An Image Fingerprinting Method Robust to Complicated Modifications

Xinghua Yu [1], Tiejun Huang [2]
[1] *Graduate University of Chinese Academy of Science, Beijing, 100080, China*
[2] *Institute of Digital Media, Peking University, Beijing 100871, China*
E-mail{xhyu, tjhuang}@jdl.ac.cn

Abstract

Image fingerprinting technique plays an important role in Digital Rights Management and related applications, especially in the detection of the illegal use of image works. Although existing image fingerprinting methods perform well in many cases, they are not robust enough to sophisticated modifications like image embedding and image combining. This paper proposes an image fingerprinting approach which is robust to such modifications. Firstly, we introduce an SIFT-based algorithm to extract image keypoints as the unique fingerprint. Secondly, a technique based on geometry isomorphic relationship is utilized to select valid keypoints in the queried image. Finally, by calculating the matched keypoints pairs between the queried image and the pre-registered image, we can make the decision whether they are homologous. Experimental results show that the proposed method is robust to image embedding and combining, without a high computational complexity.

1. Introduction

With the fast advancement of digital media techniques, DRM (Digital Rights Management) has attracted more and more attention. Among the DRM methods, the digital fingerprinting concept [1] is a novel and fascinating idea. It can be applied to a variety of digital media, such as videos, audios, and images.

The image fingerprint [2] can identify one image from others uniquely. In a fingerprinting system, the fingerprint of a queried image is compared with fingerprints of pre-registered images stored in the database to determine whether they are homologous. Moreover, it is required that the fingerprinting algorithm should be highly distinctive and robust to extensive types of image modifications [3].

Several efforts have been made to establish a practical image fingerprinting algorithm. Jin *etc.* presented an approach using the Radon transform [2]. They obtained fingerprint bits by nonlinear operation and random permutation of Radon feature. Paul *etc.* proposed a method based on the Trace transform [4]. They applied functionals over all possible lines in an image to extract representations of global features. This method has been accepted by MPEG recently.

Although the existing image fingerprinting methods perform well in many cases, they cannot deal with some sophisticated modifications, such as image embedding and combining. On Internet, it is a popular trick to use image materials illegally by embedding a rotated or cropped part of one image into another, or combining several different images together to create a new one. Obviously, it damages global features of an image. Since most of the existing methods depend on global features, they cannot handle such modifications.

Towards this problem, we propose an SIFT-based image fingerprinting method which is robust to these kinds of modifications. Intuitively, the scale and rotation invariant characteristic and the localization capability of SIFT can help us to establish an efficient and robust image fingerprint. In general, we design the algorithm in three steps. Firstly, we design a fingerprint extracting algorithm using the well-known SIFT method. The features of SIFT keypoints of an image is treated as its fingerprint. Secondly, the keypoints of the queried and pre-registered images are compared to obtain the matched pairs. After eliminating outliers, we examine each matched pair to determine whether their locations satisfy the isomorphic relationship. Finally, we calculate the number of matched pairs which satisfy the isomorphic relationship. The ratio between these pairs and total matched pairs is used to decide whether the two images are homologous or heterogeneous.

The remainder of the paper is organized as follows. In section 2, we review the SIFT algorithm briefly. In section 3, our method is described in detail. The experimental results are presented and analyzed in section 4. In section 5, we conclude the paper and discuss some future work.

2. SIFT in brief

The SIFT algorithm [5] is a famous algorithm in the field of object recognition. It draws a high attention and acquires an extensive use because of its remarkable advantages. Firstly, it provides a high distinctiveness. Secondly, it is robust to a variety of affine transforms. Thirdly, it can distinguish local features in an image. Theoretically, SIFT is established basing on the solid theory of scale-space. In general, a practice SIFT algorithm consists five steps.

(a) Generate the scale space. We build the difference-of-Gaussian scale-space $D(x, y, \sigma)$. In a formulation way, it is defined as

$$D(x,y,\sigma) = (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y) = L(x,y,k\sigma) - L(x,y,\sigma), \quad (1)$$

where $I(x, y)$ is the original image, and (x, y) is the coordinate of a pixel. In addition, $G(x, y, \sigma)$ is the scale changeable Gaussian function, defined as

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \quad (2)$$

(b) Detect the space extreme points. The value of $D(x, y, \sigma)$ of one pixel (x, y) is compared with those of its 8 neighbors in the same scale and 9*2 neighbors in the two adjacent scales. If $D(x, y, \sigma)$ possesses the maximal or the minimal value, then the pixel (x, y) will be determined to be an extreme point. In this way, we can detect all the actual extreme points as candidates for the keypoints.

(c) Localize accurate keypoints. Using the fitted 3D quadratic function, we can identify the locations and scales of candidates accurately. Then the candidates with low contrasts or unstable edge responses will be eliminated. Accordingly, the others will be retained as the accurate keypoints.

(d) Assign orientations of the keypoints. An orientation histogram is formed from the gradient orientations of sample points within a region around one keypoint. And the highest peak corresponds to the dominant orientation of this keypoint. After this step, every keypoint will have a triple of attributes: location, scale, and orientation.

(e) Generate descriptors of keypoints. The gradient orientation histogram information of the neighboring points in suitable scale is used to generate a descriptor of one keypoint. The descriptor and the triple of attributes together constitute the feature of a keypoint.

SIFT algorithm possesses many prominent merits, thus it is an attractive attempt to import the SIFT approach into the image fingerprinting field.

3. SIFT-based Image Fingerprinting

We adopt the SIFT algorithm into the image fingerprinting system. Intuitive, since the SIFT algorithm is able to discover features of keypoints which represent objects in an image, we can estimate whether two images have an exactly identical object by matching keypoints. In this way, we can determine whether two images contain the identical content entirely or partially. In general, the proposed method consists of three major steps: extracting fingerprints, matching keypoint pairs, and detecting homology. Figure 1 depicts the flow chart of the proposed method.

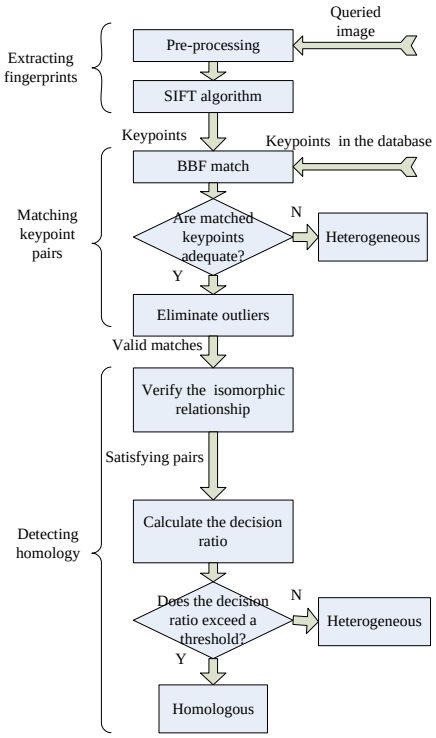

Figure 1. The flow chart of the proposed method.

3.1 Extracting fingerprint

Initially, a pre-processing is applied to the original image, which is named image *A*. The image is resized to a size of 256 pixels * N pixels, where N <= 256, while its aspect ratio is maintained. The resized image will replace the original one in the following algorithm.

After the pre-processing, the SIFT keypoints and their corresponding features of image *A* will be extracted. These features constitute the fingerprint of image *A*. Analogically, the fingerprints of the pre-registered images have been extracted and stored in the database beforehand. The fingerprints in the database will be compared with that of image *A* one by one. We take one pre-registered image *B* as an example.

3.2 Matching keypoint pairs

With the extracted fingerprints of image *A* and *B*, we then detect the matched pairs of keypoints between them. Two steps are involved to find the accurate pairs.

Firstly, we construct a k-d tree [6] from SIFT features of image *B*. In order to accelerate the matching speed, we apply the Best-Bin-First (BBF) algorithm [7] to implement keypoint matching. This can provide a speedup by about two orders of magnitude and result very few incorrect matches. If the number of matched keypoints is larger than a threshold, which is suggested to be 15 by experiment, it indicates a high probability that image *A* and *B* possesses some identical parts. Thus some further processes are required. Otherwise, we will attain a conclusion that the two images are heterogeneous.

Secondly, we decimate the outliers among the matched keypoints. Inevitably, there are always some incorrect matches which form outliers to the cluster of keypoints matched correctly. Outliers can deteriorate the result seriously, thus they must be eliminated. Intuitively, the matched keypoints of image *A* should centralize in one region and assemble to be a cluster, if image *A* and *B* contain an identical part and all matches are correct. As a result, we can determine whether one matched point is an outlier or not by its location relative to the center of the cluster of matched keypoints. If a matched keypoint is too far away from the central location, it is very possible to be an outlier. Therefore, we design a convergence process to eliminate the outliers.

(a) Calculate the centroid $P_c(x_c, y_c)$ of all matched keypoints of image *A* as

$$x_c = \frac{1}{N}\sum_{i=1}^{N} x_i \text{ and } y_c = \frac{1}{N}\sum_{i=1}^{N} y_i , \qquad (3)$$

where N is the number of matched keypoints. $P_i(x_i, y_i)$ is the coordinate of a matched keypoint of image *A*.

(b) Compute the average distances from all matched keypoints to the centroid as

$$d_c = \frac{1}{N}\sum_{i=1}^{N} \| P_i - P_c \| = \frac{1}{N}\sum_{i=1}^{N} \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} . \qquad (4)$$

(c) If the distance of one matched keypoint to the centroid is farther than d_{out}, it is considered as an outlier and will be treated as an invalid matched keypoint. Experimentally, we suppose d_{out} to be $1.4d_c$.

(d) If no keypoint satisfies the outlier condition, the outlier eliminating process will be terminated. Otherwise, the process return to step (a) and execute these steps recursively.

This process is operated on matched keypoints of image *A*. If one keypoint of image *A* is considered as an outlier, the corresponding matched keypoint of image *B* is discarded as well. The retained matched keypoints are confirmed to be valid pairs and their total numbers will be counted, denoted as *N*.

3.3 Detecting homology

After finding the valid matched pairs, we will detect homology by verifying whether the valid pairs satisfy a geometry isomorphic relationship. Basically, the geometry relationship among keypoints will not be changed by image modifications such as rotation, scale, and affine transform, *etc*, according to the principle of SIFT. Thus it is reasonable to hypothesize isomorphism.

Abstractly, keypoints in image *A* and *B* can be treated as two spaces, while the keypoints pairs matching can be seen as a mapping. Based on the theory of SIFT, the matched keypoint pairs will possess a geometry isomorphic relationship if image *A* and *B* contain an identical content, which correspond to an isomorphic mapping. An isomorphic mapping should hold operations between two spaces. In a formulation way, we say

$$L(x_1, x_2, ...x_n) = L(f(x_1), f(x_2), ... f(x_n)) , \qquad (5)$$

where *L* is a functional operator and *f* is the isomorphic mapping. x_i and $f(x_i)$ belongs to the two spaces respectively. Therefore, we can detect infringements by verifying the isomorphism. And the isomorphism can be tested by (5).

In practice, we utilize a simple geometry proportion as the testing operator. At first, we define d_{Ai} as the distance between the *i*th matched keypoint and the centrodic of image *A*. d_{Bi} represents that of image *B* correspondingly. Also, we let d_{Ac} and d_{Bc} represent the average distances between keypoints in image *A* and *B* respectively. According to (5), we should have

$$d_{Ai}/d_{Ac} = d_{Bi}/d_{Bc} \Rightarrow d_{Ai}/d_{Bi} = d_{Ac}/d_{Bc} , \qquad (6)$$

if the isomorphism exists. In order to handle perturbations, we relax condition (6) into

$$(1-\Delta) d_{Ac}/d_{Bc} \le d_{Ai}/d_{Bi} \le (1+\Delta) d_{Ac}/d_{Bc} , \qquad (7)$$

where Δ is a disturbance term, and is set to be 0.1 experientially. If valid matched pairs *i* fulfill condition (7), we consider it to satisfy the isomorphic relationship. The number of keypoints that satisfy the isomorphic relationship will be counted, denoted as *n*.

Finally, we calculate the ratio between *n* and *N* as

$$\rho = n/N , \qquad (8)$$

which is used to estimate whether image *A* and *B* include an identical part. Experimentally, it presents a high probability that a homology is detected if ρ is more than 0.9.

4. Experimental Results

To evaluate the proposed method, we carried out some experiments. The testing images are 1000 independent images selected from the CD-ROMs "Art Explosion 800000" by Nova [8]. We extract fingerprints from all the testing images in advance to build the database, and generate 500 queried images by imposing modifications of embedding and combining on the testing images. For generality, cropping, rotation and scale are involved in the modifications. One example of the queried images is shown in Figure 2. In addition, we set the parameters of the SIFT algorithm as the optimal values recommended in [5].

Figure 2. An example of embedding some content from one image to another. The matching result is: 36 total matched pairs, 35 valid pairs (n) and 34 isomorphic pairs (N). The ratio ρ between n and N is about 0.97.

Given a queried image, we perform the proposed matching process with each image in the database. Figure 2 demonstrates one matching result. In order to evaluate the accuracy of our method, we calculate the *false-alarm rate* and the *recall* as

$$false-alarm\ rate = \frac{number\ of\ \ false\ matches}{total\ number\ of\ \ processes}, \qquad (9)$$

and

$$recall = \frac{number\ of\ \ correct\ matches}{total\ number\ of\ \ homologous\ images}. \qquad (10)$$

A *match* is a pair of images which are claimed to have an identical part. A *false match* is a *match* that is claimed falsely and a *correct match* is one that is claimed correctly. In our experiments, the total number of processes is 500000, and that of homologous images is 1000 definitely. Moreover, the MPEG method [4] is also tested for comparisons.

The experimental results are exciting. As depicted in Table 1, the *false-alarm rate* and the *recall* of the proposed method is 0.0002% and 92.4% respectively. This result indicates that the proposed method can handle complicated modifications fairly well. Contrarily, the results of MPEG method are 0.0006% and 3% correspondingly while using the threshold which is set in [4]. We can see clearly that our method performs much better than MPEG proposal in detecting such a modification. Since the MPEG method uses the global features and our method uses local features, which are robust to local modifications.

The computational complexity of the proposed method is acceptable. On a PC with the Pentium 3.4GHz processor, the average time to extract a fingerprint is about 0.400 second, and the matching process is almost real time. Although the original SIFT features demand a considerably large storage size, they can be compressed without decreasing the efficiency of the algorithm obviously. To be more concrete, we can decrease the dimensions of descriptors using PCA [9] or change parameters of SIFT algorithm in order to extract fewer keypoints in our future work.

Table 1 Experimental results

	Proposed	MPEG
false-alarm rate	0.0002%	0.0006%
recall	92.4%	3%

5. Conclusion

Aiming at complicated modification of images, we propose a novel image fingerprinting solution based on the SIFT algorithm. With the fingerprint depicted by SIFT feature, we introduce a verified keypoints matching strategy according to a geometry isomorphic hypothesis. Experiments indicate that the proposed method can detect embedding or other complicated modifications accurately. In the future, we will find a way to compress the SIFT based fingerprint more efficiently, so as to make it more applicative.

6. Acknowledgement

This work is supported by the National Key Technology R&D Program [2006BAH02A10 and 2006BAH02A13] of China.

7. References

[1] P. Cano, E. Batlle, T. Kalker, J. Haitsma, "Review of Algorithms for Audio Fingerprinting", *Multimedia Signal Processing IEEE workshop*, 2002, pp.169-173.

[2] J. S. Seo, J. Haitsma, T. Kalker, *et al*, "A robust image fingerprinting system using the Radon transform", *Signal Processing: Image Communication*, 2004, 19: pp.325-339.

[3]MPEG Video Sub-Group, "Call for proposal on Image & Video Signature Tools", MPEG Doc No.N9216, July, 2007.

[4] P. Brasnett, M. Bober, "Proposed Improvements to Image Signature XM 31.0", MPEG Doc No.M14983, Oct. 2007.

[5] D.G. Lowe, "Distinctive image features from scale invariant keypoints", *International Journal of Computer Vision*, 2004, 60(2): pp.91-110.

[6] J.H. Friedman, J.L. Bentley, and R.A. Finkel, "An algorithm for finding best matches in logarithmic expected time", *ACM Transactions on Mathematical Software*, 1977, 3: pp.209-226.

[7] J. Beis, and D.G. Lowe, "Shape indexing using approximate nearest-neighbour search in high-dimensional spaces", *Conference on Computer Vision and Pattern Recognition*, Puerto Rico, 1997, pp.1000-1006.

[8] Avanquest Software, Art Explosion 800000, Oct. 2004.

[9] Y. Ke, R. Sukthankar, "PCA-SIFT: A More Distinctive Representation for Local Image Descriptors", *CVPR*, 2004, volume: 2, pp. 506- 513.