LDFT-Based Watermarking Resilient to Local Desynchronization Attacks

Huawei Tian, Yao Zhao, *Senior Member, IEEE*, Rongrong Ni, *Member, IEEE*, Lunming Qin, and Xuelong Li, *Fellow, IEEE*

***Abstract*—Up to now, a watermarking scheme that is robust against desynchronization attacks (DAs) is still a grand challenge. Most image watermarking resynchronization schemes in literature can survive individual global DAs (e.g., rotation, scaling, translation, and other affine transforms), but few are resilient to challenging cropping and local DAs. The main reason is that robust features for watermark synchronization are only globally invariable rather than locally invariable. In this paper, we present a blind image watermarking resynchronization scheme against local transform attacks. First, we propose a new feature transform named local daisy feature transform (LDFT), which is not only globally but also locally invariable. Then, the binary space partitioning (BSP) tree is used to partition the geometrically invariant LDFT space. In the BSP tree, the location of each pixel is fixed under global transform, local transform, and cropping. Lastly, the watermarking sequence is embedded bit by bit into each leaf node of the BSP tree by using the logarithmic quantization index modulation watermarking embedding method. Simulation results show that the proposed watermarking scheme can survive numerous kinds of distortions, including common image-processing attacks, local and global DAs, and noninvertible cropping.**

***Index Terms*—Local daisy feature transform (LDFT), robust, watermarking.**

I. Introduction

DIGITAL media widely spread along with the booming development of computer science and the Internet technology. However, unrestricted reproduction and convenient manipulation of digital media cause considerable financial losses to the media creators and the content providers. Digital watermarking is introduced to prevent the aforementioned infringement [1]. Digital watermarking is a process of embedding information into digital multimedia so that the information can be detected for a variety of purposes including copyright protection and digital right management. Digital watermarking has become an active and important area of research. More theories, techniques, and applications of watermarking can be found in [2]–[4].

In the past ten years, attacks against image watermarking systems have become increasingly complicated with the development of watermarking techniques [5]–[9]. The development of attacks starts with common image processing and the following global rotation, scaling, and translation (RST) transforms [10]. Then, other global affine transforms appear, such as shearing and linear geometric transform (LGT). Later, some local transform attacks are proposed, such as random bending attack (RBA) of Stirmark [11] and other RBAs including global bending attacks (GBAs), high-frequency bending (HFB), and random jitter attacks (RJAs) [12], [13]. Recently, two kinds of local transform attacks, reported by Barni *et al.*, are named as constrained local permutation with cancelation and duplication (C-LPCD) attack and Markov random field (MRF) attack [14], [15]. Generally, common image-processing attacks make the watermarking ineffective by reducing its energy rather than introducing synchronization errors. Unlike common image-processing attacks, desynchronization attacks (DAs) not only reduce watermark energy but also break the synchronization between the encoder and the decoder. Therefore, the detector fails to extract the watermark. DAs, especially local transform attacks and cropping, are more challenging to tackle than other types of attacks. There are only a few schemes such as in [16]–[18] which can counteract them.

In this paper, we design a blind watermarking resynchronization scheme which is resilient to various attacks, including local and global DAs and noninvertible cropping. First, we present a new feature transform named local daisy feature transform (LDFT), which is locally RST invariant, and each pixel can be mapped into the LDFT space. Second, we partition the geometrically invariant LDFT space with the binary space partitioning (BSP) tree [19]. Because the location of every single pixel in the feature space is invariable to geometric distortion, the BSP tree which is built using all pixels is robust to various DAs. Moreover, the location of every pixel is fixed under cropping, so the proposed scheme is also robust to cropping. Third, the watermarking sequence is embedded bit by bit into each leaf node of the BSP tree by using the logarithmic quantization index modulation (LQIM) [20] watermarking embedding method.

Manuscript received May 18, 2012; revised October 11, 2012 and January 17, 2013; accepted January 31, 2013. Date of publication March 25, 2013; date of current version November 18, 2013. This work was supported in part by the 973 Program (2011CB302204), by the National Natural Science Foundation of China (61025013, 61073159, 61125106), by the Program for Changjiang Scholars and Innovative Research Team in University (IRT 201206), and by the Shaanxi Key Innovation Team of Science and Technology (2012KCT-04). This paper was recommended by Editor M. Shin.

H. Tian, R. Ni, and L. Qin are with the Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China, and also with Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing 100044, China (e-mail: hwtian@live.cn; rrni@bjtu.edu.cn; lunming.qin@hotmail.com).

Y. Zhao is with the Institute of Information Science, Beijing Jiaotong University, and with State Key Laboratory of Rail Traffic Control and Safety, Beijing 100044, China (e-mail: yzhao@bjtu.edu.cn).

X. Li is with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, Shaanxi, China (e-mail: xuelong_li@opt.ac.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/TCYB.2013.2245415

The remainder of this paper is organized as follows. In Section II, we introduce the problem of watermarking desynchronization and give an overview of existing countermeasures in literature. Section III is devoted to proposing the resynchronization watermarking scheme. In Section IV, we evaluate the performance of the proposed watermarking algorithm in terms of robustness; false-alarm probability; missing probability; and tradeoff among capacity, imperceptibility, and robustness. Experimental results are provided in Section V. Finally, Section VI concludes this paper.

II. Desynchronization VS. Resynchronization

DAs can be classified into two categories: *global DAs* and *local DAs*. In this section, we give an overview of countermeasures between the two kinds of DAs and existing watermarking resynchronization schemes in literature.

A. Resynchronization Resisting Global DAs

Global DAs include RST, LGT shearing, and other affine transforms. Because these attacks are global and the attack parameters are not complicated, most of existing countermeasures focus on these DAs. Successful resynchronization methods generally rely on the following.

1) Exhaustive search: one obvious solution is to exhaustively search for the watermark in the space including a set of acceptable attack parameters. The computational cost in the large search space and the dramatic increase of the false-alarm probability during the search process are concerns of the exhaustive search [21].
2) RST-invariant domains: Ruanaidh and Pun [22] exploited the fact that an image's Fourier–Mellin transform domain is invariant to global RST transformations to embed a watermark shown to be robust against these attacks.
3) RST-invariant moments: these methods utilize the RST-invariant moments of the image, such as geometrical moments [23] and Zernike moments [24], to prevent the desynchronization.
4) Using template: in this kind of watermarking schemes, additional templates are intentionally embedded into original images [25]. As anchors for the alignment, these templates assist the watermark resynchronization in detection.
5) Image normalization: the algorithm achieves its robustness by both embedding and detecting the watermark message in the normalized image [26].
6) Using feature points or regions: the basic strategy is to bind a watermark with the geometrically invariant image features, so the detection of the watermark can be conducted with the help of features [17], [27], [28]. Moreover, since several copies of the watermark are embedded in a number of local regions formed by feature points, such watermarking methods can resist cropping.

Feature-based image watermarking schemes, which are also called the second-generation schemes [29], have attracted great attention in recent years. Bas *et al.* [27] used Harris detector to extract feature points and then divided the image into a set of disjoint triangles by using the Delaunay tessellation, in which both watermark embedding and detection are conducted. However, the feature points can hardly survive under scaling distortion [30]. Then, Mexican hat wavelet filtering is used for feature point extraction, and watermark is embedded in the normalized local regions centered at the feature points in [28]. The Mexican hat wavelet method is stable under noiselike processing, yet it is sensitive to some affine transforms [31], and the size of local regions remains fixed so that this scheme is vulnerable to affine transforms. To further enhance the robustness of the feature-based watermarking, scale-space theory was applied for feature point extraction [17], [32]–[35]. In [34], the feature points were extracted by using the Harris–Laplace detector. Based on scale-space theory, a size adapted local region construction method was developed, which is effective in resisting the scaling attack. However, the scheme is sensitive to affine transformations. To this end, affine-invariant feature detector was used to extract feature points. A feature selection procedure based on the graph theoretical clustering algorithm is applied to obtain a set of stable and nonoverlapped affine-invariant regions, which were utilized for watermark embedding and detection in [35]. However, feature-based image watermarking schemes suffer from three drawbacks as follows.

1) The performance of feature detectors is difficult in satisfying the watermarking system. The feature point detection [36] is the linchpin, upon which a watermarking scheme's success or failure depends. Therefore, it must have the following properties. Above all, it must be robust to many kinds of common image-processing attacks. Moreover, it must perform well in RST invariance and affine invariance. Furthermore, it must have high repeatability. Finally, the localization of the feature point must be sufficiently accurate.
2) It is difficult to select feature points for watermarking from all detected feature points. Feature point selection for obtaining nonoverlapped local regions plays an important role in achieving the desired goal of the robust watermarking scheme. In watermark extraction, if more feature points are selected, the false-alarm probability will increase; if some feature points are lost, the missing probability will increase.
3) Destroying the harmonization between local regions and the entire image. The watermark message is only embedded repeatedly in local regions of an image, so it brings on a low imperceptibility in local regions, even though the peak signal-to-noise ratio (PSNR) value between the original image and the watermarked image is very high. Moreover, it destroys the harmonization between local regions and the entire image.

B. Resynchronization Resisting Local DAs

Local DAs usually unite some kinds of localized attacks, such as the C-LPCD attack, the MRF attack, and the RBAs including GBA, HFB, and RJA. In the case that the attacks are unknown, it dramatically increases the number of attack parameters and complicates the resynchronization procedure.

To the best of our knowledge, few schemes are claimed to counteract C-LPCD attack and MRF attack, and only a few schemes [16]–[18] can counteract RBAs. In [16], the histogram shape and the mean in the Gaussian filtered low-frequency component of images are mathematically invariant to RST and statistically resistant to cropping and RBAs. Xiang *et al.*

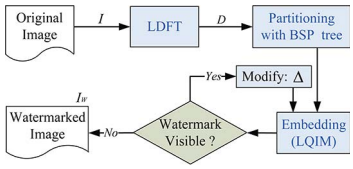


Fig. 1. Watermark embedding framework.

[16] used the two statistical features to design watermarking algorithm against different DAs. Dugelay *et al.* [18] developed a method of adding a predefined additional information to the useful message bits at the insertion step. These additional bits are labeled as resynchronization bits or reference bits, and they are modulated in the same way as the information bits. During the extraction step, the reference bits are used as anchor points to estimate and compensate for small local and global geometrical distortions. Therefore, the method is only robust to small local DAs, such as RBAs of Stirmark [11].

III. Resynchronization Watermarking Scheme

Fig. 1 is an overview of our proposed watermark embedding scheme. The watermark embedding process consists of three main steps: 1) constructing the feature space with LDFT; 2) partitioning the LDFT space with the BSP tree; and 3) embedding watermark information by LQIM watermark embedding method. The watermark extracting process resembles watermark embedding, which comprises three main steps: 1) constructing the feature space of LDFT; 2) partitioning the LDFT space with the BSP tree; and 3) watermark extraction.

A. Proposed New Feature Transform: LDFT

Most image watermarking resynchronization schemes in literature can survive individual global attacks, but only a few are resilient to local DAs. The main reason is that robust features in watermarking schemes are only globally invariable rather than locally invariable. In this section, we present a novel local invariant feature transform, named LDFT. Our LDFT is inspired by the DAISY descriptor proposed in [37], but it is more robust. Unlike the DAISY descriptor which is not RST invariant in theory, LDFT is not only globally RST invariant but also locally RST invariant.

The following are the major steps of computation used to generate the LDFT.

Step 1) *Compute the characteristic scale map.*

For an input image I, LoG operator is used to obtain the characteristic scale map S, which varies proportionally with image scaling. The map S will be used to control the size of local regions for LDFT in step 5). The same content region is used for LDFT even if the image is zoomed and the LDFT is scaling invariant. The LoG for pixel $f(x,y)$ in image I is defined as [38]

$$\mathrm{LoG}(x,y,\delta_i) = \delta_i^2 \left| L_{xx}(x,y,\delta_i) + L_{yy}(x,y,\delta_i) \right| \quad (1)$$

where $L(x,y,\delta_i) = G(x,y,\delta_i) * f(x,y)$, $L_{xx}(x,y,\delta_i) = \partial^2 L(x,y,\delta_i)/\partial x^2$, and $L_{yy}(x,y,\delta_i) = \partial^2 L(x,y,\delta_i)/\partial y^2$. $G(x,y,\delta_i)$ is the Gaussian kernel with standard deviation δ_i and mean zero. Given a set of scales δ, the characteristic scale $S(x,y)$ is the scale at which LoG attains a local maximum as the following equation:

$$S(x,y) = \arg\max_{\delta_i \in \delta} \{\mathrm{LoG}(x,y,\delta_i)\}. \quad (2)$$

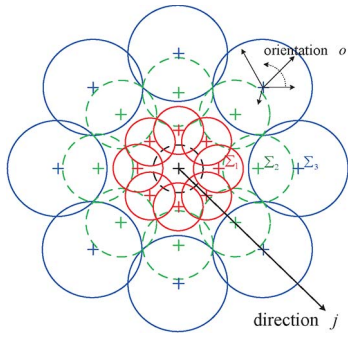


Fig. 2. DAISY descriptor: each circle represents a region where the radius is proportional to the standard deviations of the Gaussian kernels. The "+" sign represents the location where we sample the convolved orientation map center as a pixel location where the descriptor is calculated. Adapted from [37].

Step 2) *Compute the orientation maps.*

The orientation maps are defined as $G_o = max(\partial I/\partial o, 0)$, where I is an image and $o (o \in [1, H])$ is the orientation of its derivative. For an input image, we compute H orientation maps for every quantized direction. $G_o(u,v)$ is equal to the image gradient at location (u,v) for direction o if it is bigger than zero; otherwise, it is equal to zero. This can preserve the polarity of the intensity change.

Step 3) *Compute the convolved orientation maps.*

The convolved orientation maps are defined as $G_o^{\Sigma} = G_{\Sigma} * \max(\partial I/\partial o, 0)$, where G_{Σ} is a Gaussian kernel and o is the orientation of the derivative. Each orientation map is then convolved with Gaussian kernels of different standard deviation Σs to obtain convolved orientation maps for differently sized regions.

Step 4) *Normalize the convolved orientation maps of each sample point.*

As depicted in Fig. 2, the LDFT consists of a vector made of values from the convolved orientation maps located on concentric circles centered at the location, where the amount of Gaussian smoothing is proportional to the radii of the circles. Let $\mathbf{h}_{\Sigma}(x,y)$ be the vector made of the values at location (x,y) in the orientation maps after convolution by a Gaussian kernel of standard deviation Σ

$$\mathbf{h}_{\Sigma}(x,y) = \left[G_1^{\Sigma}(x,y), \ldots, G_o^{\Sigma}(x,y), \ldots, G_H^{\Sigma}(x,y) \right] \quad (3)$$

where G_1^{Σ}, G_2^{Σ}, and G_H^{Σ} denote the Σ-convolved orientation maps. Note that $H = 36$ when 36 orientations are considered. We normalize these vectors to unit norm and denote the normalized vectors by $\tilde{\mathbf{h}}_{\Sigma}(x,y)$

$$\tilde{\mathbf{h}}_{\Sigma}(x,y) = \left[\tilde{G}_1^{\Sigma}(x,y), \ldots, \tilde{G}_o^{\Sigma}(x,y), \ldots, \tilde{G}_H^{\Sigma}(x,y) \right]. \quad (4)$$

Step 5) *Compute Euclidean distance.*

$\tilde{\mathbf{h}}_{\Sigma}(x,y)$ is a normalized vector, so $\sum_{o=1}^{H} (\tilde{G}_o^{\Sigma}(x,y))^2/H = 1/H$. We compute the Euclidean distance between the normalized vector $\tilde{\mathbf{h}}_{\Sigma}(x,y)$ and the vector $[1/H, \ldots, 1/H, \ldots, 1/H]$ and denote the Euclidean distance by $E_{\Sigma}(x,y)$. Note that, if

all values of vector $\tilde{\mathbf{h}}_\Sigma(x,y)$ are zero, $E_\Sigma(x,y)=1/\sqrt{H}$. If Q represents the number of circular layers, the initial LDFT $\hat{\mathcal{D}}(x_0,y_0)$ for pixel point (x_0,y_0) is defined as the concatenation of E_Σ

$$\begin{aligned}\hat{\mathcal{D}}(x_0,y_0) &= [E_{\Sigma_1}(x_0,y_0),\\ &E_{\Sigma_1}(l_1(x_0,y_0,R_1)),\ldots,E_{\Sigma_1}(l_T(x_0,y_0,R_1)),\\ &E_{\Sigma_2}(l_1(x_0,y_0,R_2)),\ldots,E_{\Sigma_2}(l_T(x_0,y_0,R_2)),\\ &\ldots\\ &E_{\Sigma_Q}(l_1(x_0,y_0,R_Q)),\ldots,E_{\Sigma_Q}(l_T(x_0,y_0,R_Q))] \quad (5)\end{aligned}$$

where $l_j(x_0,y_0,R)$ is the location with distance R from (x_0,y_0) in the direction given by j when the direction is quantized into T values. $R=k\times S(x_0,y_0)$, $R_1=R$, $R_2=2R$, and $R_Q=Q\times R$, where S is the characteristic scale map computed in step 1). Therefore, LDFT performs on the same content region even if the image is scaled and the LDFT is scaling invariant. k is a positive number, which is used to adjust the radius of the local region. If k is too large, the LDFT will cover the most part of the image and will not obtain the local effect. If k is too small, the LDFT will cover a very small region; each value of the LDFT vector will be very close. The distinctiveness of the LDFT vector will reduce. We set it as $k=3$ in all experiments, which obtains a balance between local effect and distinctiveness. On the other hand, in the watermarking system, k can be set as a secret key, and the receiver who does not know it will not be able to generate accurate watermarking information.

Step 6) *Orientation assignment.*

By assigning a consistent orientation to the LDFT of each pixel based on local image properties, the LDFT can be represented relative to this orientation and therefore achieves invariance to image rotation. We propose an approach resembling the histogram of oriented gradients (HOG) [39] to gain the consistent orientation. An orientation histogram is formed from convolved orientation maps of sample points within a region around the center pixel (x_0,y_0). The orientation histogram has H bins covering the 360° range of orientations. Note that $H=36$ when 36 orientations are considered. Each sample is added to the histogram. We get the histogram of the oth bin with the following equation:

$$\begin{aligned}\mathcal{H}_o(x_0,y_0) &= \tilde{G}_o^{\Sigma_1}(x_0,y_0)\\ &+\tilde{G}_o^{\Sigma_1}(l_1(x_0,y_0,R_1))+\ldots+\tilde{G}_o^{\Sigma_1}(l_T(x_0,y_0,R_1))\\ &+\tilde{G}_o^{\Sigma_2}(l_1(x_0,y_0,R_2))+\ldots+\tilde{G}_o^{\Sigma_2}(l_T(x_0,y_0,R_2))\\ &+\ldots\\ &+\tilde{G}_o^{\Sigma_Q}(l_1(x_0,y_0,R_Q))+\ldots+\tilde{G}_o^{\Sigma_Q}(l_T(x_0,y_0,R_Q)). \quad (6)\end{aligned}$$

The maximum peak in the orientation histogram corresponds to dominant direction, denoted by J

$$J=\arg\max_{o\in[1,H]}\{\mathcal{H}_o(x_0,y_0)\}. \quad (7)$$

The initial LDFT $\hat{D}(\cdot)$ is then recomposed in the clockwise direction from the dominant direction J in each circular layer. Now, the LDFT becomes

$$\begin{aligned}D(x_0,y_0)=[&E_{\Sigma_1}(x_0,y_0),\\ &E_{\Sigma_1}(l_J(\cdot)),\ldots,E_{\Sigma_1}(l_T(\cdot)),E_{\Sigma_1}(l_1(\cdot)),\ldots,E_{\Sigma_1}(l_{J-1}(\cdot)),\\ &E_{\Sigma_1}(l_J(\cdot)),\ldots,E_{\Sigma_2}(l_T(\cdot)),E_{\Sigma_2}(l_J(\cdot)),\ldots,E_{\Sigma_2}(l_{J-1}(\cdot)),\\ &\ldots\\ &E_{\Sigma_2}(l_J(\cdot)),\ldots,E_{\Sigma_2}(l_T(\cdot)),\ldots,E_{\Sigma_Q}(l_1(\cdot)),\ldots,E_{\Sigma_Q}(l_{J-1}(\cdot))]\end{aligned} \quad (8)$$

where $E_{\Sigma_q}(l_j(\cdot))=E_{\Sigma_q}(l_j(x_0,y_0,R_q))$, $j\in[1,T]$, and $q\in[1,Q]$. Note that J is the dominant direction, which is the starting direction. Thus, the LDFT consists of $1+T\times Q$ values, which are extracted from $1+T\times Q$ locations, respectively. The parameters of LDFT including T, Q, and H will be discussed in detail in Section IV-D.

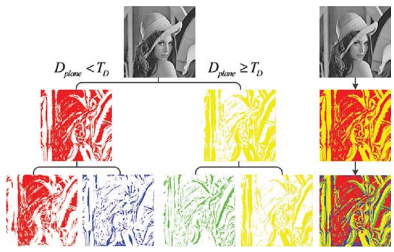


Fig. 3. BSP tree of image Lena. The feature space of Lena is partitioned into four subspaces. Every pixel of "Lena" displays on the node or leaf node which corresponds to its subspace.

B. Partitioning LDFT Space Using the BSP Tree

The accurate partitioning of multidimensional feature space is usually difficult due to the large number of feature dimensions and their overlapping distribution in space. Most of existing methods rely on clustering techniques [40]–[42]. However, these methods do not resist image cropping attack. Our previous method based on k-means clustering [43] is robust against image cropping. However, the watermarking method is semiblind because the centroids of clusters must be sent to the extractor.

In this section, we adopt the BSP tree [19] for partitioning the LDFT space. BSP tree construction is a process which takes a subspace and partitions it by any hyperplane that intersects the interior of that subspace. It generates two new subspaces that can be further partitioned by recursive processes. The algorithm to build a BSP tree for LDFT space partitioning consists of three stages.

1) *Select a partition plane*: $D_i(\cdot)=T_D$ can be regarded as the ith partition plane, where $D_i(\cdot)$ is the ith feature value of the LDFT and $i\in[1,1+T\times Q]$. The partition plane of a space can no longer be used for the subspace. To enhance security, the order of the partition plane can be scrambled by a random sequence with a key K_p. In order to achieve good results for constructing a balanced and robust BSP tree, where each leaf contains roughly the same number of pixels, we set the threshold T_D as the mean value of all $D_i(\cdot)$ of pixels in the original image. The threshold T_D will be discussed in detail in Section IV-A.
2) *Partition the set of space by the plane*: If the subset is under the partition plane and the value of the feature vector is less than T_D, the subset is on the left ramification of the BSP tree; otherwise, the subset is on the right ramification, as shown in Fig. 3.

3) *Recur with each of the two new subsets*: Repeat steps 1) and 2) from left to right until the number of leaf nodes N_{leaf} is equal to the length of watermark sequence N_w.

The BSP is a complete binary tree. The relationship between the number of partition planes N_{pp} and the number of leaf nodes N_{leaf} is $2^{N_{pp}} \geq N_{\text{leaf}}$. As shown in Fig. 3, the feature space of the Lena image is partitioned into four subspaces by using the BSP tree, where two feature values of the LDFT $D(\cdot)$ are used for partitioning.

C. Watermark Embedding

After the BSP tree is built according to the length of the watermark sequence N_w, the watermarking sequence W is embedded bit by bit into each leaf node from left to right by using LQIM [20]. The LQIM is a quantization-based data hiding method. Inspired by μ-law concept, the host signal is transformed into logarithmic domain using a compression function. The watermark data are embedded into the transformed signal using uniform quantization, and then, the quantized signal is transformed into the original domain using inverse function. Due to the logarithmic function, smaller step sizes are devoted to smaller amplitudes and vice versa. Compared with UQIM [44], the LQIM poses perceptual advantages that lead to stronger watermark insertion.

For leaf node n, according to the corresponding watermark bit w_n, $w_n \in \{0, 1\}$, each pixel $f(x, y)$ is quantized with a quantizer $LQ(\cdot; w)$ as

$$f_w(x, y) = LQ\left(f(x, y); w_n\right). \tag{9}$$

Concretely, the pixel value $f(x, y)$ must be transformed first by using the following compression function:

$$C(x, y) = \frac{\ln\left(1 + \mu \frac{f(x,y)}{X_s}\right)}{\ln(1 + \mu)}, \quad \mu > 0, X_s > 0 \tag{10}$$

where μ is a parameter defining the compression level and X_s is the parameter that scales the pixel values of the image. The best X_s value is the value which spreads most of the host signal samples into the range [0, 1]. The details of selecting the optimum μ and X_s can be found in [20]. In all of our experiments, the two parameters are chosen as $\mu = 8.25$ and $X_s = 250$. The transformed signal $C(x, y)$ is then used for data embedding. In this regard, the transformed signal $C(x, y)$ is quantized uniformly by the UQIM [44] watermark embedding method

$$C_w(x, y) = UQ\left(C(x, y); w_n\right) \tag{11}$$

where $UQ(\cdot; w)$ is a uniform scalar quantizer with a step Δ and the quantizer set consists of two quantizers shifted by $\Delta/2$ with respect to each other. Δ is used to control the embedding distortion.

The quantized pixel value is then expanded to obtain the watermarked pixel value as follows:

$$f_w(x, y) = \frac{X_s}{\mu}\left[(1 + \mu)^{C_w(x,y)} - 1\right]. \tag{12}$$

After every pixel in each leaf node is quantized according to the corresponding watermark bit w_n, the watermark embedding process is completed.

D. Watermark Extraction

The watermark extracting process, similar to watermark embedding, consists of three steps: 1) constructing the feature space with LDFT; 2) building the BSP tree along with partitioning the LDFT space; and 3) extracting the watermark.

For each pixel $f'(x, y)$ in leaf node n of the attacked image, determine the embedded watermark bit with the Euclidean distance decoder. Bits 0 and 1 are embedded in the attacked pixel value $f'(x, y)$ using the LQIM embedding method, resulting in $f'_0(x, y)$ and $f'_1(x, y)$, respectively. The watermark data can be extracted by the following equation:

$$\hat{w} = \arg \min_{i \in \{0,1\}} \left\| f'(x, y) - f'_i(x, y) \right\|^2 \tag{13}$$

where $\hat{w}$ is the extracted watermark data.

When desynchronization or common image-processing attacks occur, even in the same leaf node, some pixels are detected to embed bit 1, and some are 0. Let $Num_1(n)$ and $Num_0(n)$ denote the number of pixels hiding bit 1 and bit 0 in leaf node n, respectively. The nth bit of the watermark sequence $\hat{W}$, denoted by $\hat{w}_n$, is extracted as

$$\hat{w}_n = \begin{cases} 1, & \text{if } Num_1(n) \geq Num_0(n) \\ 0, & \text{if } Num_1(n) < Num_0(n). \end{cases} \tag{14}$$

In this section, we propose a new image watermarking resynchronization scheme that shuns or mitigates the aforementioned three drawbacks of the feature-based method in Section II.

IV. Performance Analysis and Parameter Selection

In this section, we evaluate the performance of the proposed watermarking algorithm in terms of robustness; false-alarm probability; missing probability; and tradeoff among capacity, imperceptibility, and robustness. According to these analyses, we give the method of parameter selection of LDFT, BSP tree, and watermarking embedding and extraction.

A. Robustness Analysis

The geometrical invariance of LDFT includes two aspects, namely, scaling invariance and rotation invariance. First, the characteristic scale which varies proportionally with the image scale is used to determine the radius of LDFT. Therefore, LDFT covers the same content even if the image is zoomed or rotated. Second, an approach resembling the HOG is proposed to gain the dominant direction. The initial LDFT can be recomposed in the clockwise direction from the dominant direction. Accordingly, LDFT can achieve rotation invariance.

The element in the LDFT vector will change slightly under various DAs and image-processing attacks. The alteration of the element value is the linchpin, upon which the robustness

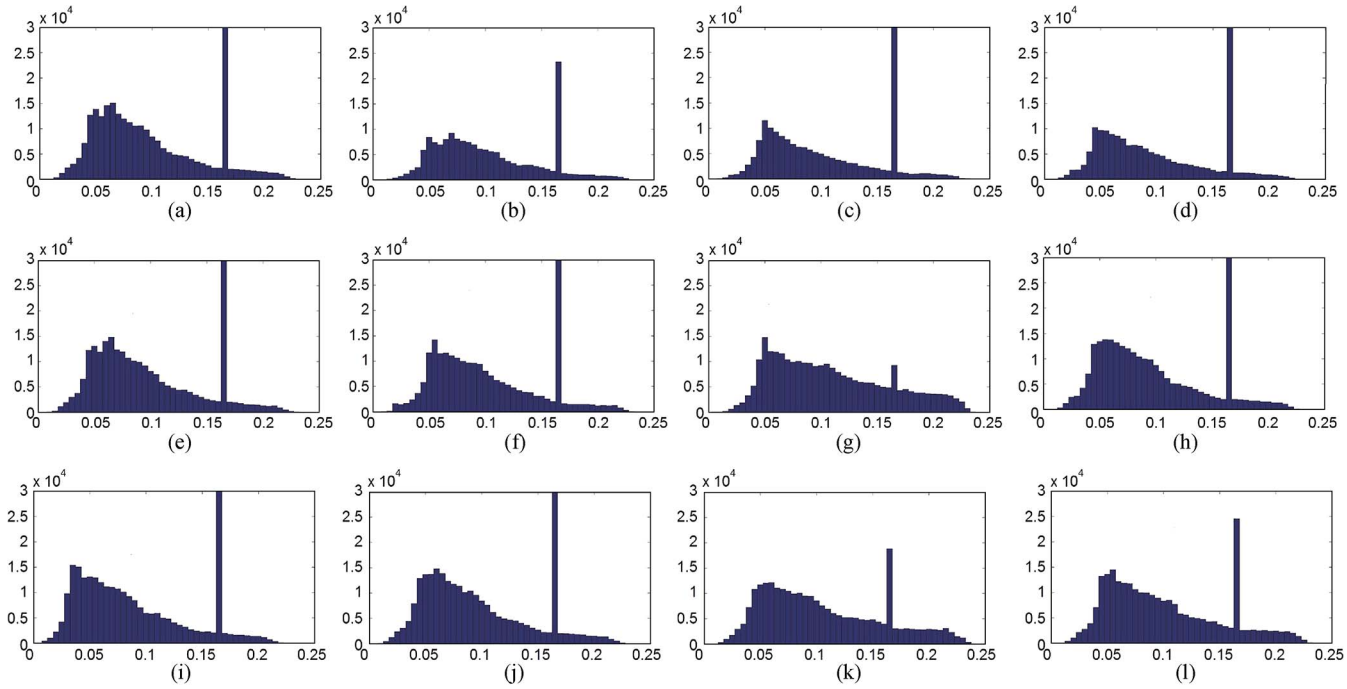


Fig. 4. Distribution of D_3^{50} for every pixel in the original "Lena" image and various desynchronization attacked "Lena" images. The size of "Lena" is 512 × 512 pixels. The x-axis is the value of D_3^{50}, and the y-axis is the amount. (a) Original "Lena" image. (b) Scaling with coefficient is 0.8. (c) Rotated by 15° plus cropping. (d) 15% cropping. (e) Removed 17 rows and 5 columns. (f) Shearing with $x = 5\%$ and $y = 5\%$. (g) LGT with (1.013, 0.008, 0.011, and 1.008). (h) GBA with $factor = 5$. (i) HFB with $factor = 0.5$. (j) RJA with $factor = 0.5$. (k) C-LPCD with $widow = 4$ and $level = 4$. (l) MRF with standard deviation is 5, and level is 5.

of the proposed watermarking scheme depends. In LDFT space partitioning, a threshold T_D is used to partition feature space. There are two main reasons for setting T_D as the mean value of $D_i(\cdot)$ of all pixels in the original image, where $D_i(\cdot)$ is the ith feature value of the LDFT and $i \in [1, 1 + T \times Q]$. First, the mean value will achieve good results for constructing a balanced and robust BSP tree, where each leaf contains roughly the same number of pixels. Second, the mean value is a stable value which can contribute to constructing a robust BSP tree. If the element of a particular pixel alters across from one side of T_D to the other, the pixel will be partitioned into another feature subspace. The wrong partition may result in false watermark extraction. Therefore, we discuss the stability of the element in the LDFT vector.

We take an element value D_3^{50} in the LDFT vector of "Lena" as an example. D_3^{50} corresponds to the sample point at the 3rd circular layer and 50° orientation of LDFT. The parameters of LDFT are set as $T = 36$, $Q = 3$, $H = 36$, and $k = 3$. The distribution of D_3^{50} for every pixel in the original "Lena" is shown in Fig. 4(a). For many pixels in the smooth regions of "Lena", if all values of vector $\tilde{\mathbf{h}}_\Sigma(x, y)$ are zero, $E_\Sigma(x, y) = 1/\sqrt{H}$. Therefore, some values of D_3^{50} are $1/\sqrt{H} \approx 0.1667$, and there is a singular bin in Fig. 4(a). Fig. 4 shows the alteration of distribution of D_3^{50} for every pixel in the "Lena" under various DAs, including rotation, scaling, cropping, randomly removing rows and columns, shearing, LGT, GBA, HFB, RJA, C-LPCD, and MRF. If the partitioning threshold is set as $T_D = 0.1$, which is the mean of all D_3^{50} of "Lena", most values will not alter across the threshold T_D, as shown in Fig. 4. Therefore, the distribution of D_3^{50} is very robust under the aforementioned 11 kinds of DAs. Obviously, the stability of the element in the LDFT vector is very good. Hence, the proposed watermarking scheme based on LDFT is robust enough to counteract various DAs.

B. False-Alarm Probability and Missing Probability

Two types of error measures in the watermark extraction are used: the *false-alarm probability* (no watermark embedded but one extracted) and the *missing probability* (watermark embedded but none extracted). We perform the performance analysis as follows.

1) False-Alarm Probability: The false-alarm error occurs when the watermark is supposed to be detected in a nonwatermarked image. The false-alarm probability of the watermarking scheme P_{fa} can be calculated by

$$P_{fa} = \sum_{n=\xi}^{N_w} (0.5)^{N_w} \cdot \frac{N_w!}{n!(N_w - n)!} \tag{15}$$

where N_w is the length of the watermark sequence and ξ is the threshold used during judging for the presence of the watermark. The decoding bit error rate (BER) B, defined as the ratio between the number of incorrectly decoded bits and the total number of embedded bits, is $(N_w - \xi)/N_w$. Fig. 5 shows the relationship between the length of the watermark sequence and the false-alarm probability with $B = 0$, $B = 0.1$, and $B = 0.15$, respectively.

2) Missing Probability: In order to embed N_w bit watermark, the image must be partitioned into N_w leaf nodes by the BSP tree. Let N_p denote the number of pixels in a leaf node, $N_p \approx (C_1 \times C_2)/N_w$, where $C_1 \times C_2$ is the size of the

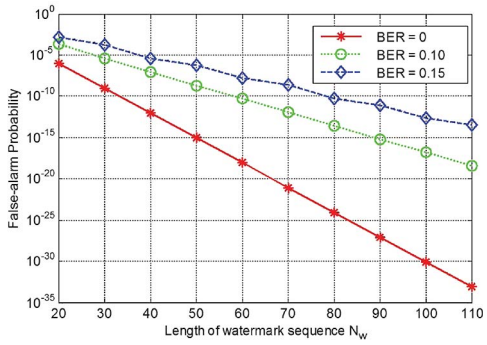


Fig. 5. False-alarm probability for the nonwatermarked image with $B = 0$, $B = 0.1$, and $B = 0.15$, respectively.

image. In an attacked watermarked image, the extracted bits from every pixel using LQIM are assumed to be independent Bemoulli random variables with the same "success" probability P_s. It is called a "success" if the extracted bit matches the embedding watermark bit. When the attacks on watermarked image are very slight, $P_s \approx 100\%$. Contrarily, when the attacks on watermarked image are very severe, $P_s \approx 50\%$. At the worst, $P_s = 50\%$. The success extraction probability of ζ bits in a BSP tree leaf node of N_p watermarked pixels is

$$P_\zeta = P_s^\zeta \cdot (1 - P_s)^{N_p - \zeta} \cdot \frac{(N_p)!}{\zeta!(N_p - \zeta)!}. \tag{16}$$

A leaf node is claimed watermarked if the number of its matching bits is greater than a threshold t_s. In the proposed watermarking scheme, it must meet the conditions of $t_s \geq N_p/2$, which means the number of matched bits is larger than the number of mismatched bits in a leaf node. The success extraction probability of a leaf node N_p is the cumulative probability of the case when $\zeta \geq t_s$

$$P_{\text{leaf}} = \sum_{\zeta = t_s}^{N_p} P_\zeta. \tag{17}$$

Furthermore, an image is claimed watermarked if at least ξ leaf nodes are detected as hiding watermark. In this case, BER is lower than $(N_w - \xi)/N_w$. Therefore, the missing probability of an image is

$$P_m = 1 - \sum_{i=\xi}^{N_w} (P_{\text{leaf}})^i \cdot (1 - P_{\text{leaf}})^{N_w - i} \cdot \begin{pmatrix} N_w \\ i \end{pmatrix}. \tag{18}$$

Users can set the thresholds of P_{fa} and P_m according to their practical application in industry. With the two thresholds, users can compute the threshold of the decoding BER according to (15) and (18). After the watermarking extraction, users can compute the decoding BER by comparing extracted watermark sequence with embedded watermark sequence. If the decoding BER is lower than the threshold, the image is watermarked, whereas if it is higher than the threshold, the image is not watermarked.

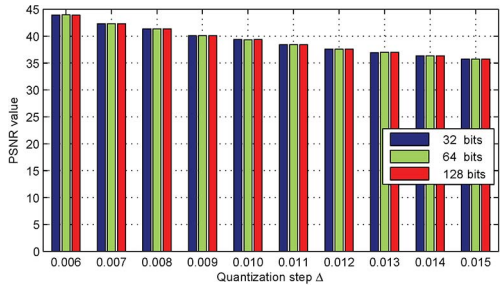


Fig. 6. Relationship between quantization step Δ and PSNR values under different watermark capacities (32, 64, and 128 b) for Lena image.

C. Tradeoff Among Capacity, Imperceptibility, and Robustness

In general, there exists a complex tradeoff among three performances in digital watermarking: capacity, imperceptibility, and robustness. We analyze the tradeoff among them as follows.

1) Tradeoff Between Imperceptibility and Robustness: In the proposed watermarking scheme, each pixel is quantized for embedding watermark with quantization step Δ. If Δ is too small, the robustness of the watermarking scheme decreases, but the imperceptibility increases; whereas if Δ is too large, the imperceptibility (measured by PSNR value) decreases, but the robustness increases. The relationship between Δ and the imperceptibility is also shown in Fig. 6.

2) Tradeoff Between Imperceptibility and Capacity: Because each pixel is quantized for embedding watermark with quantization step Δ, the embedding distortion is under control of Δ. If Δ is much smaller, the imperceptibility increases and vice versa. On the other hand, the capacity of the proposed watermarking scheme is equal to the number of leaf nodes N_{leaf} of the BSP tree for partitioning LDFT space. Therefore, the increase or decrease of capacity does not influence the imperceptibility, and the decrease of imperceptibility does not increase the capacity. As shown in Fig. 6, the increase of capacity does not decrease the PSNR value and vice versa when the quantization step Δ is constant.

3) Tradeoff Between Capacity and Robustness: The capacity of the proposed watermarking scheme is equal to the number of leaf nodes N_{leaf} of the BSP tree. If N_{leaf} is much larger, the number of pixels in a leaf node could be smaller, which results in less robustness and vice versa. In order to enhance the robustness of the watermarking scheme, we must decrease the capacity. Otherwise, if we do not care the robustness, we can increase the capacity significantly.

In case that attack on a watermarked image is very severe and the parameters are given as $P_s = 0.55$, $B = 0$, and $t_s = (512 \times 512)/(2N_w)$, the relation between watermark capacity N_w and the missing probability which is calculated according to (18) is shown in Fig. 7. As shown in Figs. 5 and 7, when the length of the watermark sequence is 100 b, the false-alarm probability is about 1.0×10^{-30}, and the missing probability is about 1.5×10^{-5}. Therefore, the effective embedding capacity of the proposed resynchronization scheme is about 100 b.

V. Experimental Results

To evaluate the capacity, imperceptibility, and robustness of the proposed watermarking scheme, we conduct experiments

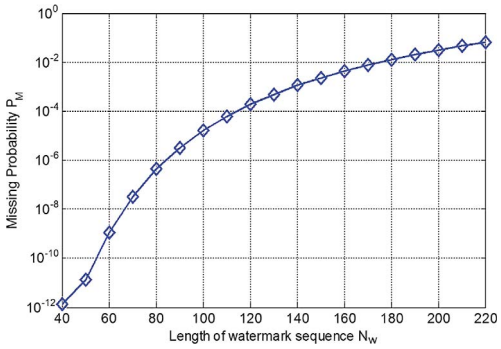


Fig. 7. Missing probability for the watermarked image with a size of 512 × 512, assuming $P_s = 0.55$, $B = 0$, and $t_s = (512 \times 512)/(2N_w)$.

on two test data sets. The first data set consists of three standard 8-b grayscale images (airplane, Lena, and peppers) of size 512 × 512. The second data set includes 100 various textures 8-b grayscale images of size 512 × 512. According to the analysis in Section IV, the parameters of LDFT are set as $T = 36$, $Q = 3$, and $H = 36$; the threshold T_D of the BSP tree is set as the mean value of $D_i(\cdot)$ of all pixels in the image. In the experiments, we use 32-, 64-, and 128-b pseudorandom noise sequences as information to embed, respectively.

A. Watermark Imperceptibility

The PSNR value between the original and watermarked images is a criterion for the watermark imperceptibility. Denote the original image and the watermarked image as $I = \{f(i,j)|i \in [1, C_1], j \in [1, C_2]\}$ and $I_W = \{f_w(i,j)|i \in [1, C_1], j \in [1, C_2]\}$. The PSNR value of I versus I_W is

$$\text{PSNR} = 10 \log_{10} \left(\frac{C_1 \cdot C_2 \cdot 255^2}{\sum_{i=1}^{C_1} \sum_{j=1}^{C_2} |f(i,j) - f_w(i,j)|} \right).$$

The embedding distortion can be controlled by using the quantization step size Δ of LQIM. In our experiments, the PSNR value is controlled to be over 40 dB. In the proposed watermarking scheme, the watermark energy is homogeneously embedded into the whole image, which is beneficial in generating the high-quality watermarked images and does not bring on a low imperceptibility in local feature regions. Fig. 7 demonstrates the performance of our watermarking scheme with 64-b watermark sequence. The feature-based method used in Fig. 8 is the method of [34]. In LQIM, smaller step sizes are devoted to smaller pixel values, and larger step sizes are associated with larger pixel values, so the imperceptibility of watermarking is very high, as shown in Fig. 8. Moreover, the watermarked images have no drawback of low imperceptibility in local regions with high PSNR values, and the watermarking scheme preserves the visual harmonization of the entire image.

B. Watermark Robustness

Experiments of common image-processing attacks, global attacks, local attacks, and noninvertible attacks have been performed to prove the effectiveness of the proposed watermarking scheme. The BER is used to evaluate the robustness of the watermarking scheme against various attacks.

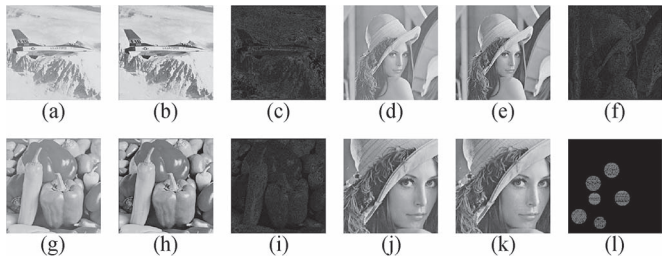


Fig. 8. Performance of our watermarking scheme. (a), (d), and (g) are the original images; (b), (e), and (h) are the watermarked images; and (c), (f), and (i) are the absolute differences between the original images and the watermarked images, multiplied by 10 for the purpose of better display. (j) is the zoomed original image, (k) is the zoomed watermarked image which is watermarked by the feature-based watermarking method in [34], and (l) is the absolute difference between the original image and the watermarked image which is watermarked by the feature-based method, multiplied by 10 for the purpose of better display.

TABLE I
ROBUSTNESS AGAINST VARIOUS COMMON IMAGE-PROCESSING ATTACKS AND COMBINED WITH SOME GLOBAL ATTACKS (%)

Attacks	Airplane		Lena		Peppers		100 images		
	32b	64b	32b	64b	32b	64b	32b	64b	128b
Watermarked image	0	0	0	0	0	0	0	0	0
Median filter 2×2	0	3.1	0	6.3	0	0	3.5	7.3	13.5
Median filter 3×3	6.3	17.3	6.3	12.5	0	15.6	9.4	14.8	19.2
Median filter 4×4	15.6	21.9	18.8	20.3	6.3	17.2	17.3	19.9	23.8
Mean filter 2×2	9.4	15.6	12.5	17.2	15.6	18.8	16.4	21.9	24.8
Gaussian filter 3×3	18.8	21.9	15.6	18.8	21.9	23.4	21.6	23.5	26.7
Uniform noise 0.01	0	0	0	0	0	0	0	0	0.1
Uniform noise 0.02	0	1.6	0	0	0	0	0	0.2	0.7
Uniform noise 0.03	43.8	48.4	15.6	23.4	6.3	12.5	14.2	19.8	21.7
Gaussian noise (σ=0.01)	0	0	0	0	0	0	0	0	0.8
Gaussian noise (σ=0.02)	10.6	13.3	18.6	35.8	15.9	22.6	15.9	23.9	29.2
JPEG 90	0	0	0	0	0	0	6.4	8.3	9.7
JPEG 80	0	1.6	3.1	4.7	3.1	6.3	10.4	14.1	16.6
JPEG 70	3.1	10.9	9.4	12.5	9.4	15.6	11.1	17.1	18.9
JPEG 60	9.4	12.5	15.6	17.2	15.6	18.5	14.8	19.7	21.9
JPEG 50	9.4	21.9	25	29.7	28.1	31.3	28.2	30.5	33.7
JPEG 40	18.8	29.7	34.4	35.9	34.4	37.5	30.1	37.4	37.9
JPEG2000 40%	0	0	0	0	0	0	0	0	1
JPEG2000 30%	0	0	0	0	0	0	0	0.2	9.1
JPEG2000 20%	0	3.1	0	0	0	4.7	0	1	11
Rotation 0.25° + JPEG90	3.1	15.6	0	1.5	6.2	15.6	7.1	10.9	17.1
Rotation 0.25° + JPEG70	9.4	23.4	9.8	13.2	18.7	20.3	11.4	18.7	27.3
Scaling 0.8 + JPEG90	7.5	14.6	11.2	13.5	9.8	19.1	11.4	13.7	23.6
Scaling 0.8 + JPEG70	11.8	34.3	15.2	25	10.7	21.7	12.8	26.6	39.1
Shearing (0%, 1%) + JPEG90	7.0	9.3	0	0	3.1	4.6	7.3	8.5	10.2
Shearing (0%, 1%) + JPEG70	18.7	21.8	9.7	14.6	12.5	17.6	12.5	18.2	25.8
LGT (1.007, 0.01, 0.01, 1.012) + JPEG90	6.2	7.8	9.3	9.3	15.6	17.1	9.4	10.9	24.2
LGT (1.007, 0.01, 0.01, 1.012) + JPEG70	21.8	23.4	31.2	23.4	31.2	34.3	28.1	26.5	32.8

1) Robust to Common Image-Processing Attacks: The performance of the proposed scheme under various common image-processing attacks is shown in Table I. These attacks include median filtering with sizes of 2 × 2, 3 × 3, and 4 × 4; mean filtering with a size of 2 × 2; Gaussian filtering with a size of 3 × 3; and adding uniform noise, Gaussian noise, JPEG, and JPEG2000 compression. The 3 × 3 Gaussian filter matrix is

$$[1 \;\; 2 \;\; 1; \;\; 2 \;\; 4 \;\; 2; \;\; 1 \;\; 2 \;\; 1].$$

The attacked image with uniform noise is

$$f'(x,y) = f(x,y) \cdot (1 + \beta \cdot n(x,y))$$

where $f(x,y)$ is the luminance pixel value of an input image at (x,y), β is a parameter that controls the strength of the additive noise, $n(x,y)$ is noise with uniform distribution and zero mean

TABLE II
ROBUSTNESS AGAINST VARIOUS GLOBAL ATTACKS OF THE PROPOSED WATERMARKING SCHEME (%)

Attacks	Airplane		Lena		Peppers		100 images		
	32b	64b	32b	64b	32b	64b	32b	64b	128b
Removed 5 rows and 17 columns	0	0	0	0	0	0	0	0	0.1
Removed 17 rows and 5 columns	0	0	0	0	0	0	0	0	0.2
Translation left 50 up 25	0	0	0	0	0	0	0	0	0
Rotation 0.25°	0	0	0	0	0	0	1.7	2.9	7.2
Rotation 0.5°	0	1.6	0	0	0	0	1.9	3.3	8.1
Rotation 1°	3.1	3.1	0	0	0	1.6	3.3	4.7	10.1
Rotation 30°	6.3	9.4	6.3	7.8	3.1	9.4	6.8	9	18
Rotation 45°	3.1	3.1	0	1.6	0	3.1	1.2	3.7	16.8
Scaling 0.7	21.9	29.7	25	28.1	21.9	26.6	21.5	25.3	27.3
Scaling 0.8	6.3	10.9	9.4	10.9	9.4	18.8	2.9	6.1	10.6
Scaling 0.9	0	1.6	0	1.6	0	3.1	2.8	4.9	10
Scaling 1.1	0	0	0	0	0	0	1.1	4.8	9.6
Scaling 1.3	0	0	0	0	0	0	1.5	5.2	10.1
Scaling 1.4	0	1.6	0	0	0	1.6	1.6	5.5	10.4
Shearing (0%, 1%)	0	0	0	0	0	0	0.2	0.6	2.7
Shearing (1%, 0%)	0	0	0	0	0	0	0.1	0.7	2.5
Shearing (5%, 0%)	0	1.6	0	0	0	0	1.1	1.7	6.8
Shearing (0%, 5%)	3.1	3.1	0	0	0	1.6	1.1	1.5	6.7
Shearing (1%, 1%)	6.3	7.8	0	1.6	3.1	4.7	2.8	4.9	10.3
Shearing (5%, 5%)	9.4	9.4	3.1	7.8	3.1	9.4	8.1	9.7	14
LGT (1.007, 0.01, 0.01, 1.012)	0	6.3	0	7.8	0	3.1	2.2	5.4	14.8
LGT (1.010, 0.013, 0.009, 1.011)	0	6.3	0	9.4	0	4.7	2.4	5.6	14.7
LGT (1.013, 0.008, 0.011, 1.008)	0	4.69	6.3	9.4	3.1	6.3	2.7	6.1	15.5

and unit variance, and $f'(x, y)$ is the luminance pixel value of the attacked image. The standard deviations of Gaussian noise are $\sigma = 0.01$ and $\sigma = 0.02$, respectively. As shown in Table I, the watermark is robust to various common image-processing attacks, including JPEG2000 compression up to a compression ratio of 20% and JPEG compression up to a quality factor of 50. The scheme performs well under Gaussian noise attack. The scheme performs very well under additive uniform noise attack with low strength, because it neither causes synchronization error nor causes LQIM extraction error for each pixel. However, it does not perform very well under additive uniform noise attack with high strength. Under the high-strength uniform noise addition attack, the shift value of the pixel is greater than $\Delta/2$ of LQIM, so the watermark bit cannot be extracted accurately (e.g., airplane in Table I).

2) Robust to Global Attacks: Global attacks are the most popular DAs. As shown in Table II, our scheme performs well under various global transform attacks such as RST, shearing, LGT etc. The BERs of rotation, translation, scaling, shearing, and LGT are all under 10%, except that the destructive zoom with scaling coefficient is lower than 0.7 when the length of the watermark sequence is 32 b. When the length is 64 b, the robustness against global attacks is also very high. The highest average BER value is 9.7% for the 100 various texture images, except the destructive zoom with a scaling coefficient of 0.7. As the analysis in Section IV-C, the increase of the watermark capacity will lead to the increase of BER. When the length is 128 b, the worst BER value is 16.8% for the 100 various texture images, except the destructive zoom with a scaling coefficient of 0.7. Many of the BERs under global attacks are 0, as shown in Table II. Obviously, the proposed scheme is successful against global transform attacks. After the watermarked image is attacked by global attacks, the image will usually be compressed by JPEG. Therefore, we do some experiments to shown the robustness of the proposed scheme against the combined global attacks and JPEG compression. As shown in Table I, the scheme is robust against the combined attacks when the quality factors of JPEG are high.

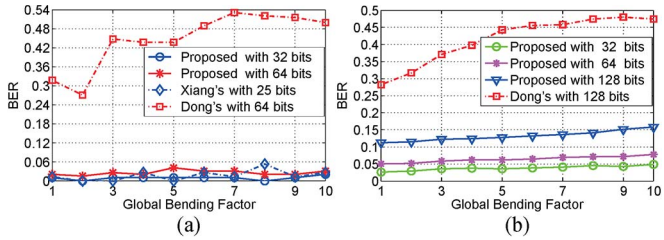


Fig. 9. Robustness to GBA attacks. (a) Average BER value of three standard images. (b) Average BER value of 100 various texture images.

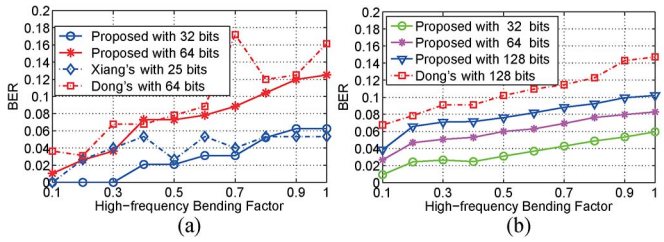


Fig. 10. Robustness to HFB attacks. (a) Average BER value of three standard images. (b) Average BER value of 100 various texture images.

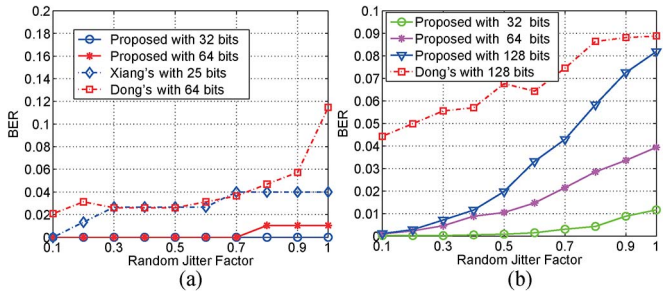


Fig. 11. Robustness to RJA attacks. (a) Average BER value of three standard images. (b) Average BER value of 100 various texture images.

3) Robust to Local Attacks: Local geometrical distortions are the destructible weakness for many watermarking schemes. In order to demonstrate robustness of the proposed watermarking scheme against local DAs, we present many simulation results and comparison results with the method of [16] and the method of [26]. As shown in Figs. 9–13, the proposed resynchronization is robust against many kinds of local DAs.

Figs. 9–11 demonstrate the robustness of the proposed watermarking scheme against RBAs including GBA, HFB, and RJA attacks. In Figs. 9(a), 10(a), and 11(a), the test data set consists of airplane, Lena, and peppers. The experimental data of the method of [16] are from literature [16], and the lengths of the watermark sequence are 25 b for the method of [16] and 32 b for the proposed method. The BER values are the averages of the BER values of airplane, Lena, and peppers. The simulation results outline that our scheme performs better than the method of [16] under HFB and RJA. Our scheme is comparable to the histogram-based method of [16] under GBA, although the data payload of [16] is lower than ours. In addition, our scheme performs better than the normalization-based method of [26] when the length of the watermark sequence is 64 b. Especially for GBA, the method of [26] falls flat. In Figs. 9(b), 10(b), and

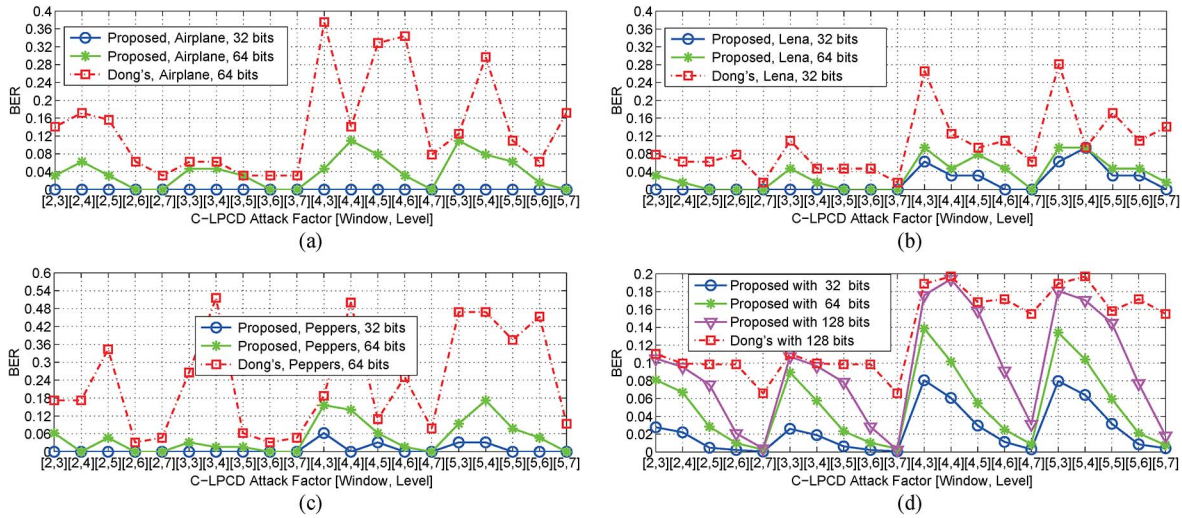


Fig. 12. Robustness to C-LPCD attacks. (a) Airplane. (b) Lena. (c) Peppers. (d) Average value of 100 various texture images.

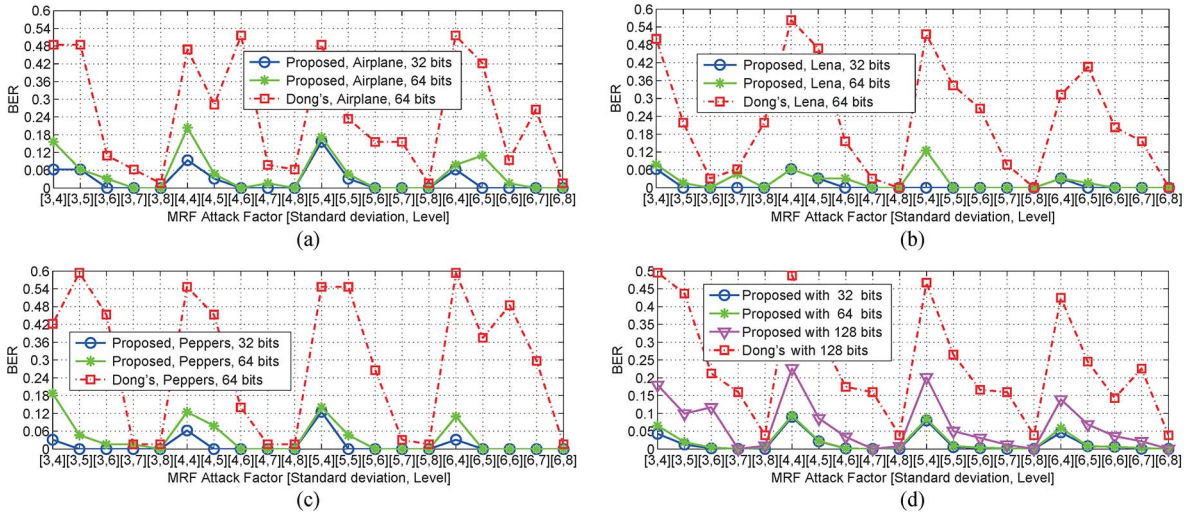


Fig. 13. Robustness to MRF attacks. (a) Airplane. (b) Lena. (c) Peppers. (d) Average value of 100 various texture images.

11(b), the test data set includes 100 various texture images. The BER values are the averages of the BER values of the 100 images. Because the effective embedding capacity of the method of [16] is from 20 to 30 b, we can only give comparison results with the method of [26] when the embedding capacity is 128 b. According to the comparison results shown in Fig. 9(b), 10(b), and 11(b), we can see that the proposed watermarking scheme is robust enough against RBAs.

Figs. 12 and 13 demonstrate the robustness of the proposed watermarking scheme against recently reported C-LPCD and MRF attacks. They show that our scheme performs very well under the two new local transform attacks. Most of the BER values are 0, and the maximal BER values of our scheme are also very small when the length of the watermark sequence is 32 b. Therefore, we need not compare it with the method of [16] when the length of the watermark sequence is 32 b. When the length is 64 b, our scheme outperforms the scheme of [26], which cannot resist C-LPCD attacks completely, as shown in Fig. 12(a)–(c). For MRF attacks, all BER values of our watermarking scheme are almost 0, and our scheme outperforms the scheme of [26], as shown in Fig. 13(a)–(c). When the length is 128 b, our scheme is also very robust against C-LPCD attack and MRF attack and outperforms the scheme of [26], as shown in Figs. 12(d) and 13(d).

Figs. 9–13 demonstrate that the proposed watermarking scheme can counteract local transform attacks effectually benefitting from the use of locally invariable LDFT and BSP tree partitioning methods.

4) Robust to Noninvertible Attacks: In our experiments, cropping and randomly removing rows and columns are noninvertible, so we named them noninvertible attacks. As shown in Table II, both BERs of removing 5 rows and 17 columns and removing 17 rows and 5 columns are 0 when the embedding capacity is 32 or 64 b. The worst BER is only 0.2% when the embedding capacity is 128 b. Under cropping, the original image cannot be recovered due to the permanent loss of partial content, so cropping is usually considered as a severe attack in the watermarking system. Praiseworthily, from Fig. 14, we can see that the proposed scheme can counteract 55% centered cropping, and it performs better than the histogram-based method [16] even if the comparison is beneficial to the

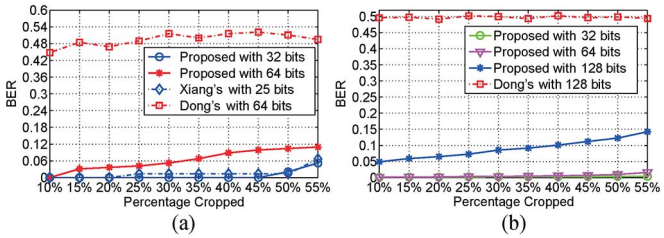


Fig. 14. Robustness to image cropping. (a) Average BER value of three standard images. (b) Average BER value of 100 various texture images.

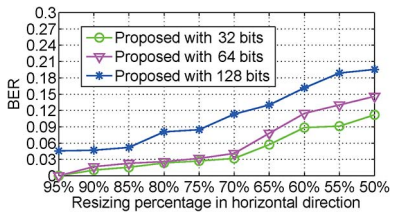


Fig. 15. Robustness to image retargeting. Average BER values of 20 images, each of which has an unambiguous salient object. All images have a resolution of 800×600.

method of [16], where the lengths of the watermark sequence are 25 b for the method of [16] and 32 b for the proposed scheme. Our scheme also performs very well when the length is 64 or 128 b. However, the scheme of [26] is helpless against cropping, as shown in Fig. 14.

The image retargeting is a new kind of noninvertible attack for image watermarking, which can preserve the regions of interest in images and discard other noninterest regions by seam carving schemes [45], [46]. The experimental results of robustness against image retargeting are shown in Fig. 15. We use a popular image retargeting method named seam carving in [46]. The forward energy is used for the seam carving. The Sobel operator used to calculate the gradient image is set as $[-1\ -2\ -1; 0\ 0\ 0; 1\ 2\ 1]$. Watermark sequences (32, 64, and 128 b) are embedded into luminance channels of 20 colored images by the proposed watermarking scheme, respectively, each of which has an unambiguous salient object. The resolutions of all images are 800×600. Owing to the visually important regions of the images that are effectively preserved, the proposed watermarking scheme is robust against seam carving when the width reduced by removing those unimportant seams is not very high, as shown in Fig. 15.

VI. Conclusion

In analyzing the drawback of the feature-based watermarking scheme, in this paper, we have proposed an effective resynchronization method against both global DAs and local DAs. The major contributions are the following: 1) it presents a new local and global RST-invariant transform referred to as LDFT, and 2) it introduces the BSP tree to partition the LDFT feature space. The location of each pixel is stable under global and local DAs and noninvertible cropping. The proposed watermarking scheme is robust against a wide variety of attacks, as indicated in the experimental results. Moreover, the scheme overcomes the drawbacks of the feature-based watermarking scheme. Our approach can be further improved by developing a more robust embedding method than LQIM.

Acknowledgment

The authors would like to thank Dr. S. Xiang for providing his source codes for performance comparisons and Prof. F. Y. Shih, Prof. W. Zeng, Prof. J.-S. Pan, and Dr. G. Cao for the very helpful discussions and suggestions while working on this paper.

References

[1] Y.-T. Wu and F. Y. Shih, "Genetic algorithm based methodology for breaking the steganalytic systems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 1, pp. 24–31, Feb. 2006.

[2] G. Langelaar, I. Setyawan, and R. Lagendijk, "Watermarking digital image and video data. A state-of-the-art overview," *IEEE Signal Process. Mag.*, vol. 17, no. 5, pp. 20–46, Sep. 2000.

[3] M. Barni, "What is the future for watermarking? (part I)," *IEEE Signal Process. Mag.*, vol. 20, no. 5, pp. 55–60, Sep. 2003.

[4] M. Barni and F. Bartolini, "Data hiding for fighting piracy," *IEEE Signal Process. Mag.*, vol. 21, no. 2, pp. 28–39, Mar. 2004.

[5] D. Zheng, Y. Liu, J. Zhao, and A. E. Saddik, "A survey of RST invariant image watermarking algorithms," *ACM Comput. Surveys*, vol. 39, no. 2, p. 5, 2007.

[6] L. Wang, H. Ling, F. Zou, and Z. Lu, "Real-time compressed-domain video watermarking resistance to geometric distortions," *IEEE MultiMedia*, vol. 19, no. 1, pp. 70–79, Jan. 2012.

[7] H. Tian, Y. Zhao, R. Ni, and J.-S. Pan, "Spread spectrum-based image watermarking resistant to rotation and scaling using radon transform," in *Proc. 6th Int. Conf. IIH-MSP*, Oct. 2010, pp. 442–445.

[8] H. Ling, L. Wang, and F. Zou, "Real-time video watermarking scheme resistant to geometric distortions," *J. Electron. Imaging*, vol. 20, no. 1, pp. 013025-1–013025-14, Mar. 2011.

[9] H. Ling, L. Wang, F. Zou, Z. Lu, and P. Li, "Robust video watermarking based on affine invariant regions in the compressed domain," *Signal Process.*, vol. 91, no. 8, pp. 1863–1875, Aug. 2011.

[10] Z.-C. Li, H. Wang, and S. S. Y. Liao, "Numerical algorithms for image geometric transformations and applications," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 1, pp. 132–149, Feb. 2004.

[11] F. A. P. Petitcolas and R. J. Anderson, "Evaluation of copyright marking systems," in *Proc. IEEE Multimedia Syst.*, Jun. 1999, vol. 1, pp. 574–579.

[12] V. Licks and R. Jordan, "Geometric attacks on image watermarking systems," *IEEE Multimedia*, vol. 12, no. 3, pp. 68–78, Jul./Sep. 2005.

[13] V. Licks, F. Ourique, R. Jordan, and F. Perez-Gonzalez, "The effect of the random jitter attack on the bit error rate performance of spatial domain image watermarking," in *Proc. Int. Conf. Image Process.*, Sep. 2003, vol. 3, pp. 455–458.

[14] M. Barni, A. D'Angelo, and N. Merhav, "Expanding the class of watermark de-synchronization attacks," in *Proc. 9th ACM Workshop MM-Sec*, Sep. 2007, vol. 1, pp. 195–204.

[15] A. D'Angelo, M. Barni, and N. Merhav, "Stochastic image warping for improved watermark desynchronization," *EURASIP J. Inf. Sec.*, vol. 2008, no. 1, p. 345 184, Mar. 2008.

[16] S. Xiang, H. J. Kim, and J. Huang, "Invariant image watermarking based on statistical features in the low-frequency domain," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 6, pp. 777–790, Jun. 2008.

[17] H. Tian, Y. Zhao, R. Ni, and G. Cao, "Geometrically robust image watermarking by sector-shaped partitioning of geometric-invariant regions," *Opt. Exp.*, vol. 17, no. 24, pp. 21 819–21 836, Nov. 2009.

[18] J. Dugelay, S. Roche, C. Rey, and G. Doërr, "Still-image watermarking robust to local geometric distortions," *IEEE Trans. Image Process.*, vol. 15, no. 9, pp. 2831–2842, Sep. 2006.

[19] M. Abrash, "BSP trees," *Dr. Dobbs Sourcebook*, vol. 20, no. 14, pp. 49–52, May/Jun. 1995.

[20] N. K. Kalantari and S. M. Ahadi, "A logarithmic quantization index modulation for perceptually better data hiding," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1504–1517, Jun. 2010.

[21] M. Barni, "Effectiveness of exhaustive search and template matching against watermark desynchronization," *IEEE Signal Process. Lett.*, vol. 12, no. 2, pp. 158–161, Feb. 2005.

[22] J. Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spectrum digital image watermarking," *Signal Process.*, vol. 66, no. 3, pp. 303–317, May 1998.
[23] M. Alghoniemy and A. Tewfik, "Image watermarking by moment invariants," in *Proc. Int. Conf. Image Process.*, Jan. 2000, vol. 2, pp. 73–76.
[24] Y. Xin, S. Liao, and M. Pawlak, "Circularly orthogonal moments for geometrically robust image watermarking," *Pattern Recognit.*, vol. 40, no. 12, pp. 3740–3752, Dec. 2007.
[25] S. Pereira and T. Pun, "An iterative template matching algorithm using the chirp-z transform for digital image watermarking," *Pattern Recognit.*, vol. 33, no. 1, pp. 173–175, Jan. 2000.
[26] P. Dong, J. G. Brankov, N. P. Galatsanos, Y. Yang, and F. Davoine, "Digital watermarking robust to geometric distortions," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2140–2150, Dec. 2005.
[27] P. Bas, J. Chassery, and B. Macq, "Geometrically invariant watermarking using feature points," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 1014–1028, Sep. 2002.
[28] C. Tang and H. Hang, "A feature-based robust digital image watermarking scheme," *IEEE Trans. Signal Process.*, vol. 51, no. 4, pp. 950–959, Apr. 2003.
[29] M. Kutter, S. K. Bhattacharjee, and T. Ebrahimi, "Towards second generation watermarking schemes," in *Proc. Int. Conf. Image Process.*, Oct. 1999, vol. 1, pp. 320–323.
[30] C. Schmid, R. Mohr, and C. Bauckhage, "Evaluation of interest point detectors," *Int. J. Comput. Vis.*, vol. 37, no. 2, pp. 151–172, Jun. 2000.
[31] H. Lee, I. Kang, H. Lee, and Y. Suh, "Evaluation of feature extraction techniques for robust watermarking," in *Proc. 4th Int. Workshop Digit. Watermarking*, Sep. 2005, vol. 1, pp. 418–431.
[32] S. S. Jin and D. Y. Chang, "Localized image watermarking based on feature points of scale-space representation," *Pattern Recognit.*, vol. 37, no. 7, pp. 1365–1375, Jul. 2004.
[33] S. S. Jin and D. Y. Chang, "Image watermarking based on invariant regions of scale-space representation," *IEEE Trans. Signal Process.*, vol. 54, no. 4, pp. 1537–1549, Apr. 2006.
[34] X. Wang, J. Wu, and P. Niu, "A new digital image watermarking algorithm resilient to desynchronization attacks," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 4, pp. 655–663, Dec. 2007.
[35] X. Gao, C. Deng, X. Li, and D. Tao, "Geometric distortion insensitive image watermarking in affine covariant regions," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 40, no. 3, pp. 151–172, May 2010.
[36] C.-C. Tsai, H.-Y. Lin, J. Taur, and C.-W. Tao, "Iris recognition using possibilistic fuzzy matching on local features," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 1, pp. 150–162, Feb. 2012.
[37] E. Tola, V. Lepetit, and P. Fua, "DAISY: An efficient dense descriptor applied to wide-baseline stereo," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 815–830, May 2010.
[38] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *Int. J. Comput. Vis.*, vol. 60, no. 1, pp. 63–86, Oct. 2004.
[39] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, vol. 1, pp. 886–893.
[40] A. Baraldi and P. Blonda, "A survey of fuzzy clustering algorithms for pattern recognition—Part I," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 29, no. 6, pp. 778–785, Dec. 1999.
[41] H. Xiong, J. Wu, and J. Chen, "k-means clustering versus validation measures: A data-distribution perspective," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 2, pp. 318–331, Apr. 2009.
[42] T. Tuytelaars and C. Schmid, "Vector quantizing feature space with a regular lattice," in *Proc. 11th IEEE Int. Conf. Comput. Vis.*, Oct. 2007, vol. 1, pp. 1–8.
[43] H. Tian, Y. Zhao, R. Ni, and J.-S. Pan, "Geometrically invariant image watermarking using scale-invariant feature transform and k-means clustering," in *Proc. Int. Conf. Comput. Collect. Intell., Technol. Appl.*, Nov. 2010, vol. 1, pp. 128–135.
[44] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Inf. Theory*, vol. 47, no. 4, pp. 1423–1443, May 2001.
[45] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," *ACM Trans. Graph.*, vol. 26, no. 3, p. 10, Jul. 2007.
[46] M. Rubinstein, A. Shamir, and S. Avidan, "Improved seam carving for video retargeting," *ACM Trans. Graph.*, vol. 27, no. 3, p. 16, Aug. 2008.

Huawei Tian received the B.E. degree in computer science and technology from the College of Computer Science, Chongqing University, Chongqing, China, in 2006. He is currently working toward the Ph.D. degree in the Institute of Information Science, Beijing Jiaotong University, Beijing, China.

His research interests include image processing, pattern recognition, digital watermarking and steganography, digital forensics, visual attention, and 3-D videos.

Yao Zhao (M'06–SM'12) received the B.S. degree from the Radio Engineering Department, Fuzhou University, Fuzhou, China, in 1989, the M.E. degree from the Radio Engineering Department, Southeast University, Nanjing, China, in 1992, and the Ph.D. degree from the Institute of Information Science, Beijing Jiaotong University (BJTU), Beijing, China, in 1996.

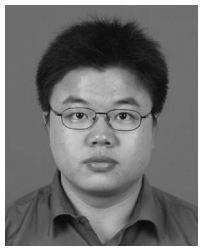
He became an Associate Professor at BJTU in 1998 and became a Professor in 2001. From 2001 to 2002, he was a Senior Research Fellow with the Information and Communication Theory Group, Faculty of Information Technology and Systems, Delft University of Technology, Delft, The Netherlands. He is currently the Director of the Institute of Information Science, BJTU. He is currently leading several national research projects from the 973 Program, the 863 Program, and the National Science Foundation of China. His research interests include image/video coding, fractals, digital watermarking, and content-based image retrieval.

Dr. Zhao was a recipient of the award from the National Science Foundation of China for Distinguished Young Scholars in 2010.

Rongrong Ni (M'10) received the Ph.D. degree in signal and information processing from the Institute of Information Science, Beijing Jiaotong University (BJTU), Beijing, China, in 2005.

Since Spring 2005, she has been with the faculty of the School of Computer and Information Technology and the Institute of Information Science, BJTU, where she has been an Associate Professor since 2008. She is currently the Principal Investigator of three projects funded by the Natural Science Foundation of China. In addition, she participates in 973, 863, and international projects as the backbone. She has published more than 60 papers on academic journals and conferences. She is the holder of six national patents. Her research interests include image processing, data hiding and digital forensics, pattern recognition, computer vision, etc.

Dr. Ni is a member of IET and EURASIP. She was selected in "Beijing Science and Technology Stars Projects" in 2008 and was awarded "Jeme Tien Yow Special Prize in Science and Technology" in 2009.



Lunming Qin received the B.E. degree in automation and the M.E. degree in signal and information processing from Beijing Jiaotong University, Beijing, China, in 2003 and 2006, respectively, where he is currently working toward the Ph.D. degree in signal and information processing in the Institute of Information Science.

His research interests include image segmentation, medical image analysis, deformable models, computer vision, and image watermarking.

Xuelong Li (M'02–SM'07–F'12) is currently a Full Professor with the Center for Optical Imagery Analysis and Learning, State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, Shaanxi, China.