbf{h}_2(i) - \mu_2)^2}} \qquad (7)$$

where μ_1 and μ_2 are the means of the fingerprint vectors $\mathbf{h}_1$ and $\mathbf{h}_2$, respectively. Obviously, the value of the correlation coefficient between two fingerprint vectors is proportional to the similarity between the corresponding images.

To evaluate the overall performance including robustness and discrimination, the F-score is used in the experiments which is a combined metric. It is defined as follows:

$$F_\beta = (1 + \beta^2)\frac{P_p * P_r}{\beta^2 P_p + P_r} \qquad (8)$$

where P_p and P_r are the precision and recall rates, respectively, and β is a parameter that defines how much weight should be given to recall versus precision. We set $\beta = 1$ in these experiments. A higher F-score indicates better algorithm performance.

To get the experimental results of the correlation coefficient or Hamming distance, such as in Fig. 11 and Table III, we compute the correlation coefficient or Hamming distance of each pair for all queries, and use the mean value as the final results. When compute the precision and recall, such as Fig. 12, we carry out the experiments for three times, and in each time, we randomly select 10% of images as queries, and then use them to search in the dataset that is composed of all the remaining images. Finally, we obtain the mean precision and recall over all experiments. In general, each image in the dataset has the equal possibility to be selected as query or be remained as searched samples.

A fix length of fingerprint is beneficial to image copy detection. Therefore, the length of the fingerprint vector is set to 256 in the proposed method. To obtain a fix length of fingerprint, we first normalize the length of feature vector in the process of relation mining to a fix length 128, and then quantize each dimension of feature vector to 2 bits in BCDQ (In the process of BCDQ, we set q to 2, which archives better performance (its reason will be described in Section IV-D), i.e., each dimension of the feature vector is quantized to 2 bits, so the length of the final fingerprint vector is 256. We should point out that the fixed length of the final fingerprint vector is set empirically by experiments. We will try to explore more efficient strategies to set the fixed length of the fingerprint vector both in relation mining and BCDQ in a future study.

TABLE II

SYMBOL NOTATIONS

Symbol	Definition
w	Normalized Hamming distance between two fingerprints of visually different images
v	Normalized Hamming distance between two fingerprints of visually similar image
τ	Threshold
$P(\bullet)$	Distribution of " $\bullet$ "

B. Threshold Analysis

Given a query image fingerprint sequence, we declare it to be a copy of an existing image if the normalized Hamming distance or correlation coefficient between these two fingerprint sequences is below or above a certain threshold, respectively. The threshold selection for the normalized Hamming distance is described in this subsection, and the same strategy can be used to select the threshold for the correlation coefficient.

In real applications, a robust image fingerprint system should be able to determine whether a query image is a modified copy. The common method is to set a threshold τ in advance. Let $\mathbf{F}(\bullet)$ be the fingerprint vector of "$\bullet$", a decision is reached as follows[1]:

$$\begin{cases} H_1: & D(\mathbf{F}(V^i), \mathbf{F}(V^j)) \leq \tau \quad V^i \; and \; V^j \; are \; similar \\ H_2: & D(\mathbf{F}(V^i), \mathbf{F}(V^j)) > \tau \quad V^i \; and \; V^j \; are \; different \end{cases} \qquad (9)$$

where $D(\bullet)$ is a distance metric. A smaller threshold can improve the true positive probability but will negatively affect the miss probability. By contrast, a larger threshold results in a lower miss probability but may cause the false alarm probability to be higher. Therefore, the threshold choice should be carefully considered in a real system. Before analyzing the choice of this threshold, we first list some symbol notations in Table II.

In our method, we model H_1 and H_2 as a Rayleigh distribution and a normal distribution, respectively. It is reasonable to assume that H_2 yields i.i.d. (independent and identically distribution), because the distances between pairs of fingerprints for visually different images are independent of each other. Therefore, $w = D(\mathbf{F}(V^i), \mathbf{F}(V^j))$ is modeled as

$$P(w) \sim N(\mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(w-\mu)^2}{2\sigma^2}} \qquad (10)$$

However, when considering the distances between visually similar images, for example, assuming V_A^1 and V_B^1 to be two copies of V^1, the distances $D(\mathbf{F}(V^1), \mathbf{F}(V_A^1))$ and $D(\mathbf{F}(V^1), \mathbf{F}(V_B^1))$ are not completely independent of each

[1] Note that the relational operators in Eq. (9) are reversed when determining the threshold for the correlation coefficient.

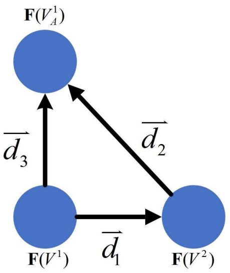


Fig. 9. Illustration of fingerprint distance.

other because both of them have some relationship with the image V^1. Therefore, the model of normal distribution is not suitable, to some extent. In this study, we model H_1 as a Rayleigh distribution. Fig. 9 illustrates the motivation. Assuming $\mathbf{F}(V^1)$ and $\mathbf{F}(V^2)$ are the fingerprints calculated from different images V^1 and V^2, $\mathbf{F}(V_A^1)$ and $\mathbf{F}(V^1)$ are the fingerprints calculated from visually similar images V_A^1 and V^1, respectively. We define $\overline{d_1} = \mathbf{F}(V^2) - \mathbf{F}(V^1)$, $\overline{d_2} = \mathbf{F}(V_A^1) - \mathbf{F}(V^2)$ and $\overline{d_3} = \mathbf{F}(V_A^1) - \mathbf{F}(V^1)$. Obviously, $\overline{d_1}$ and $\overline{d_2}$ are the distance vector of visually different images which yields normal distribution. The value $\upsilon = ||\overline{d_3}|| = D(\mathbf{F}(V^1) - \mathbf{F}(V_A^1)) = ||\overline{d_1} + \overline{d_2}||$ is the distance between two visually similar images. According to the definition of Rayleigh distribution (that is: The envelope of a variable which is the sum of two independent normal random signals obey Rayleigh distribution), the variable υ can be approximately modeled as Rayleigh distribution, and it is modeled in Eq.(11).

$$P(\upsilon) \sim \frac{\upsilon}{\sigma^2} e^{-\frac{\upsilon^2}{2\sigma^2}} \tag{11}$$

We determine some statistics about the distance values w, which are the distances between visually different images, and the values υ, which are the distances between visually similar images, and then find that the distributions of the real data approximately consist of the normal and Rayleigh distributions from Fig. 10.

According to the model assumptions, a suitable threshold can be obtained by solving the following equation:

$$\frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}} = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{12}$$

Eq. (12) can be rewritten step by step as follows:

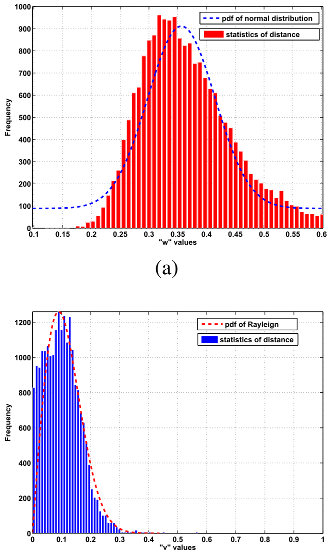
$$\frac{x}{\sigma} e^{-\frac{x^2}{2\sigma^2}} = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{13}$$

$$\ln x - \ln \sigma - \frac{x^2}{2\sigma^2} = -\ln \sqrt{2\pi} - \frac{(x-\mu)^2}{2\sigma^2} \tag{14}$$

$$\ln x - \frac{x^2}{2\sigma^2} + \frac{(x-\mu)^2}{2\sigma^2} = \ln \sigma - \ln \sqrt{2\pi} \tag{15}$$

$$\ln x = \frac{\mu}{\sigma^2} x + \ln \frac{\sigma}{\sqrt{2\pi}} - \frac{\mu^2}{2\sigma^2} \tag{16}$$

The threshold τ is given by the solution to Eq. (16). We adopt the notations $f(x) = \ln x$ and $g(x) = \frac{\mu}{\sigma^2} x + \ln \frac{\sigma}{\sqrt{2\pi}} - \frac{\mu^2}{2\sigma^2}$; then, the solution to Eq. (16) is the intersection of these two functions in the axis space.



(a)

(b)

Fig. 10. Statistics of distance values and their fit distributions: (a) distance of visually different images (w values); (b) distance of visually similar images (υ values).

To obtain an approximate solution to Eq. (16), we first estimate the values of μ and σ. For the Gaussian normal distribution $N(\mu, \sigma)$, the maximum likelihood estimations of μ and σ are given by the mean and variance, respectively, of the samples:

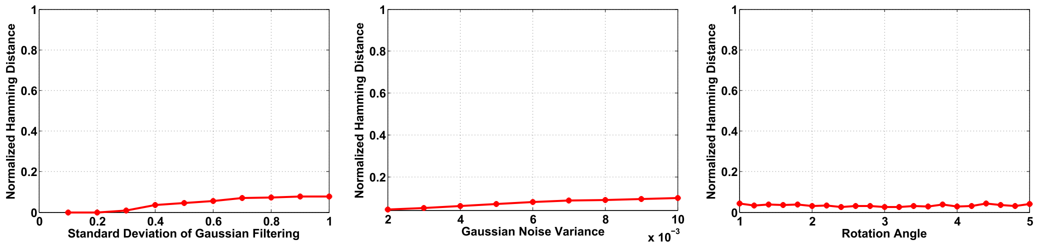
$$\widehat{\mu} = \frac{1}{N} \sum_{i=1}^{N} x_i = \overline{x}, \quad \widehat{\sigma^2} = \frac{1}{N} \sum_{i=1}^{N} (x_i - \overline{x})^2 \tag{17}$$

Once the approximate values of μ and σ have been obtained, Eq. (16) can be solved using the Newton-iterative method. Thus, the threshold for the normal Hamming distance is set to 0.15 in the proposed method.

C. Robustness Evaluation

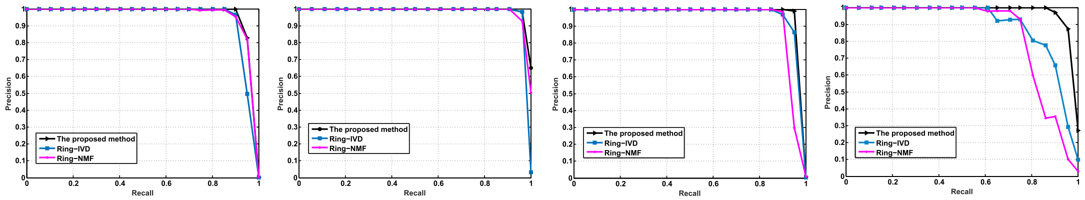
As previously mentioned, relationship mining can be used to improve the robustness of an image fingerprinting system, especially against classical attacks that would alter the positions of feature points, such as additive noise and blurring. Therefore, we first illustrate the performance of the proposed method against such attacks with various parameter settings in Fig. 11. We observe that under most of the attacks, the normalized Hamming distances are below 0.1, indicating that the proposed method achieves good robustness against such attacks. We also evaluate the precision and recall performance in the dataset under four modifications, and compare the performance of the proposed with two related works called Ring-IVD and Ring-NMF, which are shown in Fig. 12. It can also be seen that the proposed method achieves the best retrieval performance in these three methods.

Table III (where Qf, Sd, Nv, Nd, and Ra denote the quality factor, standard deviation, noise variance, noise density, and rotation angle, respectively) presents a comparison of the average normalized Hamming distances for the proposed method



(a) (b) (c)

Fig. 11. Normalized Hamming distance performance of the proposed strategy: (a) Blurring with different Gaussian filters, (b) Gaussian noise, and (c) Rotation.



(a) (b) (c) (d)

Fig. 12. PR performance of the proposed strategy: (a) Blurring with Gaussian, (b) Rotation 3 degree, (c) Contrast, and (d) Caption insertion.

TABLE III

NORMALIZED HAMMING DISTANCE STATISTICS

Modification type	Proposed	DFT-based
JPEG compression (Qf=50)	0.0227	0.0605
JPEG compression (Qf=70)	0.0234	0.0445
Blur (Sd=0.6)	0.0467	0.0498
Blur (Sd=1.0)	0.0632	0.0701
Additive Gaussian noise (Nv=0.004)	0.0514	0.1531
Additive Gaussian noise (Nv=0.006)	0.0763	0.1759
Salt-and-pepper noise (Nd=0.01)	0.0381	0.1316
Salt-and-pepper noise (Nd=0.03)	0.0872	0.2083
Image rotation (Ra=3)	0.0227	0.1253
Image rotation (Ra=5)	0.0305	0.1506
Image rotation (Ra=90)	0.0519	0.2018
Flipping	0.0532	0.1776
Caption insertion	0.0898	/
Letterboxing	0.0514	/
Picture in picture	0.0728	/

and the DFT-based method. From this table, we observe that the proposed method is more robust than the DFT-based method against most of these content-preserving operations. Image post-processing techniques such as caption insertion and letterboxing are also commonly encountered manipulations as a result of the proliferation of image editing software, and they are also standard transformations in TRECVID for content-based copy detection tasks [43]. In general, the fundamental image contents remain unchanged following such post-processing, and therefore, image fingerprinting is expected to be robust to these modifications as well. The normalized Hamming distances between the original and altered versions of images under these modifications are also shown in Table III, from which it can be seen that the values of the normalized Hamming distance are below the threshold, meaning that the proposed method demonstrates the desired robustness to post-processing.

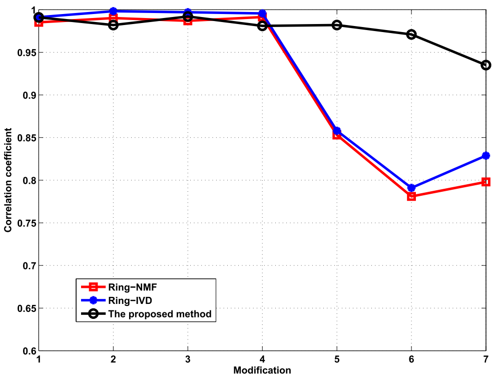


Fig. 13. Performance comparison with the ring-partition-based method: 1. rotation, 2. Additive Gaussian White Noise (AGWN), 3. Blur, 4. Contrast, 5. Letterboxing, 6. Logo/caption insertion, 7. picture in picture.

We also compared the proposed method with the Ring-NMF and Ring-IVD methods. Following these two methods, we use the correlation coefficient between the original and modified fingerprint vectors as the metric to evaluate the algorithm performance. Fig. 13 presents a comparison between the three methods for different image modification instances. The performance of the proposed method matches that of the ring-partition-based method under most modifications. However, when an image is subjected to certain post-processing manipulations, such as caption insertion, letterboxing, or picture in picture, the proposed method achieves superior performance.

In general, both the DFT-based and ring-partition-based methods generate image fingerprints based on local pixel points. The proposed method achieves better performance

TABLE IV
F-Score Statistics of Different Methods

Attacks	Proposed	GF-DLVQ	NMF	Ring-NMF
Noise	0.9847	0.9615	0.8742	0.9434
Rotation	0.9902	0.9853	0.8982	0.9851
Blur	0.9714	0.9308	0.8679	0.9213
JPEG compression	0.9947	0.9198	0.9042	0.9532
Postprocessing	0.9677	0.8915	0.8742	0.8731

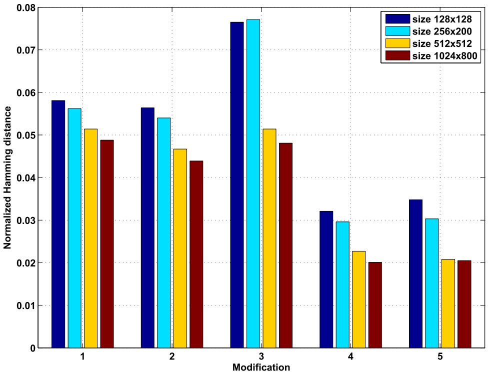


Fig. 14. Normalized Hamming distance for different normalization under different modifications: 1. AWGN, 2. Blur, 3. Letterbox, 4. JPEG compression, 5. Contrast.)

because the original structure of the relationships among the feature points remains unchanged to some degree following the modifications, whereas the pixel information employed in these two existing methods changes significantly.

Table IV compares the F-scores of the four different methods under different attacks. The post-processing represents caption or logo insertion, letterbox and picture in picture. In comparison with the other methods, the proposed method achieves a higher F-score, which indicates better robustness and discrimination performance. Especially, the performance achieves more than 5% improvement under some attack that would alter the position of feature points.

D. Effect of Different Components

Evidently, the preprocessing, relationship harnessing and BCDQ are three components of the proposed method. Therefore, we evaluate the different effects of these three parts in this subsection.

1) Effect of Preprocessing: In most of the existing methods, to alleviate effects of commonly-used digital manipulations to images, such as resizing and color space changing, the normalization is commonly exploited to produce a normalized image. In addition, the preprocessing can also make the local feature extraction more efficient. In our work, we use bi-linear interpolation for this process.

To show the effect of the preprocessing, we carry out two types of experiments. One is to evaluate the robustness with different parameter settings of the preprocessing, and the results are shown in Fig. 14; the other type of experiment is to show the robustness of the proposed method with and without the preprocessing, and the results are shown in Table V.

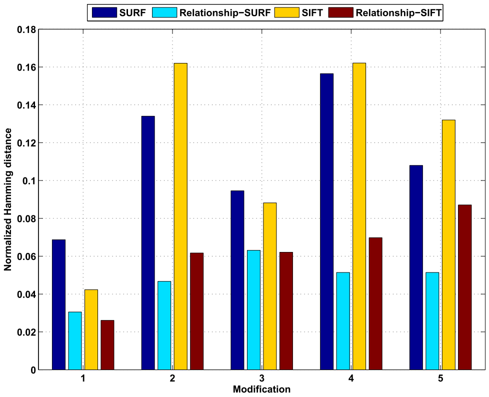


Fig. 15. Performance comparison between feature point relationships and individual feature points (1. Rotation, 2. Blur, 3. Contrast, 4. AWGN, and 5. Letterbox. SURF: the method only use SURF descriptor; SIFT: the method only uses SIFT descriptor; Relationship-SURF: The proposed method based on SURF; Relationship-SIFT: The proposed method based on SIFT.)

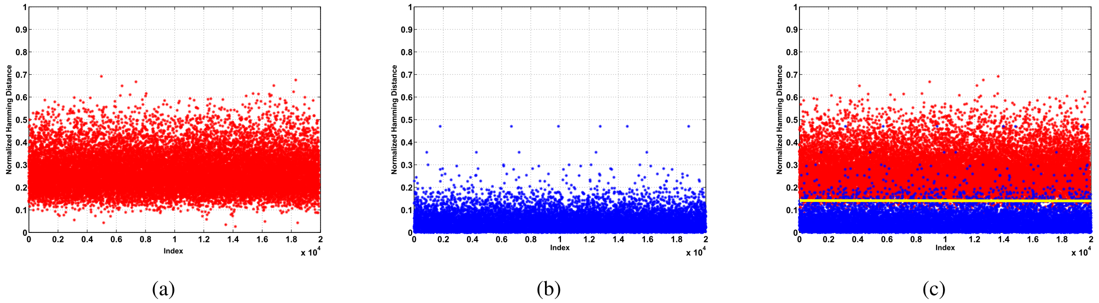
TABLE V
The Performance With and Without Preprocessing

Cases	Noise	Rotation	JPEG compression	Letterboxing
With preprocessing	0.0531	0.0298	0.0189	0.0449
Without preprocessing	0.0589	0.0201	0.0193	0.0397

Fig. 14 shows the performance under different normalized image sizes (128 × 128, 256 × 200, 512 × 512, 1024 × 800), and we can see that the performance is similar under these different sizes, although the performance will be slight better when we use larger size of normalized image, and this is reasonable because the image sizes in the dataset are usually larger than the normalized size. However, we should point out that the larger size will cause more computations in fingerprint generation to some extent. Therefore, in this study, we used the size 512 × 512 for the tradeoff.

Although the preprocessing can render the other steps in fingerprint generation more efficient, it may lead to some changes about the robustness of image fingerprint. Therefore, in the second type of experiments, we conduct some extend experiment on a public dataset UCID with and without preprocessing to evaluate whether the robustness is changed a lot after we applied the preprocessing to the images. The normalized Hamming distances under different modifications are shown in Table V, and we can see that the performance with preprocessing is similar with the one without preprocessing. In general, we can conclude that the preprocessing will not reduce the entire robustness of the proposed method, and it also benefits to other steps.

2) Effect of Relationship Harnessing: Compared with the feature point descriptors themselves, the relationships between them are more stable. To demonstrate the advantages of the relationship-based strategy, we also generated image fingerprints using two commonly used local feature points, i.e., the SURF and SIFT points, and show the comparison



(a)

(b)

(c)

Fig. 16. Normalized Hamming distances: (a) distances between visually different images; (b) distances between visually similar images; (c) overlap of these two types of distances with the yellow line as a threshold selection.

TABLE VI

F-Score Statistics of BCDQ

Attacks	1-BCDQ	2-BCDQ	3-BCDQ	4-BCDQ
Rotation	0.9353	0.9902	0.9578	0.9415
Gaussian Noise	0.9541	0.9875	0.9232	0.9434
Blur	0.9058	0.9714	0.9197	0.9286
Contrast enhancement	0.9317	0.9809	0.9690	0.9471
Letterboxing	0.9315	0.9767	0.9784	0.9731
Logo/caption insertion	0.9051	0.9518	0.9501	0.9413
Pic. in pic.	0.9085	0.9729	0.9672	0.9601
JPEG compression	0.9303	0.9947	0.9714	0.9578

results between the proposed relationship-based strategy and the feature point-based methods under different modifications in Fig. 15. We can see that the performance under these modifications is improved with the relationship-based approach, illustrating the advantage of the proposed strategy. In addition, it can be also seen that the performance of the proposed relationship-based strategy is positively related to the one of the corresponding feature descriptor.

3) Effect of BCDQ: To evaluate the effect of BCDQ, we carry out the experiments with different values of q, and we denote them as q-BCDQ. The experimental results are shown in Table VI. It can be seen that 2-BCDQ achieves better performance than others under most of the modifications. Therefore, we set q to 2 in the experiments. This phenomenon has also been observed by the works in [38] and [25]. We can also know that the multiple bit coding in each dimension would lead to a better performance than the single-bit case, however, the performance is not always better with the code length growing longer. The potential reason for this phenomenon is that a longer code may contain more noise.

E. Discrimination Analysis

To evaluate the performance in terms of discrimination, we first randomly select 200 original images with different contents and compute the normalized Hamming distances between pairs of images (distances between visually different images). Ultimately, approximately 19,900 normalized Hamming distance values are obtained, as shown in Fig. 16(a). Then, we randomly select approximately 19,900 pairs of similar images (including original images and its copies) and compute the normalized Hamming distances between each pair of similar images, which is shown in Fig. 16(b). In addition, we also carry out the experiment to overlap these two case of distances, and the Fig. 16(c) illustrates the threshold (in this case, there are about 0.95% and 1.81% red and blue crosses going in the wrong decision field, respectively), and we can see that there is a boundary (around 0.15) separating these two types of normalized Hamming distances, which indicates that the proposed method shows effective discrimination performance.

TABLE VII

Comparison of the Execution Time for One Image (Unit: Second)

Method	Proposed	DFT	Ring-NMF	Ring-IVD
Average time	0.3152	0.1577	0.0173	0.2176

F. Complexity Evaluation

Complexity is also important for image fingerprinting system. In the proposed method, the experiments are conducted on a computer with an Intel Core i7-6700 3.40 GHz 4 processor and 16 GB RAM. The operating system is 64-bit Windows 10 and the programming environment is MATLAB R2015b. Generally, the main computational time of the proposed method is consumed in finding the independent set in Algorithm 1 because of the large adjacency matrix. The mean time used in obtaining the independent RT set for one image is approximately 0.2 seconds, which makes the whole system consume more time than most of the existing methods (the comparison of average time for one image is shown in Table VII). However, most of images are performed during the offline training. Therefore, the complexity is still acceptable. We admit it is a limitation of our method, and we will find more efficient strategy to accelerate the proposed method in the future study.

V. Conclusion

In this study, we have proposed a novel image fingerprinting method based on the quantization of local and global feature point relationships. These relationships are obtained based on an independent set and the Laplacian matrix generated via Delaunay triangulation and a graph model. After obtaining the feature representation of an image, we use an unsupervised quantization strategy called BCDQ to generate the image fingerprint. Unlike existing algorithms, the proposed method not only employs information on local features but also harnesses the relationships between feature points. In addition, the proposed quantization approach can effectively preserve the neighborhood structure in each dimension of the feature vector. We used SURF points as an example to experimentally demonstrate the performance of the proposed strategy. However, the proposed method could be similarly applied in other feature-point-based image fingerprinting systems.

Nevertheless, some open questions remain to be investigated in future work. The first issue we should consider is the different influences of local and global relationships in an image fingerprinting system. To this end, an adaptive bit assignment strategy for local and global relationships should be studied in future work. In addition, we will investigate new effective methods for decreasing the time complexity of finding independent RTs, because the proposed Algorithm 1 may become time consuming when the size of the adjacency matrix is large.

Acknowledgments

We thank Dr. Yuenan Li (Tianjin University) for providing us some of database and their codes for image modifications.

References

[1] J. Song, L. Gao, L. Liu, X. Zhu, and N. Sebe, "Quantization-based hashing: A general framework for scalable image and video retrieval," *Pattern Recognit.*, vol. 75, pp. 175–187, Mar. 2018.

[2] J. Wang, T. Zhang, J. Song, N. Sebe, and H. T. Shen, "A survey on learning to hash," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.

[3] L. Gao, Y. Wang, D. Li, J. Song, and J. Song, "Real-time social media retrieval with spatial, temporal and social constraints," *Neurocomputing*, vol. 253, pp. 77–88, Aug. 2017.

[4] L. Zhu, J. Shen, L. Xie, and Z. Cheng, "Unsupervised topic hypergraph hashing for efficient mobile image retrieval," *IEEE Trans. Cybern.*, vol. 47, no. 11, pp. 3941–3954, Nov. 2017.

[5] J. Tang, L. Jin, Z. Li, and S. Gao, "RGB-D object recognition via incorporating latent data structure and prior knowledge," *IEEE Trans. Multimedia*, vol. 17, no. 11, pp. 1899–1908, Nov. 2015.

[6] A. Wang, J. Lu, J. Cai, T.-J. Cham, and G. Wang, "Large-margin multi-modal deep learning for RGB-D object recognition," *IEEE Trans. Multimedia*, vol. 17, no. 11, pp. 1887–1898, Nov. 2015.

[7] C.-M. Pun, C. Yan, and X.-C. Yuan, "Image alignment-based multi-region matching for object-level tampering detection," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 2, pp. 377–391, Feb. 2017.

[8] C.-P. Yan and C.-M. Pun, "Multi-scale difference map fusion for tamper localization using binary ranking hashing," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 9, pp. 2144–2158, Sep. 2017.

[9] X. Lv and Z. J. Wang, "Perceptual image hashing based on shape contexts and local feature points," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 1081–1093, Jun. 2012.

[10] R. Davarzani, S. Mozaffari, and K. Yaghmaie, "Perceptual image hashing using center-symmetric local binary patterns," *Multimedia Tools Appl.*, vol. 75, no. 8, pp. 4639–4667, 2016.

[11] F. Zou, Y. Chen, J. Song, K. Zhou, Y. Yang, and N. Sebe, "Compact image fingerprint via multiple kernel hashing," *IEEE Trans. Multimedia*, vol. 17, no. 7, pp. 1006–1018, Jul. 2015.

[12] Z. Tang, X. Zhang, X. Li, and S. Zhang, "Robust image hashing with ring partition and invariant vector distance," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 1, pp. 200–214, Jan. 2016.

[13] X. Nie, J. Liu, J. Sun, L. Wang, and X. Yang, "Robust video hashing based on representative-dispersive frames," *Sci. China Inf. Sci.*, vol. 56, no. 6, pp. 1–11, 2013.

[14] M. Li and V. Monga, "Desynchronization resilient video fingerprinting via randomized, low-rank tensor approximations," in *Proc. IEEE 13th Int. Workshop Multimedia Signal Process. (MMSP)*, Oct. 2011, pp. 1–6.

[15] J. Zhang, W. Zhu, B. Li, W. Hu, and J. Yang, "Image copy detection based on convolutional neural networks," in *Proc. Chin. Conf. Pattern Recognit.*, 2016, pp. 111–121.

[16] S. Battiato, G. M. Farinella, E. Messina, and G. Puglisi, "Robust image alignment for tampering detection," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 4, pp. 1105–1117, Aug. 2012.

[17] A. Joly, O. Buisson, and C. Frélicot, "Content-based copy retrieval using distortion-based probabilistic similarity search," *IEEE Trans. Multimedia*, vol. 9, no. 2, pp. 293–306, Feb. 2007.

[18] Y. Li and P. Wang, "Robust image hashing based on low-rank and sparse decomposition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 2154–2158.

[19] Z. Zhou, Y. Wang, Q. J. Wu, C.-N. Yang, and X. Sun, "Effective and efficient global context verification for image copy detection," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 1, pp. 48–63, Jan. 2017.

[20] Z. Li, G. Liu, H. Jiang, and X. Qian, "Image copy detection using a robust Gabor texture descriptor," in *Proc. 1st ACM Workshop Large-Scale Multimedia Retr. Mining*, 2009, pp. 65–72.

[21] G. Yang, N. Chen, and Q. Jiang, "A robust hashing algorithm based on SURF for video copy detection," *Comput. Secur.*, vol. 31, no. 1, pp. 33–39, 2012.

[22] X. Nie, Y. Chai, J. Liu, J. Sun, and Y. Yin, "Spherical torus-based video hashing for near-duplicate video detection," *Sci. China Inf. Sci.*, vol. 59, no. 5, p. 059101, 2016.

[23] C. Qin, C.-C. Chang, and P.-L. Tsou, "Robust image hashing using non-uniform sampling in discrete Fourier domain," *Digit. Signal Process.*, vol. 23, no. 2, pp. 578–585, 2013.

[24] Z. Tang, X. Zhang, and S. Zhang, "Robust perceptual image hashing based on ring partition and NMF," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 3, pp. 711–724, Mar. 2014.

[25] W. Liu, J. Wang, S. Kumar, and S.-F. Chang, "Hashing with graphs," in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*, 2011, pp. 1–8.

[26] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.

[27] Y. Cao, H. Zhang, Y. Gao, and J. Guo, "An efficient duplicate image detection method based on Affine-SIFT feature," in *Proc. 3rd IEEE Int. Conf. Broadband Netw. Multimedia Technol. (IC-BNMT)*, Oct. 2010, pp. 794–797.

[28] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 3304–3311.

[29] S.-A. Berrani, L. Amsaleg, and P. Gros, "Robust content-based image searches for copyright protection," in *Proc. 1st ACM Int. Workshop Multimedia Databases*, 2003, pp. 70–77.

[30] W. Kong and W.-J. Li, "Double-bit quantization for hashing," in *Proc. AAAI*, 2012, pp. 634–640.

[31] S. Moran, V. Lavrenko, and M. Osborne, "Variable bit quantisation for LSH," in *Proc. ACL*, 2013, pp. 753–758.

[32] Z. Wang, L.-Y. Duan, J. Lin, X. Wang, T. Huang, and W. Gao, "Hamming compatible quantization for hashing," in *Proc. IJCAI*, 2015, pp. 2298–2304.

[33] Y. Kalantidis, L. G. Pueyo, M. Trevisiol, R. van Zwol, and Y. Avrithis, "Scalable triangulation-based logo recognition," in *Proc. 1st ACM Int. Conf. Multimedia Retr.*, 2011, p. 20.

[34] F. P. Preparata and M. I. Shamos, *Computational Geometry: An Introduction*. New York, NY. USA: Springer, 2012.

[35] D.-T. Lee and A. K. Lin, "Generalized Delaunay triangulation for planar graphs," *Discrete Comput. Geometry*, vol. 1, no. 1, pp. 201–217, 1986.

[36] V. Rajan, "Optimality of the Delaunay triangulation in $\mathbb{R}^d$," *Discrete & Comput. Geometry*, vol. 12, no. 2, pp. 189–202, 1994.

[37] V. J. D. Tsai, "Delaunay triangulations in TIN creation: An overview and a linear-time algorithm," *Int. J. Geograph. Inf. Sci.*, vol. 7, no. 6, pp. 501–524, 1993.

[38] W. Kong, W.-J. Li, and M. Guo, "Manhattan hashing for large-scale image retrieval," in *Proc. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2012, pp. 45–54.

[39] G. Schaefer and M. Stich, "UCID: An uncompressed color image database," *Proc. SPIE*, vol. 5307, pp. 472–480, Dec. 2003.

[40] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 248–255.

[41] Y. Li, Z. Lu, C. Zhu, and X. Niu, “Robust image hashing based on random Gabor filtering and dithered lattice vector quantization,” *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1963–1980, Apr. 2012.

[42] V. Monga and M. K. Mihçak, “Robust and secure image hashing via non-negative matrix factorizations,” *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 3, pp. 376–390, Sep. 2007.

[43] G. Awad, P. Over, and W. Kraaij, “Content-based video copy detection benchmarking at TRECVID,” *ACM Trans. Inf. Syst.*, vol. 32, no. 3, 2014, Art. no. 14.

Xiushan Nie (M’12) received the Ph.D. degree from Shandong University, Jinan, China, in 2011. He is currently a Professor with the Shandong University of Finance and Economics, Jinan, China. From 2013 to 2014, he was a Visiting Scholar at the University of Missouri-Columbia, USA. His research interests include data mining, multimedia retrieval, and indexing and computer vision.

Xiaoyu Li received the bachelor’s degree in computer science and technology from the Shandong University of Finance and Economics, Jinan, where she is currently pursuing the M.Sc. degree. Her research interests include machine learning and multimedia information processing.

Yane Chai received the bachelor’s degree in electronic information engineering, Shandong Yingcai University, Jinan. She is currently pursuing the M.Sc. degree in Shandong University of Finance and Economics, Jinan, China. Her research interests include multimedia information processing and multimedia retrieval.

Chaoran Cui received the B.S. degree in software engineering and the Ph.D. degree from the Shandong University, Jinan, China, in 2010 and 2015, respectively. He was a Research Fellow at Singapore Management University from 2015 to 2016. He is currently a Professor with the School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan, China. His research interests include information retrieval, analysis and understanding on multimedia information, and computer vision.

Xiaoming Xi received the Ph.D. degree from Shandong University in 2015. He is currently a Lecturer with the Shandong University of Finance and Economics. His research interests include machine learning, data mining, and medical image analysis.

Yilong Yin received the Ph.D. degree from Jilin University, Changchun, China, in 2000. He is currently the Director of the Machine Learning and Applications Group and a Professor with Shandong University, Jinan, China.

From 2000 to 2002, he was a Post-Doctoral Fellow with the Department of Electronic Science and Engineering, Nanjing University, Nanjing, China. His research interests include machine learning, data mining, computational medicine, and biometrics.