Multimed Tools Appl (2011) 54:341–363
DOI 10.1007/s11042-010-0534-y

A robust content based image watermarking using local invariant histogram

Xiang-Yang Wang · Pan-Pan Niu · Lan Meng · Hong-Ying Yang

Published online: 18 May 2010

Abstract Desynchronization attack is known as one of the most difficult attacks to resist, which can desynchronize the location of the watermark and hence causes incorrect watermark detection. Based on multi-scale SIFT (Scale Invariant Feature Transform) detector and local image histogram shape invariance, we propose a new content based image watermarking algorithm with good visual quality and reasonable resistance toward desynchronization attacks in this paper. Firstly, the stable image feature points are extracted from the original host by using multi-scale SIFT detector, and the local feature regions (LFRs) are constructed adaptively according to the feature scale theory. Then, the discrete Fourier transform (DFT) is performed on the LFR, and the local image histogram is extracted from a selected DFT amplitude range. Finally, the bins of the histogram are divided into many groups, and the digital watermark is embedded into LFR by reassigning the number of DFT amplitudes in bin groups. By binding the watermark with the geometrically invariant image features, the watermark detection can be done without synchronization error. Experimental results show that the proposed image watermarking is not only invisible and robust against common image processing operations such as sharpening, noise adding, and JPEG compression, but also robust against the desynchronization attacks such as rotation, translation, scaling, row or column removal, and cropping.

This work was supported by the National Natural Science Foundation of China under Grant No. 60773031 & 60873222, the Open Foundation of State Key Laboratory of Networking and Switching Technology of China under Grant No. SKLNST-2008-1-01, the Open Foundation of State Key Laboratory of Information Security of China under Grant No. 03-06, the Open Foundation of State Key Laboratory for Novel Software Technology of China under Grant No. A200702, the Open Foundation of Key Laboratory of Modern Acoustics Nanjing University under Grant No. 08-02, and Liaoning Research Project for Institutions of Higher Education of China under Grant No. 2008351.

X.-Y. Wang (✉) · L. Meng · H.-Y. Yang
School of Computer and Information Technology, Liaoning Normal University, No.850 Huanghe Road, Shahekou District, Dalian 116029, China
e-mail: wxy37@126.com

P.-P. Niu
School of Information Science & Technology, Dalian Maritime University, Dalian 116026, China

Keywords Image watermarking · Desynchronization attacks · Multi-scale SIFT detector · Histogram · DFT

1 Introduction

WITH the rapid growth and widespread use of network distributions of digital media content, there is an urgent need for protecting the copyright of digital content against piracy and malicious manipulation. One solution to achieve data security is the digital watermarking technology that embeds hidden information or secret data in a host media signal. It serves as a suitable tool to identify the source, creator, owner, distributor, or authorized consumer of a document or an image. It can also be used to detect whether a document or an image is illegally distributed or modified [5, 9, 20].

In recent years, many image watermarking schemes have been proposed for copyright protection. On the other hand, attacks against image watermarking systems have become more sophisticated [26]. In general, these attacks can be categorized into common image processing operations such as median filtering, sharpening, noise adding, and JPEG compression, and desynchronization attacks such as rotation, scaling, translation (RST), random bending attack (RBA), and cropping. While the common image processing operations reduce watermark energy, desynchronization attacks induce synchronization errors between the original and the extracted watermark during the detection process. Most of the previous image watermarking schemes are robust to common image processing operations, but show severe problems to desynchronization attacks. Nowadays, several approaches that counterattack desynchronization attacks have been developed. These schemes [14, 25, 26] can be roughly divided into invariant transform, template insertion and feature-based algorithms.

Invariant transform The most obvious way to achieve resilience against desynchronization attacks is to use an invariant transform. In [21–24, 26], the watermark is embedded in an affine-invariant domain by using Fourier-Mellin transform, generalized Radon transform, geometric moments, and histogram shape respectively. Despite that they are robust against global affine transformations, those techniques involving invariant domain suffer from implementation issues and are vulnerable to cropping and RBA.

Template insertion Another solution to cope with desynchronization attacks is to identify the transformation by retrieving artificially embedded references. By focusing on a simple example, Barni [1] investigated the effectiveness of exhaustive watermark detection and resynchronization through template matching against watermark desynchronization. Motallebi et al. [12] gave a RST invariant wavelet-based image watermarking, in which the so-called Fourier domain for embedding a template has been employed and the watermark was embedded in wavelet domain. LPM (Log-Polar Mapping) domain and phase information of the image were used to determine RST attacks. Liu et al. [10] presents an image rectification scheme that can be used by any image watermarking algorithm to provide robustness against rotation, scaling and translation (RST). In the watermarking, a small block is cut from the LPM domain as a matching template, and a new filtering method is proposed to compute the cross-correlation between this template and the magnitude of the LPM of the image having undergone RST transformations to detect the rotation and scaling parameters. However, this kind of approach can be tampered with by the malicious attack. In addition, they are similarly vulnerable to local geometrical distortions.

Feature-based The last category is based on media features and our approach belongs to this category. Its basic idea is that, by binding the watermark with the geometrically invariant image features (Local Feature Region, LFR), the watermark detection can be done without synchronization error. Bas et al. [2] use the Harris detector to extract features and Delaunay Tessellation to define watermark embedding regions. In [17], the Mexican hat wavelet is used to extract feature points as well, and several copies of the watermark are embedded in the disks centered at the feature points. Lee et al. [7] proposed a robust image watermarking scheme that uses local invariant features. Wang et al. [18] propose a feature-based image watermarking scheme by using the scale space theory and image normalization technique. Lee et al. [8] proposed a watermarking method that is robust to geometric distortions. In order to synchronize the location for watermark insertion and detection, the circular Hough transform is used, which extracts circular features that are invariant to geometric distortions. The circular features are then watermarked using additive way on the spatial domain. Seo et al. [16] introduce a novel content-based image watermarking method based on invariant regions of an image. The invariant regions are self-adaptive image patches that deform with geometric transformations. Three different invariant-region detection methods based on the scale-space representation of an image were considered for watermarking. At each invariant region, the watermark is embedded after geometric normalization according to the shape of the region. By binding watermarking with invariant regions, resilience against geometric transformations can be readily obtained. Pham et al. [13] present a robust object-based watermarking algorithm using the local image feature in conjunction with a data embedding method based on DCT, and the digital watermark is embedded in the DCT domain of randomly generated blocks in the selected object region. In [19], a feature-based image watermarking scheme against desynchronization attacks is proposed. The robust feature points, which can survive some signal-processing and affine transformation, are extracted by using the Harris-Laplace detector. In watermark detection, the digital watermark is recovered by maximum membership criterion.

The theoretical analysis and experimental results show that the feature-based approaches are better than others in terms of robustness [25, 26]. However, some drawbacks indwelled in current feature-based schemes restrict the performance of watermarking system. Firstly, the feature point extraction techniques adopted by current feature-based approaches, for instance, Harris detector (including Harris-Laplace detector) or Mexican hat wavelet, are not fully affine invariant and sensitive to image modification. Secondly, the fixed value is used to determine the size of LFR so that the watermarking scheme is vulnerable to the scale change of the image. Thirdly, the current feature-based watermarking has not constructed the invariant region of LFR, which lowers the robustness against local desynchronization attacks.

In this paper, a new content based image watermarking algorithm with good visual quality is proposed, in which the multi-scale SIFT detector and local image histogram shape invariance is utilized. Experimental results show that the proposed watermarking is not only invisible and robust against common image processing operations, but also resilient to desynchronization attacks.

The rest of this paper is organized as follows. Section 2 presents the feature extraction used in the proposed scheme. Section 3 describes a new LFR construction method. In Section 4, image histogram and its shape invariance are discussed. Section 5 contains the description of our watermark embedding procedure. Section 6 covers the details of the watermark detection procedure. Simulation results in Section 7 will show the performance of our scheme. Finally, Section 8 concludes this presentation.

2 Multi-scale SIFT detector and feature points extraction

In order to detect watermarks without the help of the original image, we must look for reference points that are perceptually significant and can resist various common image processing operations and desynchronization attacks. These reference points can also act as mark for locating resynchronization between watermark embedding and detection. We will use the term "feature points" to denote these reference points.

Mexican Hat wavelet and Harris detector have been widely used to extract feature points [6]. Mexican Hat wavelet is stable under noise-like processing, but it is sensitive to some affine transformations (especially scaling). Harris detector is stable under majority attacks such as rotation, noise addition and illumination change, etc. However, it is very sensitive to changes in image scale. Recently, there has been an impressive body of work on extending local feature points to be invariant to full affine transformations [3, 15]. However, none of these approaches are yet fully affine invariant, as they start with initial feature scales and locations selected in a non-affine invariant manner due to the prohibitive cost of exploring the full affine space. The affine frames are also more sensitive to noise than those of the scale-invariant features. To resolve the weakness of current feature extraction, a novel method called multi-scale SIFT detector is adopted in this paper.

The multi-scale SIFT (Scale-Invariant Feature Transform) detector was proposed by Lowe [11] and has proved to be invariant to image rotation, scaling, translation, partial illumination changes, and projective transformations. This descriptor extracts feature points by considering local image characteristics and describes the properties of each feature point such as the location, scale, and orientation. The basic idea of multi-scale SIFT descriptor is detecting feature points efficiently through a staged filtering approach that identifies stable points in the scale-space.

The multi-scale SIFT descriptor can extract local feature points from following steps: (1) select candidates for feature points by searching peaks in the scale-space from a Difference of Gaussian (DoG) function, (2) localize feature points using measures of their stability, (3) assign orientations based on local image properties, and (4) calculate feature descriptors which represent local shape distortions and illumination changes.

2.1 Selecting candidate image feature points

In multi-scale SIFT detector, the image feature points will be detected by using a cascade filtering approach that uses efficient algorithms to identify candidate locations that are then examined in further detail. The first stage of feature points detection is to identify locations and scales that can be repeatably assigned under differing views of the same object. Detecting locations that are invariant to scale change of the image can be accomplished by searching for stable features across all possible scales, using a continuous function of scale known as scale-space.

It has been shown by Koenderink and Lindeberg [11] that under a variety of reasonable assumptions the only possible scale-space kernel is the Gaussian function. Therefore, the scale-space of an image is defined as a function, $L(x,y,\sigma)$, that is produced from the convolution of a variable-scale Gaussian, $G(x,y,\sigma)$, with an input image, $I(x,y)$:

$$L(x,y,\sigma) = G(x,y,\sigma)^{*}I(x,y) \tag{1}$$

Where * is the convolution operation in x and y, and

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

In order to efficiently detect stable feature points locations in scale-space, Lowe et al [11] have proposed using scale-space extrema in the Difference-of-Gaussian function convolved with the image, $D(x,y,\sigma)$, which can be computed from the difference of two nearby scales separated by a constant multiplicative factor k:

$$\begin{aligned} D(x,y,\sigma) &= (G(x,y,k\sigma) - G(x,y,\sigma))^{*}I(x,y) \\ &= L(x,y,k\sigma) - L(x,y,\sigma) \end{aligned} \tag{3}$$

An efficient approach to construction of $D(x,y,\sigma)$ is shown in Fig. 1. To generate a scale-space, we use a Difference of Gaussian function, in which we successively smooth an image with a variable scale (σ_1, σ_2, and σ_3) Gaussian filter and calculate difference images by subtracting two successive smoothed images. The parameter σ is a variance (called a scale) of the Gaussian function. The scale of the scale-space images is determined by the nearby scale σ_1, σ_2, or σ_3 of the Gaussian-smoothed image. In this scale-space, we retrieve all local maximums and minimums by checking 8 closest neighborhoods in the same scale and 9 neighborhoods in the scale above and below (see Fig. 1). These locations are invariant to the scale change of images. Usually, the scale corresponding to a candidate image feature point is called feature scale.

2.2 Eliminate the unstable image feature points

After candidate feature points are found, some unstable image feature points that have a low contrast or are poorly localized along edges are removed by measuring stability of each feature point at its location and scale.

For stability, it is not sufficient to reject feature points with low contrast. The Difference-of-Gaussian function will have a strong response along edges, even if the location along the edge is poorly determined and therefore unstable to small amounts of noise.

A poorly defined peak in the Difference-of-Gaussian function will have a large principal curvature across the edge but a small one in the perpendicular direction. The principal

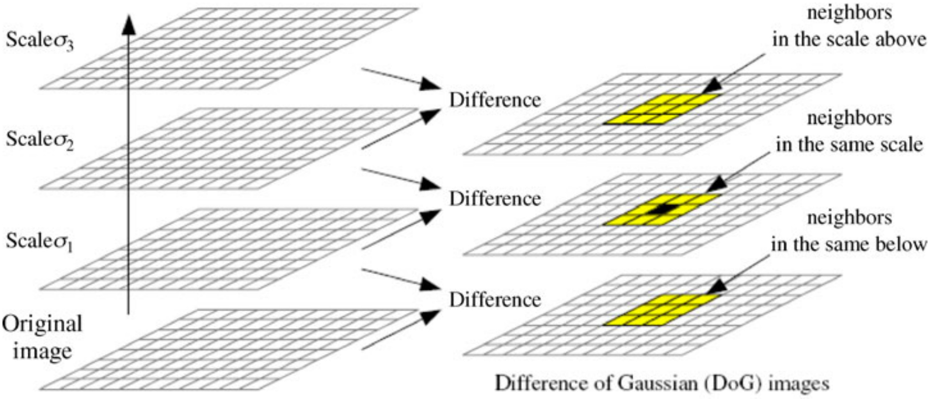

Scaleσ_3
Scaleσ_2
Scaleσ_1
Original image
Difference
Difference
Difference
neighbors in the scale above
neighbors in the same scale
neighbors in the same below
Difference of Gaussian (DoG) images

Fig. 1 Scale-space from the Difference of Gaussian function and the closest neighborhoods of a pixel

curvatures can be computed from a 2 × 2 Hessian matrix, M_H, computed at the location and scale of the feature point:

$$M_H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \tag{4}$$

The derivatives are estimated by taking differences of neighboring sample points.

The eigenvalues of M_H are proportional to the principal curvatures of D. Borrowing from the approach used by Harris [11], we can avoid explicitly computing the eigenvalues, as we are only concerned with their ratio. Let α be the eigenvalue with the largest magnitude and β be the smaller one. Then, we can compute the sum of the eigenvalues from the trace of M_H and their product from the determinant:

$$\begin{gathered} Tr(M_H) = D_{xx} + D_{yy} = \alpha + \beta \\ Det(M_H) = D_{xx}D_{yy} - \left(D_{xy}\right)^2 = \alpha\beta \end{gathered} \tag{5}$$

In the unlikely event that the determinant is negative, the curvatures have different signs so the point is discarded as not being an extremum. Let γ be the ratio between the largest magnitude eigenvalue and the smaller one, so that $\alpha = \gamma\beta$. Then,

$$\frac{Tr(M_H)^2}{Det(M_H)} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(\gamma\beta+\beta)^2}{\gamma\beta^2} = \frac{(\gamma+1)^2}{\gamma}$$

Which depends only on the ratio of the eigenvalues rather than their individual values. The quantity $(\gamma+1)^2/\gamma$ is at a minimum when the two eigenvalues are equal and it increases with γ. Therefore, to check that the ratio of principal curvatures is below some threshold, γ, we only need to check

$$\frac{Tr(M_H)^2}{Det(M_H)} \le \frac{(\gamma+1)^2}{\gamma} \tag{6}$$

So if the above inequality fails, the image feature point is removed from the candidate list. This is very efficient to compute, with less than 20 floating point operations required to test each feature. The experiments in this paper use a value of γ=10, which eliminates feature points that have a ratio between the principal curvatures greater than 10.

2.3 Orientation assignment

By assigning a consistent orientation to each feature point based on local image properties, the feature point descriptor can be represented relative to this orientation and therefore achieve invariance to image rotation.

In order to assign an orientation, the gradient magnitude m and orientation θ are computed by using the pixel difference as follows

$$\begin{gathered} m(x,y) = \sqrt{\left(L_{x+1,y} - L_{x-1,y}\right)^2 + \left(L_{x,y+1} - L_{x,y-1}\right)^2} \\ \theta(x,y) = \tan^{-1}\left(L_{x,y+1} - L_{x,y-1} / L_{x+1,y} - L_{x-1,y}\right) \end{gathered} \tag{7}$$

where L is the Gaussian smoothed image with the closest scale where feature points are found. The histogram of orientations is formed from the gradient orientation at all sample points within a circular window of a feature point, whose size is dependent on the scale of the feature. Peaks in this histogram correspond to dominant directions of each feature point.

Finally, for illumination condition invariance, we define 8 orientation planes, make the gradient magnitude and orientation smooth by applying a gaussian filter, and then sample over a 4 by 4 grid of locations with 8 orientation planes. This feature vector, 4×4× 8 elements, is normalized by dividing the square root of the sum of squared components to reduce the effect of illumination changes.

The multi-scale local feature points obtained through above process are invariant to rotation, scaling, translation, and partly illumination changes of the image. Figure 2 gives the feature points extracted by multi-scale SIFT descriptor for test images $256\times256\times8bit$ Lena, Barbara, and Mandrill.

3 Local feature region (LFR) construction

The LFR, which reflect the important semantics, are the subsets of the host image. As watermark embedding and detection regions, the LFR must solve the problem of geometrical synchronization. Bas et al. [2] proposed the triangular regions, one of which is generated by three feature points, but this approach is sensitive to cropping. In [17] and [18], the disk regions, one of which is generated by only one feature point, are proposed. Since these disks are independent with each other, the resilience against cropping therefore is improved in some measure. However, the local image characteristic is not considered in this approach so that it is sensitive to scaling.

Based on the fact that the feature scale of feature points varies with the local image characteristic, we can select some strong feature points to be the centers of the LFR, and the radius of the LFR is chosen in accordance with the feature scale of the selected points. As a result, a set of adaptive disks (LFR) is generated. The radius of these disks is defined as:

$$\Re = \tau \cdot round(\sigma) \tag{8}$$

Where $\Re$ is the radius of the disk, σ is the feature scale of feature points, and τ is a positive integer, which is used to adjust the size of the disks. A large value of τ would increase the capacity of the watermarking scheme. But, the robustness of watermarking

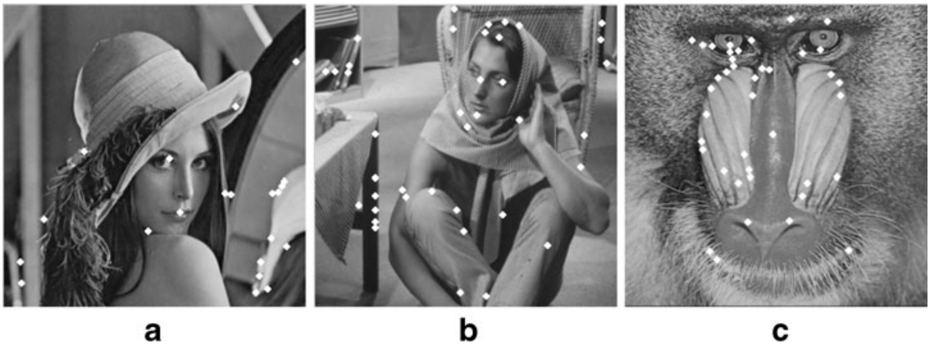


Fig. 2 The feature points for some test images. **a** The test image Lena; **b** The test image Barbara; **c** The test image Mandrill

scheme would decrease for large τ. Hence, there is a trade off between capacity and robustness. The range of $\Re$ is restricted to be:

$$round(\sigma) \leq \Re \leq \frac{\min(M, N)}{2} \tag{9}$$

Where M and N are the height and width of the host image respectively. Since the LFRs should not interfere with each other, we select the LFR with a large number of feature points when there is interference between any two LFR because the selected LFR is perceptually highly textured. An example of LFR constructed by using the proposed method is shown in Fig. 3. For convenience, we represent only one LFR. It can be easily found that the LFR is formulated robustly even with rotation and scaling of the image.

We apply feature points extraction (see Section 2) and LFR construction (see Section 3) to the popular test images $256 \times 256 \times 8bit$ Lena, Barbara, and Mandrill. As shown in Fig. 4, the number of LFRs is 7, 9 and 6 in Lena, Barbara, and Mandrill, respectively.

4 The histogram and its shape invariance

From the probability viewpoint, the frequency of gray may be seen as the probability, thus, histogram corresponds to the probability density function (PDF), and cumulative distribution function is cumulative sum of histogram, i.e., integral of PDF. A histogram is often used to describe the data distribution. It is a display of statistical information that uses rectangles to show the frequency of data items in successive numerical intervals of equal

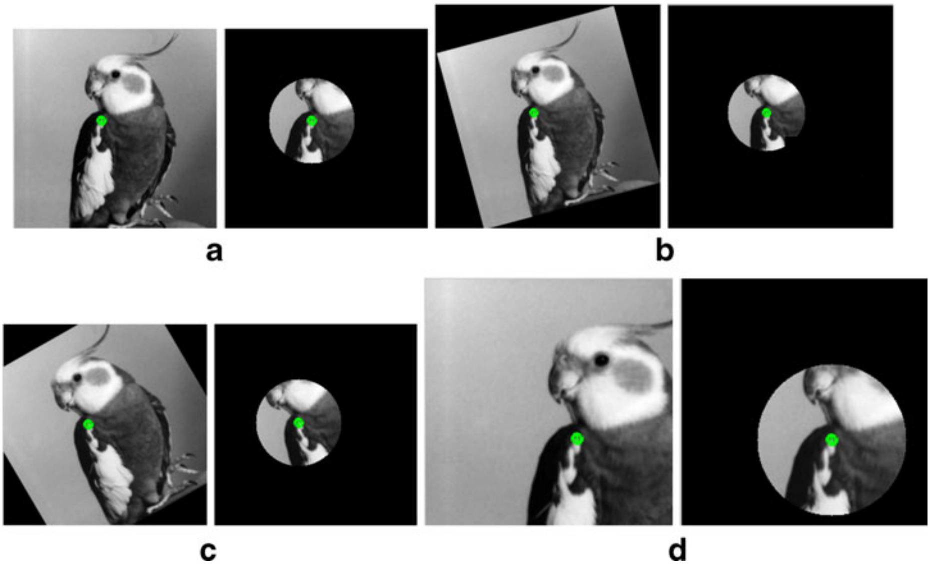


Fig. 3 **a** The original image (Bird) and its LFR (the feature scale is 3.58); **b** 15 degrees rotated image and its LFR (the feature scale is 3.58); **c** 30 degrees rotated and cropped image and its LFR (the feature scale is 3. 58)); **d** 10% sheared and 140% scaled image and its LFR (the feature scale is 5.07). To determine the radius of LFR, τ is set to 11

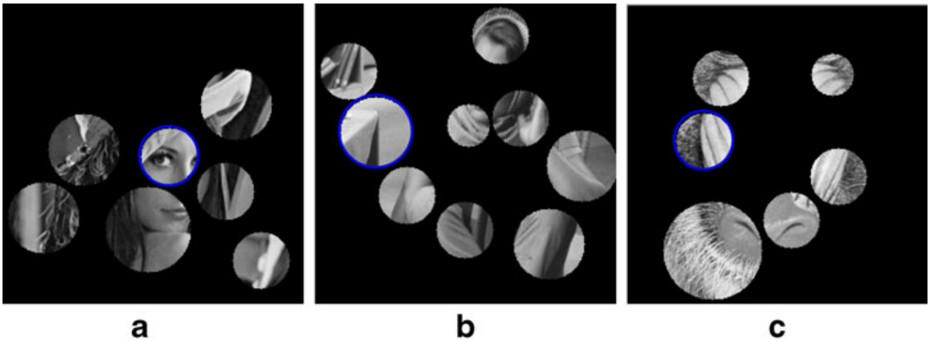


Fig. 4 The local feature regions (LFRs) for test images. **a** The test image Lena; **b** The test image Barbara; **c** The test image Mandrill

size. In the most common form of histogram, the independent variable is plotted along the horizontal axis and the dependent variable is plotted along the vertical axis. It's the most basic statistical feature for digital image [21].

For digital image, the most common form of the image histogram is obtained by splitting the range of the image pixel into equal-sized bins. Then, the number of pixels from the image that fall into each bin is counted. The image histogram may be described by

$$H = \{h(i)|i = 1, \ldots, L\} \tag{10}$$

where H is a vector denoting the gray-level histogram of the image $F = \{f(x,y)|x = 1,2,\ldots,M, y = 1,2,\ldots,N\}$, and $h(i), h(i) \geq 0$ denotes the number of pixels in the i^{th} bin satisfying $\sum_{i=1}^{L} h(i) = M \times N$. Suppose that the resolution of the image is P bits, the number of bins is calculated as

$$L = \begin{cases} 2^P/Q & if \quad \mathrm{mod}(2^P/Q) = 0 \\ \lfloor 2^P/Q \rfloor + 1 & other \end{cases} \tag{11}$$

where Q is the bin width, $h(i)$ covers the range $[-2^{P-1} + (i-1) \cdot Q, -2^{P-1} + i \cdot Q - 1]$, and $\lfloor\rfloor$ is the floor function.

Consider the case of pure non-proportional scaling over the image F. Suppose that $F' = \left\{f'\left(x', y'\right)\right\}$ is the scaled image with the scaling factors α and β in both vertical and horizontal directions. $f'\left(x', y'\right)$ is the value in the point $\left(x', y'\right)$, theoretically satisfying the expression $f'\left(x', y'\right) = f(x/\alpha, y/\beta)$. In the new version, the number of rows and columns are calculated as $M' = \alpha \cdot M$ and $N' = \beta \cdot N$. The histogram of F' can be formulated as

$$H' = \left\{h'(i)|i = 1, \ldots,, L\right\} \tag{12}$$

which satisfies the expression $h'(i) = h(i) \cdot \alpha \cdot \beta$ in theory, referred to Eq. 7. Equation 8 indicates the invariance of the histogram shape to the scaling operation because under the scaling the number of elements in the bins is modified linearly. In practice, the number of the pixels in each bin may be slightly modified due to interpolation.

Rotation and translation are two common operations, which will be able to modify the pixel positions in the image plane. In the two cases, the histogram shape will be invariant due to the fact that the histogram is independent of the pixel position [4]. This kind of special property also provides the histogram shape a capability against other challenging geometric attacks such as cropping.

5 Watermark embedding scheme

According to the communication model of digital watermarking (the host image, watermark bits and various attacks are viewed as the channel, transmitted information and noises respectively), we can know that all LFRs generated from the host image can be viewed as independent communication channels. To improve the robustness of transmitted information, all channels carry the same copy of the chosen watermark. Based on multi-scale SIFT (Scale Invariant Feature Transform) detector and local image histogram shape invariance, a new content based image watermarking algorithm robust to desynchronization attacks is proposed. Firstly, the stable image feature points are extracted from the original host by using multi-scale SIFT detector, and the local feature regions (LFRs) are constructed adaptively according to the feature scale theory. Then, the DFT transform is performed on the LFR, and the local image histogram is extracted from a selected DFT amplitude range. Finally, the bins are divided into many groups, and the digital watermark is embedded into LFR by reassigning the number of DFT amplitudes in bin groups.

The watermark embedding procedure is shown in Fig. 5.

Let $F = \{f(i,j), 1 \leq i \leq M, 1 \leq j \leq N\}$ denote a host digital image (gray image), and $f(i,j)$ denotes the pixel value at position (i,j). The digital watermark embedding scheme can be summarized as follows.

Step 1: Watermark Generation

A random sequence (Digital Watermark) $W = \{w_i | i = 1, \cdots, L\}$ is generated by the secret key K_1, where L is the size of the watermark sequence, and $w(i) \in \{0, 1\}$.

Step 2: Extraction of Feature Points

The multi-scale SIFT detector is applied to the host image F, and a set of feature points, denoted as $P_i(i = 1, \cdots, n)$, is obtained (see Section 2).

Step 3: Construction of Local Feature Region (LFR)

A set of LFRs, denoted as $O_k(k = 1, \cdots, m)$, is constructed in accordance with the locations of host image and the feature scales of the feature points $P_i(i = 1, \cdots, n)$ (see Section 3).

Step 4: The DFT Transform of LFR

In our watermarking scheme, the digital watermark is embedded in the DFT domain of the LFRs. But it is very hard to perform directly DFT transform on the circle area (LFR) O_k, so a zero-padding operation is considered to solve this problem.

The LFR O_k is mapped to the block of size $2R_k \times 2R_k$ by using zero-padding method, where R_k is the radius of LFR. After watermark embedding, the zero-removing should be

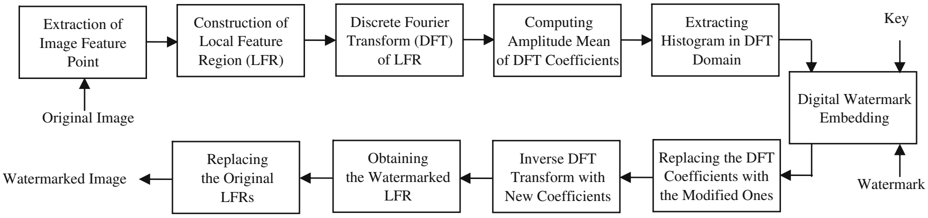

Extraction of Image Feature Point
Construction of Local Feature Region (LFR)
Discrete Fourier Transform (DFT) of LFR
Computing Amplitude Mean of DFT Coefficients
Extracting Histogram in DFT Domain
Key
Original Image
Digital Watermark Embedding
Watermarked Image
Replacing the Original LFRs
Obtaining the Watermarked LFR
Inverse DFT Transform with New Coefficients
Replacing the DFT Coefficients with the Modified Ones
Watermark

Fig. 5 Watermark embedding scheme

used to the block in order to convert it circle area. This procedure is shown in Fig. 6. In fact, there is an energy loss during the zero-padding/ zero-removing procedure, but it is so small and does not affect watermark detection [19].

After the zero-padding operation has been performed on each LFR O_k so as to map the LFR to a square region (Block), the 2D DTF transform is applied to each square consequently and we can obtain the DFT spectrum F_k, amplitude spectrum $M_k = \{m_k(l), l = 1, \cdots, N_k\}$ and phase spectrum φ_k.

Step 5: Computing Mean of DFT Amplitude

According to the statistics results of DFT spectrum, there usually are a few bigger DFT amplitudes and a few smaller ones, which are not suitable for constructing the robust histogram. In order to improve the robustness of DFT amplitude histogram, we will select some significant DFT amplitudes by using statistics mean.

The mean of a given signal is calculated by adding up all the sample values and dividing by the number of them. Usually, it is a statistical measurement of the spread of data values and the divergence of the data values from normal distribution patterns. In this paper, the statistics mean of DFT amplitude is calculated as the sum of the DFT amplitude over its duration

$$\overline{A}_k = \frac{1}{N_k}\sum_{l=1}^{N_k} m_k(l) \tag{13}$$

where $\overline{A}_k$ denotes the DFT amplitude mean for LFR O_k, and $m_k(l)$ is the l^{th} DFT amplitude in LFR O_k.

Step 6: Extracting Histogram of DFT Amplitude

The significant DFT amplitudes (embedding range) $B = \left[(1-\lambda)\overline{A}_k, (1+\lambda)\overline{A}_k\right]$ are selected from LFR O_k, and the histogram of significant DFT amplitudes $H_k = \{h_k(i)|i = 1, \ldots, , L_k\}$ is extracted for embedding digital watermark, where $L_k \geq 2L$. Here, λ is a selected positive number for satisfying $h_k(i) >> L_k$. In this paper, λ=0.6 is a suggested value so that the bins extracted from B have a satisfied bin width and enough DFT amplitudes.

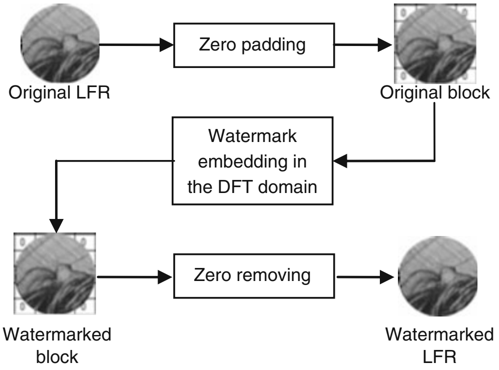


Fig. 6 The zero-padding and the zero-removing operation

Step 7: Digital Watermark Embedding

In our image watermarking, the bins of histogram are divided into a series of groups, each two neighboring bins as a group is used to carry a bit of watermark information by reassigning the number of DFT amplitudes in each two bins.

Let BIN_1 and BIN_2 denote two consecutive bins. Their DFT amplitudes in the number are N_a and N_b. We apply the following rule to embed one bit of digital watermark, described as

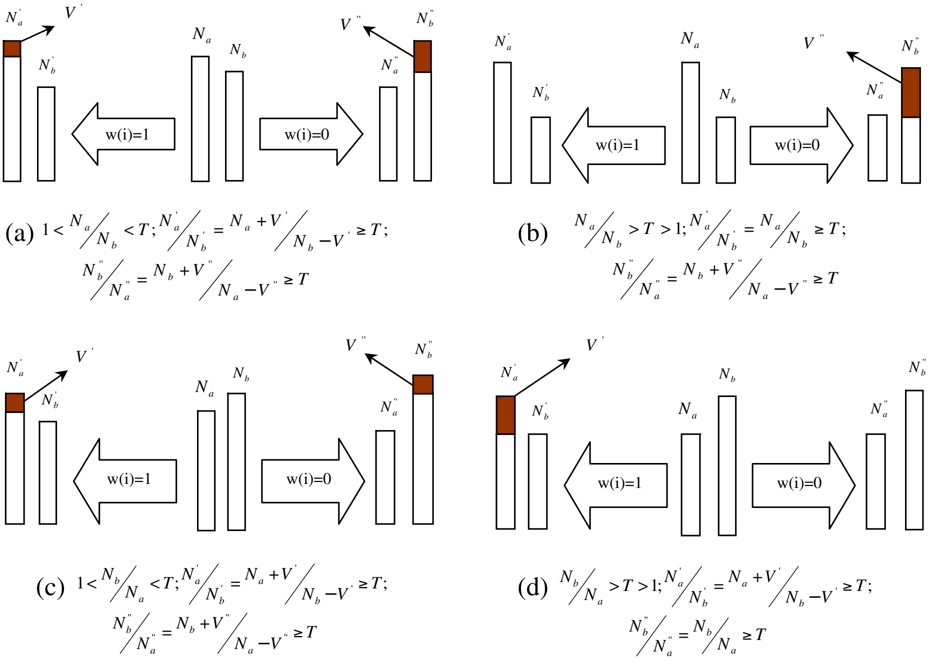
$$\begin{cases} N_a/N_b \ge T & if \quad w(i) = 1 \\ N_b/N_a \ge T & if \quad w(i) = 0 \end{cases} \tag{14}$$

where T is a selected threshold used to control the watermark robustness performance and the embedding distortion.

The embedding one bit digital watermark into two neighboring bins is depicted in Fig. 7.

Corresponding to Eq. 9, if the embedded watermark bit is "1" and $N_a/N_b \ge T$, no operation is needed; Otherwise, if the embedded watermark bit is "1" but $N_a/N_b < T$, some randomly selected DFT amplitudes from BIN_1, in the number denoted by V', will be modified to fall into BIN_2, satisfying the condition $N'_a/N'_b \ge T$. The rule on how to modify the DFT amplitudes from BIN_1 to BIN_2 is referred to Eq. 15

$$m'_{k,1}(i) = m_{k,1}(i) + \Delta,\ 1 \le i \le V' \tag{15}$$



(a) $1 < \frac{N_a}{N_b} < T; \frac{N'_a}{N'_b} = \frac{N_a + V'}{N_b - V'} \ge T;$
$\frac{N''_b}{N''_a} = \frac{N_b + V''}{N_a - V''} \ge T$

(b) $\frac{N_a}{N_b} > T > 1; \frac{N'_a}{N'_b} = \frac{N_a}{N_b} \ge T;$
$\frac{N''_b}{N''_a} = \frac{N_b + V''}{N_a - V''} \ge T$

(c) $1 < \frac{N_b}{N_a} < T; \frac{N'_a}{N'_b} = \frac{N_a + V'}{N_b - V'} \ge T;$
$\frac{N''_b}{N''_a} = \frac{N_b + V''}{N_a - V''} \ge T$

(d) $\frac{N_b}{N_a} > T > 1; \frac{N'_a}{N'_b} = \frac{N_a + V'}{N_b - V'} \ge T;$
$\frac{N''_b}{N''_a} = \frac{N_b}{N_a} \ge T$

Fig. 7 Illustration of embedding one bit digital watermark

where Δ is the bin width, $m_{k,1}(i)$ denote the i^{th} modified DFT amplitude in Bin_1 of k^{th} LFR O_k, $m'_{k,1}(i)$ is the modified version of $m_{k,1}(i)$. V' can be computed by using the following mathematical expressions

$$V' = T^*N_b - N_a/1 + T \tag{16}$$

where $N'_a = N_a + V'$, and $N'_b = N_b - V'$.

In case of embedding "0", the procedure is similar. If the embedded bit is "0" and $N_b/N_a \geq T$, no operation is needed. In the case of N_b/N_aT, some selected DFT amplitudes from BIN_2, in the number denoted by V'', will be modified to fall into BIN_1, satisfying the condition $N'_b/N'_a \geq T$. The rule on how to modify the DFT amplitudes from BIN_2 to BIN_1 is referred to Eq. 11

$$m'_{k,2}(i) = m_{k,2}(i) - \Delta,\ 1 \leq i \leq V'' \tag{17}$$

where Δ is the bin width, $m_{k,2}(i)$ denote the i^{th} modified DFT amplitude in Bin_2 of k^{th} LFR O_k, $m'_{k,2}(i)$ is the modified version of $m_{k,2}(i)$. V'' can be computed by using the following mathematical expressions

$$V'' = T^*N_a - N_b/1 + T \tag{18}$$

where $N''_a = N_a - V''$, and $N''_b = N_b + V''$.

By repeating the above procedure, the L-bits digital watermark can be embedded into the LFR O_k.

Finally, the watermarked LFR O^*_k can be obtained by applying the IDFT transform and zero-removing.

Step 8: Obtaining the Watermarked Image

Repeat the step 4-step 7 until all LFRs are performed, and we can obtain the watermarked image F^* by replacing the original LFRs O_k with the watermarked LFRs O^*_k.

6 Watermark detection scheme

Similarly to watermark insertion, the first step for watermark detection is analyzing contents to find LFRs. The watermark is then detected from the LFRs. If the watermark is correctly detected from more than one LFR, we can prove ownership successfully. Our process for watermark detection can be summarized as follows.

Step 1: Extraction of Feature Points

The multi-scale SIFT detector is applied to the watermarked image F^*, and a set of feature points, denoted as $P^*_i\left(i = 1, \cdots, n^*\right)$, is obtained (see Section 2).

Step 2: Construction of Local Feature Region (LFR)

A set of LFRs, denoted as $O^*_k\left(k = 1, \cdots, m^*\right)$, is constructed in accordance with the locations of watermarked image and the feature scales of the feature points $P^*_i\left(i = 1, \cdots, n^*\right)$ (see Section 3).

Fig. 8 The watermarked images obtained by the proposed scheme. **a** Lena (PSNR=52.77 dB); **b** Barbara (PSNR=52.95 dB); **c** Mandrill (PSNR=55.25 dB

Step 3: The DFT Transform of LFR

The LFR $O_k^*\left(k = 1, \cdots, m^*\right)$ is mapped to the block of size $2R_k^* \times 2R_k^*$ by using zero-padding method, where R_k^* is the radius of LFR. After the zero-padding operation has been performed on each LFR O_k^* so as to map the circular region to a square region, the DFT transform is applied to each square consequently and the DFT amplitude spectrum $M_k^* = \left\{m_k^*(l), l = 1, \cdots, N_k^*\right\}$ are obtained.

Step 4: Computing Mean of DFT Amplitude

The statistics mean of DFT amplitude is calculated as the sum of the DFT amplitude over its duration

$$\overline{A}_k^* = \frac{1}{N_k^*} \sum_{l=1}^{N_k^*} m_k^*(l) \tag{19}$$

where $\overline{A}_k^*$ denotes the DFT amplitude mean for LFR O_k^*, and $m_k^*(l)$ is the l^{th} DFT amplitude in LFR O_k^*.

Step 5: Extracting Histogram of DFT Amplitude

The significant DFT amplitudes $B^* = \left[(1-\lambda)\overline{A}_k^*, (1+\lambda)\overline{A}_k^*\right]$ are selected from LFR O_k^*, and the histogram of significant DFT amplitudes $H_k^* = \left\{h_k^*(i) \middle| i = 1, \ldots, , L_k^*\right\}$ is extracted for detecting digital watermark.

Step 6: Digital Watermark Extraction

Our goal is to get an estimation of the hidden bits, $W^* = \left\{w_i^* | i = 1, \cdots, L\right\}$, from the LFR O_k^* at a low error rate. Suppose that the number of DFT amplitudes in the i^{th} two consecutive bins are a^* and b^*. We can extract the hidden watermark bit w_i^* by comparing a^* and b^*,

$$w_i^* = \begin{cases} 1 & if \quad a^* \big/ b^* \geq 1 \\ 0 & \quad otherwise \end{cases} \tag{20}$$

The process is repeated until all hidden watermark bits are extracted. In the extraction, the parameters, L and λ, are beforehand known, so the detection process is blind.

The error named false-alarm probability (no watermark embedded but detected having one) is possible in detection. To eliminate this error, a comparison between the extracted

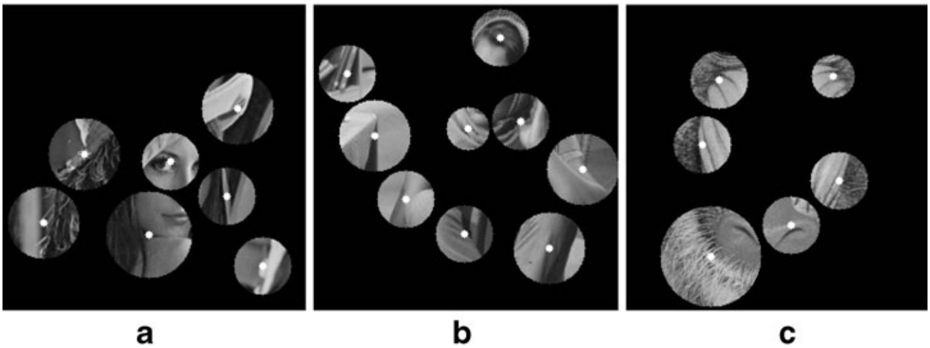


Fig. 9 The detection results from Fig. 8. **a** Lana; **b** Barbara; **c** Mandrill

watermark and the original watermark is necessary. For an unwatermarked image, the extracted bits are assumed to be independent random variables (Bernoulli trials) with the same "success" probability $P_{success}$ (It is called a "success" if the extracted bit matches the embedded watermark bit). We assume the success probability $P_{success}$ = 0.5. Let r be the number of "success" bits in each LFR, and let L be the size of the digital watermark. Then, based on the Bernoulli trials assumption, r is an independent random variable with binomial distribution. A LFR is claimed watermarked if the number of its "success" bits is greater than a threshold. The threshold for a LFR is denoted as T_d. The false-alarm error probability of a LFR is, therefore, the cumulative probability of the cases that $r \geq T_d$. That is:

$$P_{F\_LFR} = \sum_{r=T_d}^{L} (0.5)^L \cdot \left(\frac{L!}{r!(L-r)!} \right) \tag{21}$$

Furthermore, an image is claimed watermarked if at least two LFRs are detected as "success". Under this criterion, the false-alarm probability of an image is:

$$P_{F\_\mathrm{Image}} = \sum_{i=2}^{M} (P_{F\_LFR})^i \cdot (1 - P_{F\_LFR})^{m-i} \cdot \binom{M}{i} \tag{22}$$

where M is the total number of LFRs in an image (Base on our experiences, M=10). Given P_{F_Image}, the T_d can be computed.

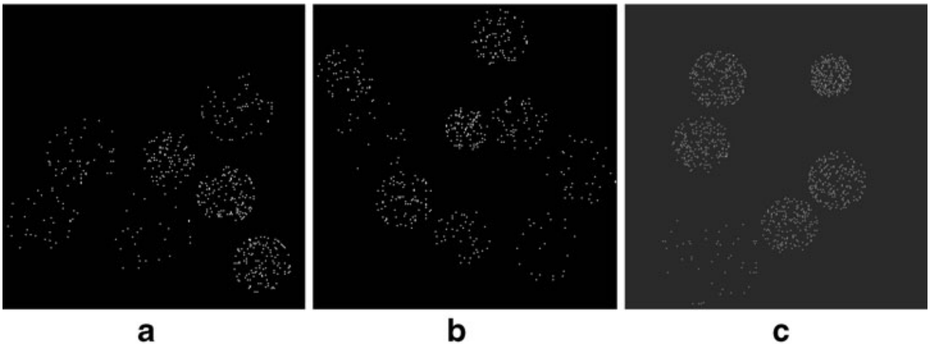


Fig. 10 The difference image between the original image and the watermarked image. **a** Lana; **b** Barbara; **c** Mandrill

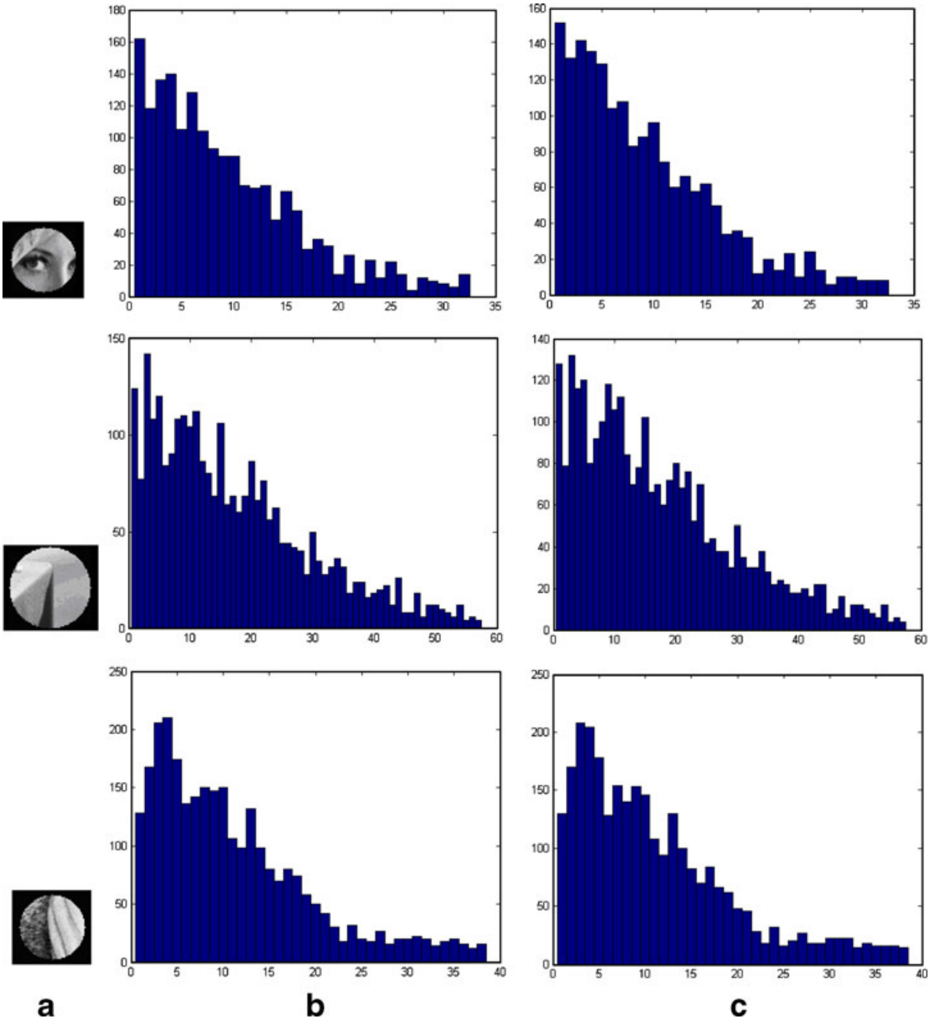


Fig. 11 The histograms of DFT amplitudes for original LFR and watermarked LFR from Lena, Barbara and Mandrill. **a** LFR; **b** The histograms of original LFR; **c** The histograms of watermarked LFR

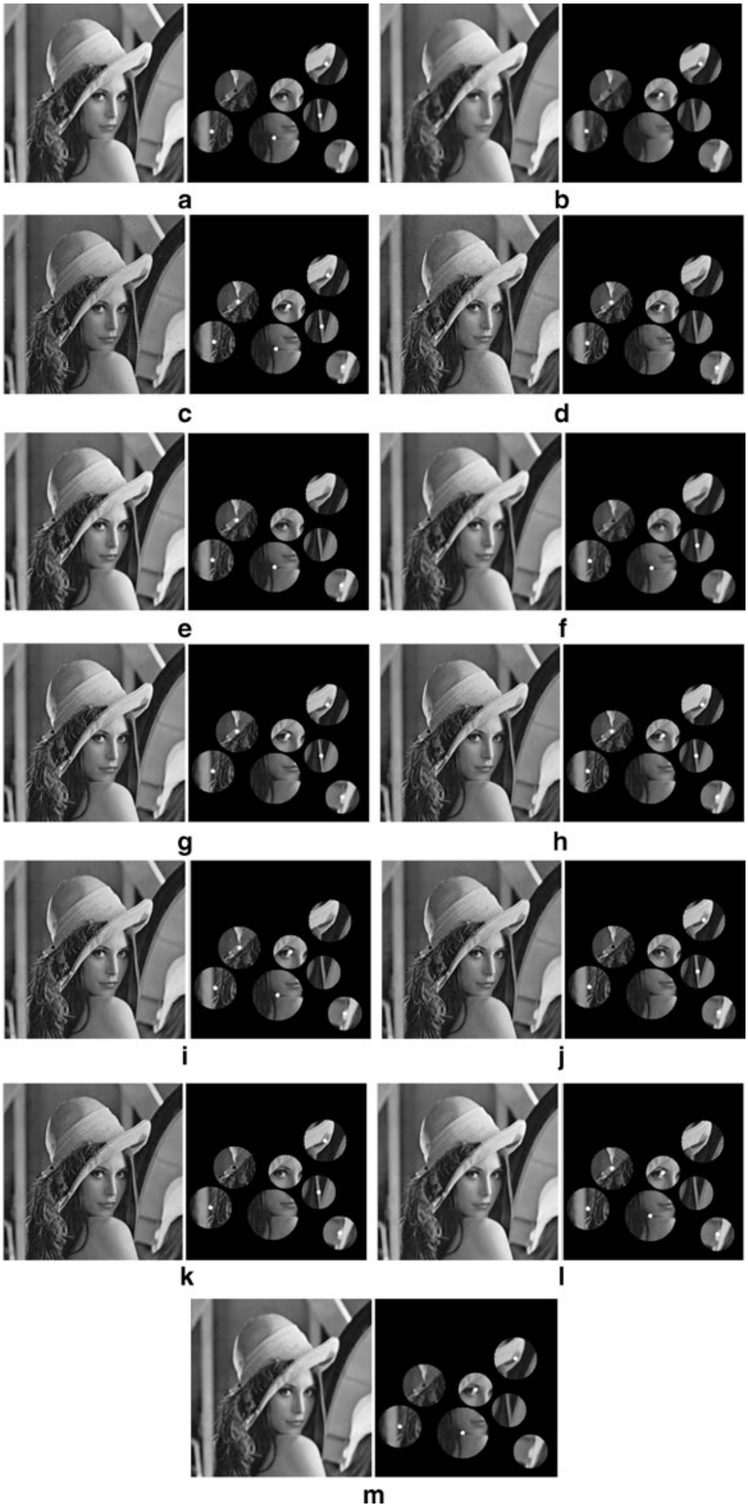
The final detection is claimed “success” when at least two disks are claimed watermarked; otherwise, it “fails”.

7 Simulation results

We test the proposed watermarking scheme on the popular test images 256×256×8bit Lena, Barbara, and Mandrill. A pseudorandom bipolar sequence of size 32-bits is used as

Fig. 12 The simulation results for common image processing operations obtained by the proposed scheme. **a** Median filter (3×3); **b** Median filter (5×5); **c** Salt & Pepper noise (0.02); **d** Guassian noise (0.03); **e** Guassian filter (5×5); **f** Mean filter (3×3); **g** JPEG 70; **h** JPEG 60; **i** JPEG 50; **j** JPEG 40; **k** JPEG 30; **l** Median filter (3×3)+JPEG 90; **m** Median filter (3×3)+JPEG 50

▶


a
b
c
d
e
f
g
h
i
j
k
l
m

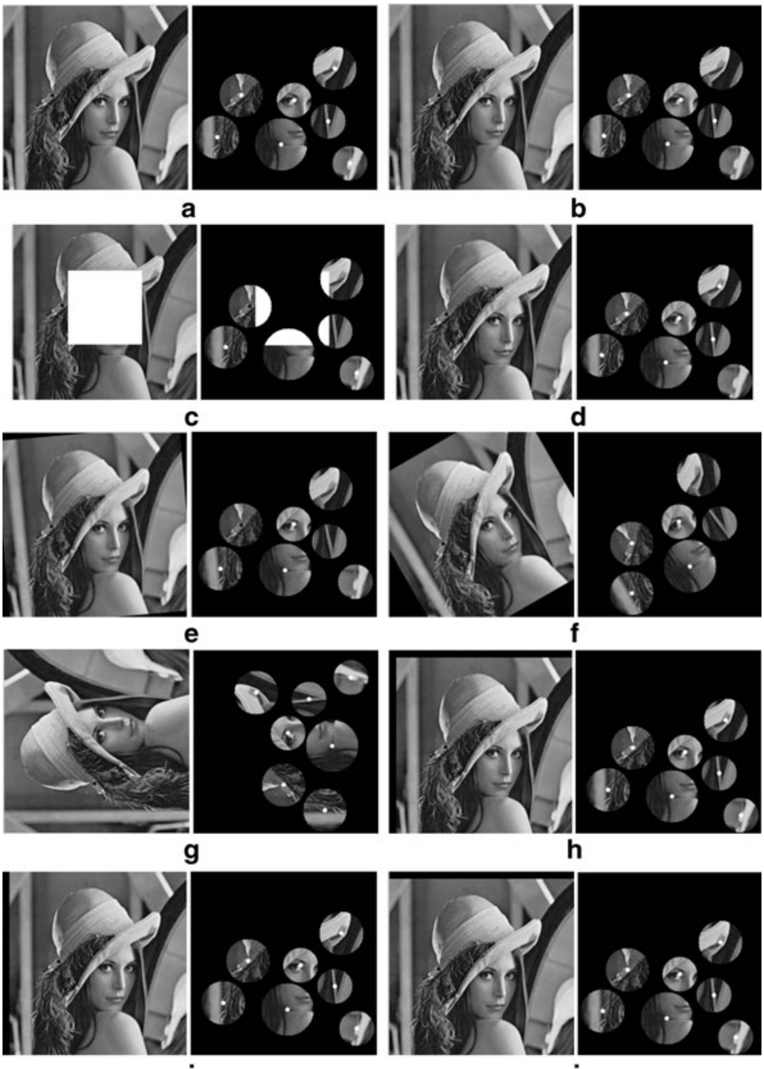


Fig. 13 The simulation results for desynchronization attacks obtained by the proposed scheme. **a** Row/Col Removal (8, 16); **b** Row/Col Removal (16, 8); **c** Cropping (40%); **d** Shearing (X-5% and Y-5%); **e** Rotation 5 degrees+Cropping; **f** Rotation 30 degrees+Cropping; **g** Rotation 90 degree; **h** Translation (X-10 and Y-10); **i** Translation (X-10); **j** Translation (Y-10); **k** Scaling (0.8); **l** Scaling (1.2); **m** Scaling (0.9); **n** Scaling (1.4); **o** Affine transformation; **p** Flip (horizon); **q** Cropping (10%)+JPEG 50; **r** JPEG50+Rotation 5 degrees+ Cropping; **s** Rotation 5 degrees+Scaleing (0.9)+Cropping; **t** Scaling (0.9)+Rotation 5 degree+Translation (X-10 and Y-10)

the watermark pattern. The adaptive positive integer is set to τ=6, and the threshold T=1.5 is selected. The false-alarm Probability is $P_{F\_\mathrm{Image}} \approx 4 \times 10^{-4}$ when the detection threshold $_{Td}$ is set to 20. Besides, the PSNR (Peak Signal-to-Noise Ratio) is used to measure the visual quality of the watermarked images.

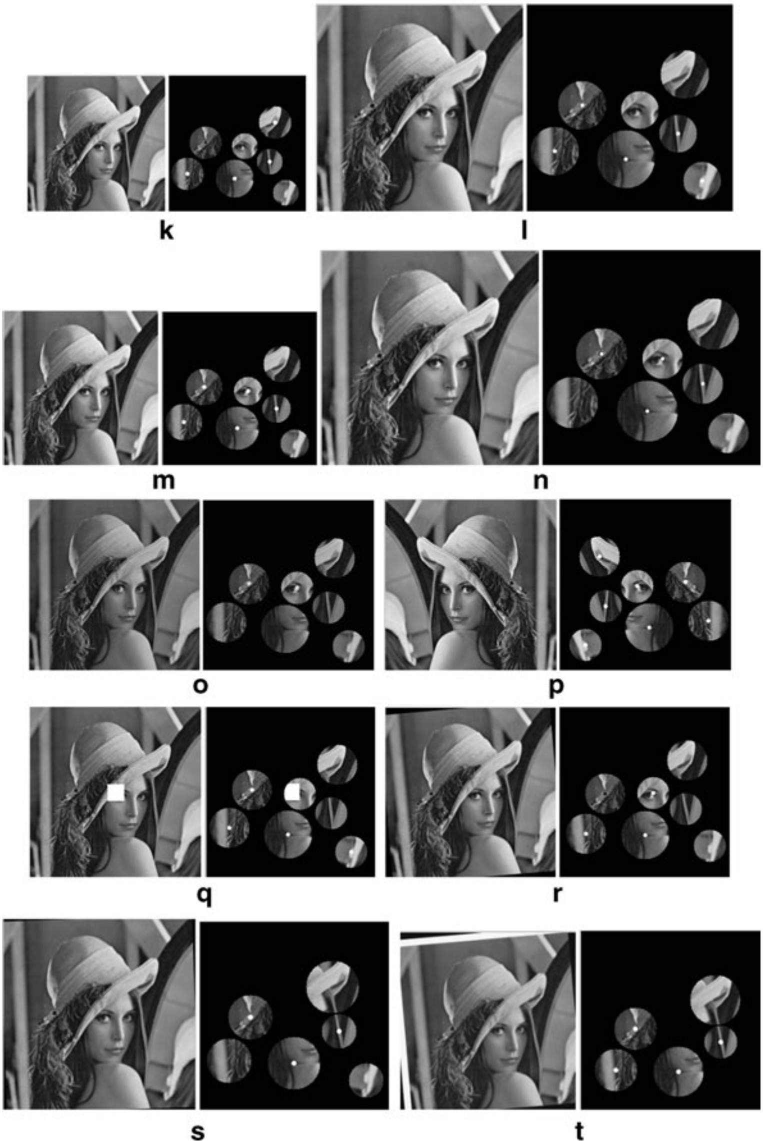


Fig. 13 (continued)

7.1 Invisibility test

As shown in Fig. 8, (a), (b) and (c) are the watermarked images (Lena, Barbara, and Mandrill) obtained by using the proposed scheme. Figure 9 (a), (b) and (c) show the detection results (Notes: the LFRs identified by "white point" denote that they are detected correctly). The number of successfully detect LFRs of the watermarked image is 7, 9 and 6 in Lena, Barbara, and Mandrill, respectively. Figure 10 (a), (b) and (c) give the difference image between the original image and the watermarked image (Lena, Barbara, and Mandrill).

Table 1 The watermark detection rate under common image processing operations

Attack	Lena	Barbara	Mandrill	Attack		Lena	Barbara	Mandrill
Median filter (3×3)	4/7	5/9	4/6	JPEG	70	5/7	5/9	4/6
Median filter (5×5)	3/7	4/9	2/6		60	5/7	3/9	5/6
Salt & Pepper noise (0.02)	7/7	8/9	6/6		50	5/7	4/9	3/6
Guassian noise(0.03)	4/7	4/9	1/6		40	4/7	2/9	3/6
Uniform noise(0.05)	5/7	6/9	3/6		30	4/7	3/9	4/6
Guassian filter (5×5)	4/7	5/9	2/6	Median filter (3×3) +JPEG 90		5/7	6/9	4/6
Mean filter (3×3)	3/7	3/9	3/6	Median filter (3×3) +JPEG 50		4/7	4/9	2/6

7.2 Invariance test for histogram shape

Figure 11 (a), (b) and (c) show the histograms of DFT amplitudes for original LFR and watermarked LFR (The blue marked LFR, See Fig. 4) from Lena, Barbara, and Mandrill. From Fig. 11, we can see that the histograms of DFT amplitudes for original LFR and watermarked LFR are almost the same, and the invariance property of histograms is suitable for embedding watermark.

7.3 Robustness test

Simulation results, which are obtained by the proposed watermarking scheme, for common image processing operations and desynchronization attacks are shown in Figs. 12 and 13 respectively.

Table 2 The watermark detection rate under desynchronization attacks

Attack	Lena	Barbara	Mandrill	Attack	Lena	Barbara	Mandrill
Row/Col Removal (8, 16)	7/7	7/9	6/6	Scaling (0.8)	4/7	3/9	3/6
Row/Col Removal (128,128)	6/7	6/9	6/6	Scaling (1.2)	5/7	4/9	3/6
Cropping (Centered 40%)	2/7	4/9	1/6	Scaling (0.9)	4/7	5/9	3/6
Shearing (X-5% and Y-5%)	6/7	7/9	5/6	Scaling (1.4)	4/7	4/9	2/6
Rotation 5 degrees+Cropping	3/7	5/9	4/6	Affine transformation	3/7	4/9	3/6
Rotation 30 degrees +Cropping	4/7	3/9	2/6	Flip (horizon)	7/7	9/9	6/6
Rotation 90 degree	7/7	8/9	6/6	Cropping (10%)+ JPEG 50	4/7	4/9	3/6
Translation (X-10 and Y-10)	7/7	5/9	5/6	JPEG50+Rotation 5 degrees+Cropping	3/7	4/9	3/6
Translation (X-10)	7/7	6/9	6/6	Rotation 5 degrees+ Scaleing (0.9	3/7	4/9	4/6
Translation (Y-10)	7/7	9/9	5/6	Scaling (0.9) +Rotation 5 degree+Translation (X-10 and Y-10)	4/7	3/9	2/6

Tables 1 and 2 summarize the watermark detection results against common signal processing operations and desynchronization attack respectively. The tables show the ratio between the number of correctly detected watermarked LFR and the number of original embedded watermarked LFR. In this paper, we use the term "*detection rates*" to denote it. Here is the list of the different attacks.

8 Conclusion

Desynchronization attacks, and more precisely local desynchronization attacks, are the Achilles heel for many watermarking schemes. Based on multi-scale SIFT (Scale Invariant Feature Transform) detector and local image histogram shape invariance, a new feature-based image watermarking algorithm robust to both common image processing operations and desynchronization attacks is proposed. There are several key elements in our scheme:

(i) The feature points for host image extracted by using multi-scale SIFT detector are reliable under various attacks. It is helpful for resynchronization between the watermark embedding and detection.
(ii) Based on multi-scale space theory, a size adapted LFRs construction method is developed, which is effective to resist scaling attack.
(iii) By utilizing local image histogram shape invariance, the resilience of the watermark against local desynchronization attacks can be significantly improved.

Drawbacks of the proposed image watermarking scheme are related to its lesser watermark volume. In addition, due to the computation time for the multi-scale SIFT detector, our scheme cannot be used effectively in real-time applications. Future work will focus on eliminating these drawbacks.

References

1. Barni M (2005) Effectiveness of exhaustive search and template matching against watermark desynchronization. IEEE Signal Process Lett 12(2):158–161
2. Bas P, Chassery JM, Macq B (2002) Geometrically invariant watermarking using feature points. IEEE Trans Signal Process 11(9):1014–1028
3. Brown M, Lowe DG (2002) Invariant features from interest point groups. In Proceedings of the British Machine Vision Conference, Cardiff, Wales, September, 2002: 656–665
4. Chakravarti R, Xiannong M (2009) A study of color histogram based image retrieval. The Sixth International Conference on Technology: New Generations, 27–29 April, 2009: 1323–1328
5. Huang H-C, Chen Y-H, Abraham A (2010) Optimized watermarking using swarm-based bacterial foraging. J Inf Hiding Multimed Signal Process 1(1):51–58
6. Lee H-Y et al (2005) Evaluation of feature extraction techniques for robust watermarking. The 4th International Workshop on Digital Watermarking 2005, Siena, Italy, September 15–17, 2005, Lecture Notes in Computer Science 3710, Springer 2005: 418–431
7. Lee H-y, Kim H, Lee H-K (2006) Robust image watermarking using local invariant features. Opt Eng 45 (3):037002(1–11)
8. Lee H-Y, Lee C-h, Lee H-K (2007) Geometrically invariant watermarking: synchronization through circular Hough transform. Multimed Tools Appl 34(3):337–353
9. Lian S, Kanellopoulos D, Ruffo G (2009) Recent advances in multimedia information system security. Informatica 33(1):3–24
10. Liu Y, Zheng D, Zhao J (2007) An image rectification scheme and its applications in RST invariant digital image watermarking. Multimed Tools Appl 34(1):57–84

11. Lowe DG (2004) Distinctive image features from local scale-invariant key points. Int J Comput Vis 60 (2):91–110
12. Motallebi F, Aghaeinia H (2008) RST invariant wavelet-based image watermarking using template matching techniques. Proceedings of the 2008 Congress on Image and Signal Processing 5:720–724
13. Pham V-Q, Miyaki T, Yamaski T, Aizama K (2008) Robust object-based watermarking using feature matching. IEICE Trans Inf Syst E91-D(7):2027–2034
14. Qi H, Zheng D, Zhao J (2008) Human visual system based adaptive digital image watermarking. Signal Process 88(1):174–181
15. Schaffalitzky F, Zisserman A (2002) Multi-view matching for unordered image sets, or "How do I organize my holiday snaps?". Proceedings of the 7th European Conference on Computer Vision-Part I, May 28–31, 2002: 414–431
16. Seo JS, Yoo CD (2006) Image watermarking based on invariant regions of scale-space representation. IEEE Trans Signal Process 54(4):1537–1549
17. Tang CW, Hang HM (2003) A feature-based robust digital image watermarking scheme. IEEE Trans Signal Process 51(4):950–958
18. Wang X-y, Hou L-m, Wu J (2008) A feature-based robust digital image watermarking against geometrical attacks. Image Vis Comput 26(7):980–989
19. Wang X, Wu J, Niu P (2007) A new digital image watermarking algorithm resilient to desynchronization attacks. IEEE Trans Inf Forensics Secur 2(4):655–663
20. Wang S, Yang B, Niu X (2010) A secure steganography method based on genetic algorithm. J Inf Hiding Multimed Signal Process 1(1):28–35
21. Xiang SJ, Huang JW (2007) Histogram-based audio watermarking against time-scale modification and cropping attacks. IEEE Trans Multimed 9(7):1357–1372
22. Xiang S, Kim HJ, Huang J (2008) Invariant image watermarking based on statistical features in the low-frequency domain. IEEE Trans Circ Syst Video Technol 18(6):777–790
23. Xin Y, Liao S, Pawlaka M (2007) Circularly orthogonal moments for geometrically robust image watermarking. Pattern Recognit 40(12):3740–3752
24. Zhang L, Qian G-b, Xiao W-w, Ji Z (2007) Geometric invariant blind image watermarking by invariant Tchebichef moments. Opt Express 15(5):2251–2261
25. Zheng D, Liu Y, Zhao J, El Saddik A (2007) A survey of RST invariant image watermarking algorithms. ACM Comput Surv 39(2):1–91
26. Zheng D, Wang S, Zhao J (2009) RST invariant image watermarking algorithm with mathematical modeling and analysis of the watermarking processes. IEEE Trans Image Process 18(5):1055–1068

Xiang-Yang Wang was born in Tieling, China, in 1965. He is currently a professor with the School of Computer and Information Technology at the Liaoning Normal University, China. He obtained his B.S. degree from the Lanzhou University, China and his M.S. degree from the Jilin University, China, in 1988 and 1995, respectively. His research interests include signal processing and communications, digital multimedia data hiding and information assurance, applications of digital image processing, computer vision. He has published more than 150 journal papers, 20 conference papers, and contributed in 2 books in his areas of interest.

Pan-Pan Niu was born in Panjin, China, in 1983. She is currently pursuing the D.E. degree in the school of Information Science & Technology at the Danlian Maritime University, China. She obtained her B.S. degree and her M.S. degree from the Liaoning Normal University, China, in 2006 and 2009, respectively. Her research interests include signal processing and communications, digital multimedia data hiding and information assurance.

Lan Meng received the B.S. degree from the School of Computer and Information Technology, Liaoning Normal University, China, in 2006, where she is currently pursuing the M.S. degree. Her research interests include image watermarking and signal processing.

Hong-Ying Yang is currently a assistant professor with the School of Computer and Information Technology at the Liaoning Normal University, China. She received her B.S. degree from the Liaoning Normal University, China and her M.S. degree from the Fudan University, China, in 1989 and 1993, respectively. His research interests include signal processing and communications, digital multimedia data hiding.