A Desynchronization Resilient Watermarking Scheme

Xiaojun Qi and Ji Qi

Department of Computer Science, Utah State University, Logan, UT 84322-4205
Xiaojun.Qi@usu.edu, jiqi79@gmail.com

Abstract. This paper presents a content-based desynchronization resistant digital image watermarking scheme. The image content is represented by strong important feature points obtained by our robust and improved Harris corner detector. These feature points are more resistant to geometric attacks and are therefore used by the Delaunay triangle matching and image restoration method to reduce synchronization errors. The spread-spectrum-based blind watermark embedding and retrieval scheme is applied in the Fourier domain of each perceptually highly textured subimage. The multiplicative scheme is then used to embed the same copy of the watermark at highly secure mid-frequency positions generated by one-way hash functions with different secret keys. The watermark detection decision is based on the number of matched bits between the recovered and embedded watermarks in embedding subimages. Experimental results demonstrate that our scheme is more robust to desynchronization attacks (i.e., geometric and common image processing attacks) than other peer feature-based approaches.

Keywords: Desynchronization resilient digital watermarking, important feature points, spread-spectrum-based blind watermarking, Delaunay triangle matching.

1 Introduction

With the rapid growth of multimedia data distribution, many digital watermarking schemes have been proposed for copyright protection. The robustness of the watermark to common image processing and geometric attacks is essential in many applications [1]. However, these desynchronization attacks are difficult to tackle due to induced synchronization errors in watermark detection and decoding. Several watermarking schemes, including template-based, invariant-domain-based, moment-based, and feature-based, have been developed to counterattack desynchronization distortions.

Template-based watermarking algorithms [2], [3], [4] embed image independent templates to assist synchronization in watermark detection process. However, these template features can be exploited [5] to destroy the synchronization pattern. Invariant-domain-based watermarking algorithms [6], [7], [8], [9] generally provide a RST (Rotation, Scaling, and Translation) invariant domain to maintain synchronization under affine transforms. The Fourier-Mellin transform and log-polar resampling are two typical examples. However, the interpolation accuracy and aliasing in the resampling and integration may cause problems in synchronization. Moment-based watermarking

Y.Q. Shi (Ed.): Transactions on DHMS IV, LNCS 5510, pp. 29–48, 2009.

algorithms utilize ordinary geometric moments [10], [11], [12] or Zernike moments [13], [14] to solve the geometric invariance problem. However, perfect invariance cannot be achieved due to the discretization errors. Feature-based watermarking algorithms [15], [16], [17], [18], [19], [20] use image dependent feature points to represent invariant reference points for both embedding and detection. They generally are the best approaches to resisting desynchronization distortions since feature points provide stable references for both watermark embedding and detection. Several related, representative schemes are briefly reviewed here.

Bas *et al.* [15] used the Harris corner detector for feature extraction. These feature points were mixed with a Delaunay tessellation to mark each triangle for embedding the watermark. The original watermark triangles were then warped during the detection to correlate with the corresponding marked triangles. Similarly, Seo and Yoo [16], [17] extracted feature points using the Harris-Laplace detector and decomposed the image into disjointed local circular or elliptical regions for watermark embedding and extraction. Lee *et al.* [18] proposed to use the SIFT (Scale-Invariant Feature Transform) to determine the local circular regions for watermarking. The simulation results from all of the above methods show that the robustness of each scheme depends on the capacity for preserving feature points after geometric transformation, especially on images with more texture and images with less texture and large homogeneous areas. Moreover, these methods embed the watermark in the spatial domain after geometric normalization according to the shapes of the region. Consequently, watermark robustness to common image processing is not satisfactory and the feature points-based transformation domain watermarking schemes were proposed. Tang and Hang [19] adopted the Mexican hat wavelet scale interaction method to extract feature points. They embedded and extracted the watermark in the normalized disks centered at the extracted feature points. Wang *et al.* [20] improved Tang's method by using the scale invariant Harris-Laplace detector to find the radius of each circular region. However, Tang's scheme performs well under only mild geometric distortion and certain common image processing attacks. The watermark capacity of both schemes is only 16 bits, which may restrict their practical applications.

In this paper, we develop a desynchronization resilient watermarking scheme. This scheme combines the advantages of important feature extraction, perceptual analysis, one-way hash functions, and spread-spectrum-based blind watermark embedding and retrieval to reduce the watermark synchronization problem and resist different attacks. Section 2 describes the proposed robust feature extraction method. Section 3 briefly presents our variants of two important techniques used in the proposed scheme. Section 4 covers the details of the watermark embedding and detection procedure. Section 5 compares our scheme with three feature-based approaches in terms of robustness against both geometric distortions and common image processing attacks. In addition, we also demonstrate the performance of our proposed scheme on 105 images of various textures under different Stirmark attacks. Section 6 concludes this presentation.

2 Feature Points Extraction

Extracting feature points is the most important step in the proposed digital image watermarking scheme. In order to detect watermarks without access to the original images, we look for feature points that are perceptually significant and can thus resist

various types of desynchronization distortions. These image-content-bounded feature points can be further used as synchronization markers (i.e., anchor points) in watermark detection. To this end, we propose a robust and improved Harris corner detector to find relatively strong IFPs (Important Feature Points) to reduce the synchronization errors in watermark detection. Our two major contributions are:

- Improve the Harris corner detector [21] to reduce noise effect and regulate the density of IFPs based on the dimension and texture of the image.
- Strengthen the improved Harris corner detector to discard some non-stable IFPs.

These two improvements can also be applied to other detectors to achieve better performance.

The algorithmic flow of our improved Harris corner detector is summarized as follows:

1. Apply a rotationally symmetric 3×3 Gaussian low-pass filter with the standard deviation of 0.5 to blur the original image to increase the noise resistance.
2. Compute three derivative images, A, B, and C, by convolving the blurred image with the horizontal, vertical, and diagonal edge filters, respectively.
3. Apply the same Gaussian low-pass filter to blur three derivative images (e.g., A, B, and C) to further increase the noise tolerance.
4. Calculate the corner response function R as defined in [21], i.e., $R = (AB\text{-}C)^2\text{-}0.4(A+B)^2$, within a circular window, which is at the image center and covers the largest area of the original image. The resulting function reduces the effect of image center based rotation attacks and removes the corner points near the image border.
5. Search for IFPs whose corner response value $R(x,y)$'s are larger than a threshold T and are the local maxima within a circular neighborhood centered at (x, y).

We choose the circular neighborhood window to avoid the increasing detector anisotropy and to obtain a homogeneous distribution of feature points in the image. It is also important to determine the appropriate window size since a small window makes the feature points concentrate on textured areas and a large window tends to isolate the feature points. Fig. 1 illustrates the effect of different window sizes on the resultant feature points. We can easily observe that the smaller window yields more feature points. However, these "extra" feature points will require more computational cost in the detection process. The larger window yields fewer feature points and requires fewer computational cost in the detection process. However, the possibility of losing some IFPs is also increased. In order to make a compromise between the number of feature points and the computational cost in watermark detection, we determine a suitable window size based on the dimension and texture of the image. We roughly classify image textures as high, medium, and low, based on the ratio of the feature points to the total number of pixels in an image, wherein the feature points are obtained by using our improved Harris corner detector with a fixed 3×3 neighborhood window. That is:

$$\text{image has} \begin{cases} \text{high texture} & \text{if ratio} \geq 0.01 \\ \text{medium texture} & \text{if ratio} \geq 0.002 \\ \text{low texture} & \text{if ratio} \geq 0.0001 \end{cases} \tag{1}$$

The diameter of the circular window is calculated:

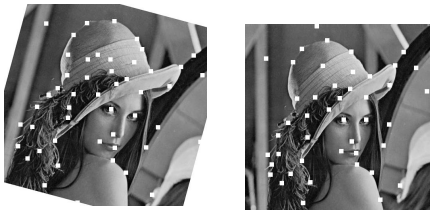
$$D = \frac{\sqrt{wh}}{np} \tag{2}$$

where integers w and h respectively represent the width and height of the image; integer p is an empirical value (i.e., $p = 5$) for obtaining a reasonable number of feature points for images with large homogeneous areas; and integer n is the window size quantizer, which depends on the texture of the image. It is set to be 2, 2.5, and 3 for images with high, medium, and low textures, respectively.

(a) (b)

Fig. 1. The effect of different window sizes on the feature points. (a) 26×26. (b) 52×52.

Our improved Harris corner detector ensures to extract IFPs in a noisy image. However, these IFPs may not be robust under certain geometric or common image processing attacks. Fig. 2 demonstrates this observation by displaying the IFPs of two attacked Lena images. We can clearly see that some IFPs are present in both attacked images while others are only present in one attacked image.



(a) (b)

Fig. 2. The non-robustness of IFPs under attacks. (a) IFPs after rotation. (b) IFPs after resizing.

In order to keep the strong IFPs which resist geometric attacks, we further enhance the improved Harris detector by quantitatively evaluating its capacity for preserving the IFPs when the image undergoes certain attacks. This evaluation function is computed as:

$$Score = \frac{Num_{kept} - (Num_{new} + Num_{loss})}{Num_{kept} + Num_{loss}} = \frac{Num_{kept} - (Num_{new} + Num_{loss})}{Num_{ori}} \tag{3}$$

where Num_{ori} denotes the number of IFPs present in the original image; Num_{new}, Num_{loss}, and Num_{kept} represent the number of IFPs that have been created, destroyed, and preserved after certain attacks, respectively. The preservation, creation, or loss status of these feature points is evaluated by performing an inverse transformation, which restores the attacked image to be aligned with the original image, on the IFPs obtained from the attacked image. This evaluation function yields the maximal value of 1 when the detector precisely finds the IFPs under a certain attack, i.e., $Num_{new} = Num_{loss} = 0$. In general, the larger the *Score*, the more capacity of the detector to preserve the IFPs under attacks.

Fig. 3 illustrates the block diagram of finding the relatively strong IFPs. Various image rotation and scaling attacks have been performed on the original image to find several attacks that preserve most of the strong IFPs in the original image. Our extensive experimental results show that random attacks can roughly estimate all possible rotations and scaling factors. The best attacks that achieve the highest *Score* values will be further used to individually pre-attack the original image to find the IFPs in its corresponding attacked images. Choosing these pre-attacks ensures that a decent number of strong IFPs can be obtained to reduce the synchronization errors. The strong IFPs are obtained by:

$$IFP_{strong} = IFP_{ori} \bigcap_{i=1}^{n} \left\{ Align\left(IFP_{Attack_i} \right) \right\} \tag{4}$$

where IFP_{ori} represents the IFPs in the original image; IFP_{Attack_i} represents the IFPs after the *i*th best attack, which can be either a rotation or a scaling attack; and $Align(\bullet)$ is an alignment operator to register the original and attacked images; and *n* is the number of the best attacks and is adaptively chosen based on the image texture. In our system, we set *n* to be 6 for high textured images, 4 for medium textured images, and 2 for low textured images.

Fig. 4 illustrates the effect of different pre-attacks on the resulting IFPs and shows the relatively strong IFPs obtained by intersecting IFPs of the original and its pre-attacked images. We can easily observe the sensitivity of the IFPs to different attacks. That is, some IFPs may disappear or show up or shift a bit after attacks. Consequently, it is impossible to locate the robust IFPs which can survive various attacks in the real world. To address this challenging issue, we select a few best pre-attacks to extract the strong IFPs that are likely preserved under different attacks. Such a choice keeps the balance between the robustness and the number of strong IFPs. Specifically, the intersection operation increases the robustness of the extracted strong IFPs and keeps enough IFPs to regain the synchronization (i.e., align the watermarked image with the original image). In other words, a robust resynchronization can be achieved by using a few best pre-attacks for a compromise between the robustness and the number of strong IFPs. The rationale for choosing the best pre-attacks is based on the following observations:

1. Most IFPs surviving a rotation or scaling attack can survive any combined RST attacks or any image processing attacks if the image is not cropped. The following intuitive proofs support this observation: a) Cropping may result in substantial loss of the IFPs and therefore will not be considered as a pre-attack. b) A feature point,

major characteristic in an image, must be differentiated from its neighbors no matter what kinds of the attacks are performed on the image. c) Any rotation or scaling attack, like any combined RST attack, requires an interpolation operation to modify the intensity of each pixel. The image processing attacks will modify the intensity of each pixel in another way. However, these modifications should not make the differentiation between the feature point and its neighbors disappear. That is, most feature points should still stand out when compared to their neighbors after geometric and image processing attacks. As a result, we can always use a rotation or scaling attack to find potential IFPs.

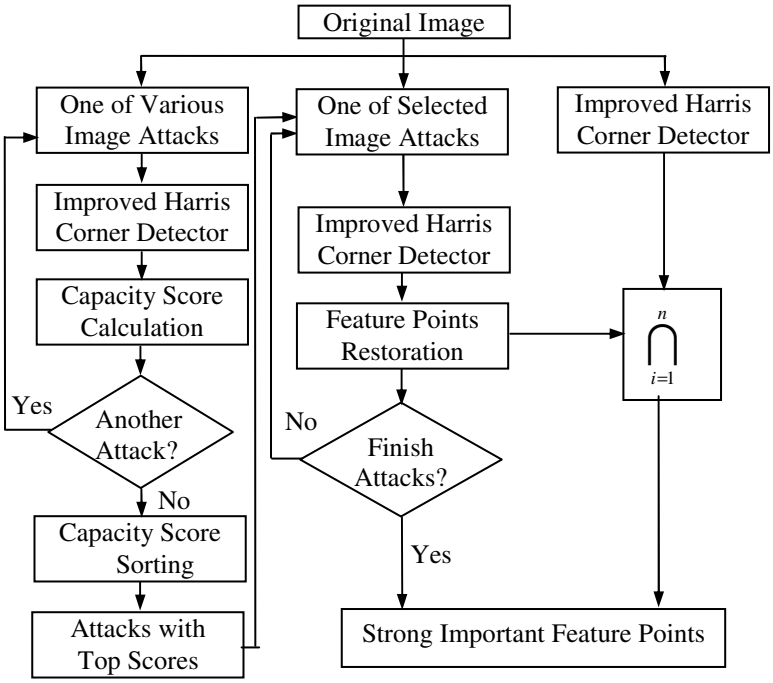


Fig. 3. Block diagram of extracting strong IFPs

2. The IFPs which are preserved after several pre-attacks are more robust against attacks than the other IFPs which are destroyed or created after the pre-attacks.
3. The best pre-attacks usually yield more IFPs than any other pre-attacks (e.g., random attacks, worst attacks, or image processing attacks). This leads to more anchor points for synchronization and therefore increases the possibilities for finding the geometric transformation in watermark detection.
4. The IFPs obtained from the best pre-attacks always intersect with majority IFPs obtained from an attacked image. This indicates that we can always find enough IFPs to perform synchronization.

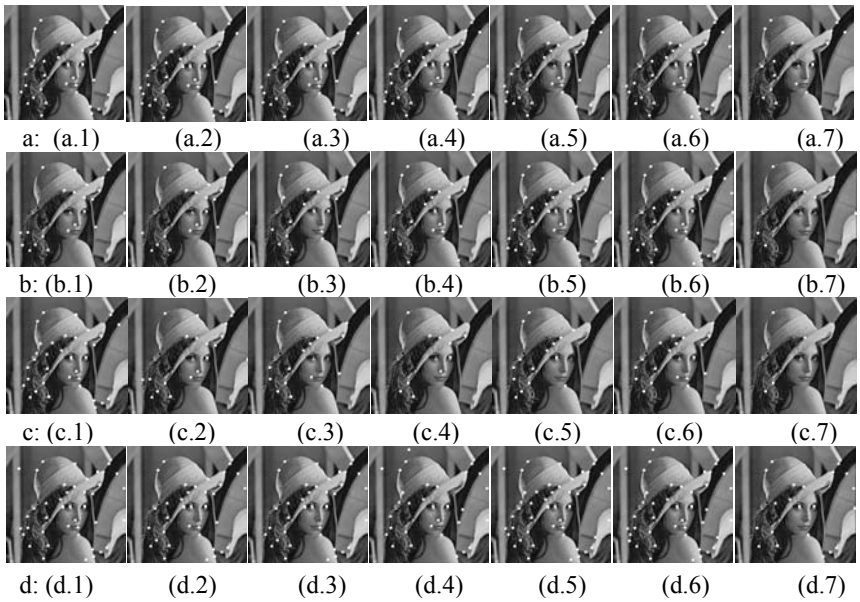


Fig. 4. The effect of different pre-attacks on the resultant strong IFPs. (a) IFPs obtained from 6 best pre-attacks. (a.1) R180º (a.2) R250º (a.3) R265º (a.4) R280º (a.5) R60º (a.6) S0.95 (a.7) Final strong IFPs by intersecting the original and its 6 best pre-attacked images. (b) IFPs obtained from 6 worst pre-attacks. (b.1) R145º (b.2) R235º (b.3) R220º (b.4) R320º (b.5) R190º (b.6) S0.90 (b.7) Final strong IFPs by intersecting the original and its 6 worst pre-attacked images. (c) IFPs obtained from 6 random pre-attacks. (c.1) R16º (c.2) R145º + S0.95 (c.3) R240º + S0.87 (c.4) R110º + S0.96 + T[5, 5] (c.5) R68º + S0.85 + T[10, 10] (c.6) S0.8 +T[10, 10] (c.7) Final strong IFPs by intersecting the original and its 6 random pre-attacked images. (d) IFPs obtained from 6 common image processing pre-attacks. (d.1) Median filtering 2×2 (d.2) Median filtering 3×3 (d.3) Mean filtering 3×3 + JPEG90% (d.4) Sharpening (d.5) Gaussian filtering (d.6) Histogram equalization (d.7) Final strong IFPs after intersecting the original and its 6 common image processing pre-attacked images. Here, R denotes rotation, S denotes scaling, and T denotes translation.

5. The IFPs obtained from the worst pre-attacks seem to be the most stable ones. However, some IFP may not be preserved under certain attacks due to the sensitivity of the IFPs to different attacks. In addition, these IFPs may not possess sufficient anchor points for the synchronization in the detection process.

Fig. 5 demonstrates the final preserved strong IFPs by applying our proposed robust and improved Harris corner detector on four images with different textures. That is, random rotation and scaling operations are respectively applied on each image to find its best pre-attacks. The improved Harris detector is then applied to find the IFPs for each original image and its corresponding best pre-attacked images. The intersection of all IFPs yields strong IFPs to be saved for the detection process. As shown in Fig. 5, the proposed approach effectively eliminates some unreliable feature points which fail to be detected after certain geometric attacks.

Fig. 5. Relatively strong and important feature points

3 Our Variants of Related Techniques

3.1 The PN-Sequence and One-Way Hash Functions

We use the PN (Pseudo-Noise) sequence based spread spectrum method in our system due to its robustness against noise and its capability to achieve error free transmission near or at the limits set by Shannon's noisy channel coding theorem [22]. This sequence combined with the watermark is adaptively embedded into mid-frequency positions in the DFT domain since high-frequency watermark can be easily eliminated by lossy compression and low-frequency watermark is usually noticeable.

We improve the one-way hash function [23], which is easy to compute and difficult to invert, to generate the highly secure mid-frequency positions by the following seven steps:

1. Save all middle frequency positions into vector V.
2. Randomly choose two large prime numbers p and q, and compute the secret key $n=pq$.
3. Obtain two seeds X and Y using the encipher process :

$$X = m^K \bmod n;\ Y = X^2 \bmod n; \tag{5}$$

   where m is the original image identification number (i.e., a numerical serial number for registering the image) and K is the second secret key.
4. Calculate an index l by:

$$Y = Y^2 \bmod n\ ; l = (Y \bmod n) \bmod length(V); \tag{6}$$

5. Choose the lth item in V as the embedding position.
6. Remove the lth item from V so no duplicated positions are produced and no collision occurs.
7. Repeat steps 4-6 until the total number of embedding positions is reached.

These highly secure embedding positions can be easily reproduced given the same secret keys n and K. In the meantime, the reproduction of these positions is computationally infeasible without knowing n and K. To ensure the attackers cannot find out the watermark embedding positions by comparing several watermarked copies, different secret keys are used to generate embedding positions for each embedding subimage.

3.2 Blind Embedding and Retrieval

A watermarking system should be secure and reliable. It is also desirable to extract the watermark independent of the original image. In our system, we adopt the blind retrieval scheme of MPEG video watermark [24], [25] to eliminate the need of storing values at embedding positions of each host subimage. We employ this blind retrieval scheme in the DFT domain instead of their proposed DCT domain. The multiplicative instead of additive embedding is also applied in our modified scheme. That is:

$$\hat{FI}_i = FI_i + FI_i \times G \times W_i \times p_i, \quad i = 1,...,N \tag{7}$$

where FI_i is the original mid-frequency DFT sequence with length N, $\hat{FI}_i$ is the watermarked DFT sequence, G is the embedding strength, p_i is the bipolar PN sequence generated by a secret key, and W_i is the blind watermark bit sequence obtained by repeating the bipolar watermark message w_i by a spreading factor s such that $W_i = w_j$ for $js \le i < (j+1)s$.

The blind retrieval can be achieved by de-spreading the blind watermarked bit sequence using the correlation detector. The same embedding PN-sequence p_i is used to multiply the possibly watermarked sequence:

$$W_i' = p_i \hat{FI}_i \tag{8}$$

The W_i' is further grouped into blocks of size s where each block is computed:

$$\sum_{i=js+1}^{(j+1)s} p_i \hat{FI}_i = \underbrace{\sum_{i=js+1}^{(j+1)s} p_i FI_i}_{S1} + \underbrace{G \sum_{i=js+1}^{(j+1)s} p_i^2 FI_i W_i}_{S2} \tag{9}$$

where $j = 0,...,n-1$ and n is the length of w_i. For a large s, FI_i is statistically uncorrelated with p_i. As a result, $S_2 >> S_1$ and Eq. (9) is simplified:

$$\sum_{i=js+1}^{(j+1)s} p_i \hat{FI}_i \approx \underbrace{G \sum_{i=js+1}^{(j+1)s} p_i^2 FI_i W_i}_{S2} = \underbrace{G \sum_{i=js+1}^{(j+1)s} FI_i W_i}_{S2} \tag{10}$$

Since FI_i and G are positive, the embedded bipolar watermark bit w_j is therefore similar to the sign of the correlation sum $\hat{w}_j$. That is:

$$w_j \approx \hat{w}_j = sign\left(\sum_{i=js+1}^{(j+1)s} p_i \hat{FI}_i \right), \quad j = 0,...,n\text{-}1 \tag{11}$$

4 Watermark Embedding and Detection Scheme

4.1 Watermark Embedding Scheme

Our watermark is designed for copyright protection. We view all possible embedding subimages as independent communication channels. To improve the robustness of the

transmitted watermark bits, all channels carry the same copy of the chosen watermark. During the detection process, we claim the existence of watermark if one copy of the embedded watermark is correctly detected in one embedding subimage. The watermark embedding process is detailed step by step as follows:

1. **Image tessellation:** Evenly divide the 8-bit grayscale image into 3×3 nonoverlapping subimages. The last several nondivisible rows and columns are not used for embedding.
2. **Perceptual analysis:** Apply the Harris corner detector to find all feature points in the original image by using a 3×3 window. Choose the subimages that have a large number of feature points to be embedding blocks. These blocks are perceptually high textured.
3. For each perceptually high textured subimage *SubA*:
   3.1 **DFT:** Apply global DFT to obtain *FSubA*.
   3.2 **Position Generator:** Generate highly secure embedding positions in the mid-frequencies between f_1 and f_2 in the upper half plane of *FSubA* by using our one-way hash function.
   3.3 $\oplus$ **operation:** Embed the spread bipolar watermark message bit W_j into each position (x_k, y_k) by using the multiplicative formula [26]:

$$FSubA(x_k, y_k)' = FSubA(x_k, y_k) + FSubA(x_k, y_k) \times G \times W_j \times p_j \tag{12}$$

   where $p_j \in \{1, -1\}$ (zero mean and unit variance) is the PN-sequence generated by a secret key. The same changes are carried out at center-based symmetric positions due to the constraints in the DFT domain for obtaining a real image.
   3.4 **IDFT (Inverse DFT):** Apply the IDFT to *FSubA'* to obtain the watermark embedded subimage *SubA'*, which replaces the original subimage *SubA*.

The proposed robust and improved Harris detector is finally used to find strong IFPs in the watermarked image. The position of each strong IFP, the bipolar watermark message bit sequence W_i, the number of embedding positions *Len*, two secret keys *n* and *K* for our one-way hash function in each subimage, two middle frequency ratios, and the secret key for generating the PN-sequence are saved for watermark detection. Since strong IFPs are obtained via the intersection operation, the number of IFPs is optimized and the storage is minimal compared to the cost of saving the image itself. If all the information is compressed, the storage cost will be further minimized.

4.2 Watermark Detection Scheme

The watermark detection procedure does not need the original image. The relatively strong IFPs are first extracted by intersecting the IFPs obtained by applying our proposed improved Harris corner detector on the probe image and a few randomly rotated probe images. Two sets of Delaunay tessellation-based triangles [27] are generated using the strong IFPs found in the probe image and the saved strong IFPs, respectively. These two sets of triangles are then matched to determine the possible geometric transformations the probe image has undergone. These geometric transformations are further utilized to restore the probe image so synchronization errors are minimized in the detection.

The choice of Delaunay tessellation is based on two attractive properties: 1) Local property: If a vertex disappears, the tessellation is only modified on connected triangles. 2) Stability area: Each vertex is associated with a stability area where the tessellation pattern is not changed when the vertex is moved within this area. That is, the tessellation patterns of other triangles remain the same even though losing or shifting an IFP affects the triangle(s) connected to it. In addition, two properties of the Delaunay tessellation always ensure that an identical generation of triangles can be obtained if the relative positions of the IFPs do not change. We implemented the Qhull algorithm [28] to generate the IFPs-based triangles due to its fast speed and less memory constraints.

In our system, the angle radians are used to match Delaunay tessellation-based triangles. That is, if two triangles have very similar angle radians (i.e., the angle difference is less than 0.01 radian), they are claimed to be likely matched. The possible geometric transformations are determined from the matched triangle pairs since the IFPs-based triangles undergo the same transformation as the image itself. The detailed steps are:

1. Calculate the scaling factor *SF* by resizing the probe triangle to the same size as the target matched triangle.
2. Calculate the translation factor *TF* by registering one of the vertices of the matched triangle pair.
3. Calculate the rotation factor *RF* by aligning the other two unregistered vertices of the matched triangle pair.

These factors form a 3-element tuple (*SF*, *TF*, *RF*) where *SF* measures the scaling ratio up to a precision of 1/10, *TF* measures the translation in pixel numbers, and *RF* measures the rotation angle in an integer degree.

Since an image and the within triangles undergo exactly the same transformation, we use the majority of the identical 3-element tuples obtained from all matched triangle pairs to restore the probe image. The same embedding procedure is applied to the restored probe image to obtain the watermark embedded DFT sequence $FSubA_i$'' for each potential embedding subimage *i*. The blind watermark retrieval scheme is then applied to extract the bipolar watermark bit sequence W_j', which is compared with the original watermark bit sequence W_j to determine the presence of the watermark. That is, the number of matched bits in a potential embedding subimage is compared with a threshold to determine whether the watermark is present in the probe image. This threshold is calculated based on the false-alarm probability that may occur in watermark detection. The Bernoulli trails are used to model W_j' since every watermark bit is an independent random variable. The probability of a *k*-bit match between extracted and original watermark bit sequences with a length of *n* is calculated as:

$$p_k = \binom{n}{k} \cdot p^k (1-p)^{n-k} \tag{13}$$

where *p* is the success probability for the extracted bit to be matched with the original watermark bit. We further simplify Eq. (13) by assuming *p* to be 1/2:

$$p_k = \left(\frac{1}{2}\right)^n \cdot \left(\frac{n!}{k!(n-k)!}\right) \tag{14}$$

The false-alarm probability for each embedding subimage is a cumulative probability of the cases that $k_i \geq T_i$, where k_i and T_i respectively represent the number of matching bits and the threshold for each subimage i. It is computed as:

$$P_{false-alarm}(i) = \sum_{k_i=T_i}^{n} \left(\frac{1}{2}\right)^n \cdot \left(\frac{n!}{k_i!(n-k_i)!}\right) \tag{15}$$

Based on Eq. (15), the perfect match between the extracted and original watermarks in a single embedding subimage leads to a false alarm probability of 6.10×10^{-5}. This is a low false alarm probability so we can confidently claim the watermark exists. In our system, we will check the perfect match in any embedding subimage to indicate the presence of the watermark.

5 Experimental Results

To evaluate the performance of the proposed watermarking scheme, we conducted experiments on various standard 8-bit grayscale images and different kinds of attempting attacks. We first perform the watermark invisibility test using four 512×512 8-bit gray-level images. We then illustrate the effectiveness of the proposed strong IFPs-based image restoration scheme, which functions as a self-synchronization scheme to align the possibly geometrically distorted watermarked image with the original one. Next, we perform extensive comparisons with three well designed feature-based RST resilient watermarking schemes proposed by Tang and Hang [19], Wang *et al.* [20], and Bas *et al.* [15]. Finally, we summarize the performance of our proposed scheme under a variety of Stirmark attacks on 105 8-bit watermarked grayscale images.

5.1 Watermark Invisibility Test

We evaluate watermark invisibility on four images: Lena, Pepper, Airplane, and Baboon. These four images correspond to several texture categories. For example, Baboon includes textured areas with high frequency components; Lena and Airplane include large homogeneous areas whereas Lena has sharp edges; and Pepper falls in a low-textured category. The PSNRs of these four watermarked images are 43.33, 44.06, 42.27, and 37.62, respectively. These PSNR values are all greater than 35.00db, which is the empirical value for the image without any perceivable degradation [29].

5.2 Important Restoration Test

Image restoration is an important step in the proposed watermarking scheme. In general, we apply the Delaunay tessellation on the strong IFPs to generate triangles, and use angle degrees to find the matched triangles between the original and probe

images. We further use these matched triangles to find the possible geometric attacks. Table 1 lists four image texture dependent parameters and the number of strong IFPs determined by applying our image-texture-based improved and robust Harris corner detector on four images with different textures. These four parameters are *Ratio* (the factor for classifying image textures), *Type* (the texture decided by Eq. (1)), *D* (the diameter of the circular window of the robust and improved Harris corner detector), and *SNum* (the number of embedding subimages determined by perceptual analysis). It clearly shows that diameter *D* is determined by the image texture. That is, the more complicate the texture, the larger the diameter *D*. The value of *SNum* indicates the distribution of the perceptually high texture within an image. These adaptive parameters are automatically determined based on image textures. They improve the accuracy in finding the image-content-based strong IFPs and the robustness in resisting geometric and common image processing attacks on different textured images. We also observe that the number of IFPs is less than 35 for all the test images with different textures. This observation clearly demonstrates that our improved and robust Harris corner detector does regulate the number of IFPs. It also indicates that the cost of saving IFPs for watermark synchronization is minimal compared with the cost of saving the host image.

Table 1 also lists the ratios between the number of matched triangle pairs for determining the geometric transformation and the total number of matched triangle pairs under four random geometric attacks. All simulation results yield ratios of larger than 85%, which indicate a high accuracy in finding the possible geometric transformation a probe image may undergo. When comparing the results between the images, it should be noted that the number of matched triangle pairs is not linearly related to the number of strong IFPs due to the sensitivity of the IFPs to different attacks. However, our improved and robust Harris corner detector generates relatively strong IFPs to reduce the synchronization errors. In addition, two properties of the Delaunay tessellation always ensure that there are enough matched triangles, as indicated by high ratios in Table 1, for restoring the probe image.

Table 1. Image adaptive parameters and ratios under different attacks

Images	**Images Adaptive Parameters**				**Robust IFPs**	**Geometric Attacks**			
	Ratio	**Type**	**D**	**SNum**		**(a)**	**(b)**	**(c)**	**(d)**
Lena	0.002	Med	41	3	27	15/16	23/25	10/11	15/16
Baboon	0.01	High	51	9	33	18/20	25/27	12/14	12/12
Pepper	0.0013	Low	34	4	25	21/22	17/18	13/15	16/17
Plane	0.0033	Med	41	6	28	17/18	24/26	10/10	15/16

5.3 Comparison with Feature-Based RST Robust Watermarking Schemes

We compare the proposed scheme with three feature-based RST robust watermarking schemes, namely Tang's scheme [19], Wang's scheme [20], and Bas's scheme [15] in Tables 2 through 4. Each gray cell indicates that the corresponding method fails to detect the watermark under the corresponding distortion. Table 2 summarizes the

detection results compared with the schemes of Tang [19] and Wang [20] against common image processing attacks. Table 3 summarizes the detection results compared with the schemes of Tang [19] and Wang [20] against desynchronization attacks. These two tables show the ratio between the number of correctly detected watermarked embedding regions and the number of original embedded watermarked embedding regions. Here, we use the term "detection rate" to denote it. Similarly, the detection rate in Tang's scheme refers to the fraction of correctly detected watermark embedding disks and the detection rate in Wang's scheme refers to the fraction of correctly detected watermark embedding LCR (Local Characteristic Regions). That is, the subimages in our scheme correspond to the disks in Tang's scheme and the LCRs in Wang's scheme, respectively.

Table 2. Comparison of the detection rates (i.e., the fraction of correctly detected watermark embedding regions) under common image processing attacks. (1) Median filter (3×3). (2) Shapening (3×3). (3) Gaussian noise. (4) JPEG 70%. (5) JPEG 50%. (6) JPEG 30%. (7) Median filter (3×3) + JPEG 90%. (8) Sharpening (3×3) + JPEG 90%.

Attacks	**Lena**			**Baboon**			**Pepper**		
	Our Method	Wang's Method	Tang's Method	Our Method	Wang's Method	Tang's Method	Our Method	Wang's Method	Tang's Method
(1)	1.0	0.5	0.125	1.0	0.583	0.182	0.5	0.5	0.25
(2)	1.0	0.5	0.375	1.0	0.5	0.364	1.0	0.625	0.5
(3)	1.0	0.333	0.25	1.0	0.333	0.273	1.0	0.5	0.5
(4)	1.0	0.667	0.625	1.0	0.75	0.728	1.0	0.875	0.75
(5)	1.0	0.667	0.5	1.0	0.75	0.545	0.5	0.75	0.5
(6)	1.0	0.333	0.25	1.0	0.75	0.364	0.5	0.5	0
(7)	1.0	0.5	0.125	1.0	0.583	0.091	0.5	0.5	0.25
(8)	1.0	0.5	0.375	1.0	0.5	0.182	1.0	0.75	0.75

Table 2 clearly shows that our scheme and Wang's scheme successfully pass all the tests while Tang's scheme fails median filtering, JPEG compression 30%, and median filtering with JPEG compression 90%. Our scheme shows better stability than Wang's scheme due to the higher detection ratios under all the attacks. Our method also yields almost equal performance on all three images due to the regulation of the number of strong IFPs on images with different textures whereas Tang's method performs better on high textured images such as Baboon. In summary, the performance of our scheme is much more stable under different attacks in the comparison with Tang's scheme and Wang's scheme. One reason is that our relatively strong IFPs are more stable than those found by the Mexican hat detector and the scale invariant Harris-Laplace detector. These robust IFPs ensure more accurate synchronization between the probe and original watermarked images. Another reason is that the watermark is embedded in the mid-frequencies, which are in positions that are unlikely changed by common image processing attacks.

Table 3 clearly shows that our scheme performs the best in all the desynchronization attacks except the large cropping and local random bending attacks. Specifically,

our scheme can successfully resist several attacks Tang's scheme failed to tackle, namely, random relatively large rotations, scaling ratios, and any combination of RST attacks. Our scheme can handle large scaling attacks, which Wang's scheme failed to handle. These successes are mainly due to the following three reasons: 1) Our proposed robust and improved Harris corner detector finds relatively strong IFPs which are more resistant to desynchronization attacks. 2) The strong IFPs-based Delaunay triangle matching and image restoration technique ensures enough matched triangles for accurate self-synchronization under a variety of RST and aspect ratio changing attacks, while the local characteristic region derived from the scale-space theory is not geometrically invariant space, as explained in Wang's scheme. 3) The DFT domain itself is robust to translation and moderate cropping so it can accommodate the cropping attacks and further compensate the slightly inaccurate IFPs-based geometric correction. However, our scheme is more vulnerable to the local random bending attacks than both Tang's and Wang's schemes, due to the possible inaccurate image restoration resulted from the shifts of the bending IFPs or the shifts of the embedding pixels in a limited number (usually less than 9) of the embedding subimages. A large amount of embedding disks in Tang's and Wang's schemes make it more robust to local bending attacks since bending may not affect all the embedding disks and LCRs. Our scheme is also more vulnerable to large cropping attacks since the potential embedding regions may be removed. Tang's and Wang's schemes may survive large cropping attacks as long as a few disks and the LCRs containing the IFPs are not cropped.

Table 3. Comparison of the detection rate (i.e., the fraction of correctly detected watermark embedding regions) under geometric attacks. (1) Removed 8 rows and 16 columns. (2) Cropping 55%. (3) Rotation 5º. (4) Rotation 15º. (5) Rotation 30º. (6) Translation *x*-10 and *y*-19. (7) Scaling 0.6. (8) Scaling 0.9. (9) Scaling 1.4. (10) Local random bening. (11) Cropping 10% and JPEG 70%. (12)Rotaion 5º + Scaling 0.9. (13) Translation *x*-10 and *y*-10 + Rotation 5º + Scaling 0.9.

Attacks	**Lena**			**Baboon**			**Pepper**		
	Our Method	Wang's Method	Tang's Method	Our Method	Wang's Method	Tang's Method	Our Method	Wang's Method	Tang's Method
(1)	1.0	0.833	0.125	0.780	0.583	0.182	0.375	0.75	0
(2)	0	0.667	0.125	0	0.5	0.182	0	0.625	0
(3)	1.0	0.667	0.375	1.0	0.417	0.273	1.0	0.625	0.25
(4)	1.0	0.5	0.125	1.0	0.333	0.182	0.75	0.5	0
(5)	1.0	0.333	0	1.0	0.333	0	0.75	0.25	0
(6)	1.0	0.833	0.25	1.0	0.833	0.727	1.0	0.625	0.25
(7)	1.0	0.167	0	0.78	0.167	0.091	0.25	0.375	0.25
(8)	1.0	0.5	0.125	1.0	0.417	0.182	0.5	0.375	0.25
(9)	0.5	0.167	0	0.25	0.083	0	0.25	0.25	0
(10)	0	0.5	0.375	0	0.583	0.364	0.25	0.75	0.25
(11)	1.0	0.333	0.25	1.0	0.417	0.182	0.67	0.5	0.25
(12)	1.0	0.5	0	1.0	0.417	0.091	0.5	0.625	0
(13)	0.66	0.333	0	1.0	0.25	0.091	0.5	0.125	0

Table 4. The comparison between the proposed method and Bas's method [15] under different processing and geometric distortion attacks

Attacks	Lena		Airplane		Car	
	Our Method	Bas's Method	Our Method	Bas's Method	Our Method	Bas's Method
No Attack	√	√	√	√	√	√
8×8 Median Filtering	√	×	√	×	√	×
3×3 Gaussian Filtering	√	√	√	√	√	√
Shearing *x*-1%, *y*-1%	√	√	√	√	√	√
Shearing *x*-0%, *y*-5%	√	√	√	√	√	√
Shearing *x*-5%, *y*-5%	√	√	√	√	√	√
Rotation 10º	√	√	√	√	√	√
Scaling 90%	√	√	√	√	√	√
Scaling 80%	√	√	√	√	√	√
JPEG 80%	√	√	√	√	√	√
JPEG 45%	√	√	√	√	√	√
JPEG 30%	√	×	√	×	√	√
Stirmark General Attacks	√	√	√	√	√	√

Table 4 compares the proposed scheme with Bas's scheme [15], which is a feature-based spatial domain method, in terms of the robustness against different attacks such as median filtering, Gaussian filtering, shearing, rotation, scaling, JPEG compression, and Stirmark general attacks. All these tests are performed on images of Lena, Airplane, and Car for a fair comparison. The comparison results upon various common image processing attacks and geometric distortions are listed side by side in Table 4. Since Bas's scheme did not record the detection ratio, we use a " √" to indicate the method successfully detects the watermark after the attack and a " ×" to indicate the method fails to detect the watermark. As shown in Table 4, both methods successfully pass the small shearing, rotation, scaling, and Stirmark general attacks. However, our method has advantages on common image processing tests such as 8×8 median filtering and 30% JPEG compression due to the following reasons: 1) Our method embeds the watermark into the mid-frequencies which are unlikely to be changed by common image processing attacks, while Bas's method introduces an interpolation problem when doing the watermark triangle wrapping in detection. 2) Our method embeds the watermark in the DFT domain, while Bas's scheme is suitable for directly adding watermarks into the spatial domain due to irregular shape of the embedding area. This makes the method vulnerable to image processing distortions.

5.4 Comprehensive Simulation Results under Stirmark Attacks

We performed a variety of attacks on 105 8-bit watermarked grayscale images of size 512×512 using Stirmark 3.1. These images are evenly distributed with high, medium, and low textures according to Eq. (1). That is, the database contains 35 images for each texture level. The overall average PSNR value for these 105 watermarked images is 41.73db. Fig. 6 demonstrates the simulation results of 15 kinds of common attacks on the 105 watermarked images. The simulated attacks are listed on *x*-axis

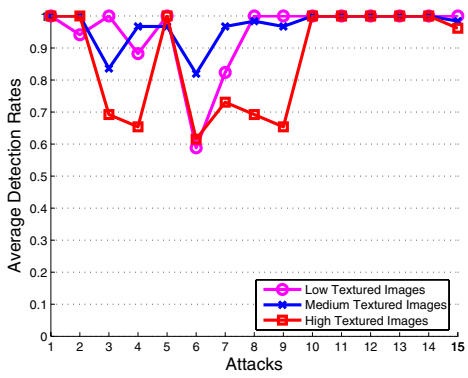


Fig. 6. The average successful detection rates for three kinds of textured images under desynchronization attacks

where all the numerically labeled attacks sequentially correspond to a category of distortions including no attacks, translation, scaling, rotation, cropping up to 5%, linear geometric transform, row and column removal with a maximum of 20 rows and columns removed, median filtering, mean filtering, sharpening, Gaussian filtering, histogram equalization, and JPEG compressions with quality factors of 50%, 40%, and 30%. All the filtering operations use the maximum filter size of 7×7. Each distortion category (i.e., numbers 2 to 12 on *x*-axis in Fig. 6) contains 3 random attacks. The *y*-axis summarizes the average detection rates of all images in each texture level under each distortion category. Fig. 6 clearly demonstrates that our scheme achieves good robustness under both image processing and geometric distortions and performs the worst for low and high textured images under the linear geometric attacks. Specifically, the average detection rates for all simulated geometric attacks are 94.89%, 89.75%, and 80.45% for medium, low, and high textured images, respectively. The average detection rates for all simulated image processing attacks are 100%, 100%, and 93.15% for medium, low, and high textured images, respectively. The average detection rates for all simulated attacks are 97.45%, 94.88%, and 86.8% for medium, low, and high textured images, respectively. The overall average detection rate for all images under all simulated attacks is 93.04%.

In summary, the all-around result of our proposed watermark scheme outperforms the peer feature-based schemes. It yields positive detection results for most images with low, medium, and high textures under different desynchronization distortions. It also works well on highly textured images due to the relatively large number of strong IFPs for image restoration. Our proposed scheme achieves a good balance between invisibility and robustness. Specifically, the robust and improved Harris corner detector is capable of finding the relatively strong IFPs for different textured images. The Delaunay triangle matching and image restoration scheme is able to efficiently minimize the synchronization errors and eliminate fake IFPs showed up in the high and extremely high textured images in the matching process. The spread spectrum

embedding and detection makes our scheme more resistant to image processing attacks. The perceptually highly textured subimage based embedding scheme helps our system to survive some localized image attacks in Stirmark. However, our scheme does not perform well on extremely low textured images due to its insufficient number of IFPs. It also fails the JPEG compression with a quality factor of lower than 30% due to the missing IFPs resulted from high compression.

6 Conclusions

In this paper, we propose a novel and effective content-based desynchronization resilient watermarking scheme. The major contributions consist of:

- Robust and improved Harris corner detector: This detector is capable of finding the relatively strong IFPs in different textured images. These IFPs are more stable than the feature points extracted by Bas's, Tang's, and Wang's schemes due to the superior performance of the Harris corner detector, the additional noise reduction, the regulation of the density of the IFPs based on the image dimension and texture, and the intersection of IFPs extracted from the pre-attacked images. Consequently, they are more robust against desynchronization attacks.
- Perceptually highly textured subimage based watermark embedding: These embedding subimages carry the same copy of the bipolar watermark bit sequence to improve the robustness of transmitted watermark bits. They also aid the proposed watermark scheme in surviving some localized image attacks in Stirmark.
- Spread-spectrum-based blind watermark embedding and retrieval in the DFT domain: The spread spectrum scheme makes our scheme more resistant to common image processing attacks. The DFT domain ensures more resistant to translation and moderate cropping. The blind retrieval scheme does not require the original image once the probe image is aligned with the original image.
- Delaunay triangle matching and image restoration: This scheme can efficiently eliminate fake IFPs showed up in the highly textured images and accurately determine the possible transformation a probe image may undergo. The determined transformation effectively reduces the synchronization errors without introducing substantial interpolation errors as in Bas's, Tang's, and Wang's schemes, where the affine transform, interpolation, and the image normalization are applied.

The proposed method is robust against a wide variety of tests as indicated in the experimental results. Particularly, it is more robust against JPEG compression and the combination of the geometric distortions with large scaling ratios and rotations than other feature-based watermarking techniques. It works successfully for images with low, medium, and high textures. It can be further improved by developing a more reliable feature extraction method under severe geometric distortions and a more efficient and accurate triangle matching and image restoration method.

The algorithm can be applied to color images by embedding watermark in the luminance component of the color image. In the real world, this watermarking technique can be applied to a lot of different areas, such as photograph, audio, and video.

References

1. Peticolas, F., Anderson, R., Kuhn, M.: Attacks on Copyright Marking Systems. In: Proc. of the 2nd Workshop on Information Hiding, pp. 218–238 (1998)
2. Pereira, S., O'Ruanaidh, J.J.K., Deguillaume, F., Csurka, G., Pun, T.: Template Based Recovery of Fourier-based Watermarks Using Log-Polar and Log-Log Maps. In: Proc. of IEEE Int. Conf. Multimedia Computing Systems, vol. 1, pp. 870–874 (1999)
3. Pereira, S., Pun, T.: Robust Template Matching for Affine Resistant Image Watermarks. IEEE Trans. on Image Processing. 9(6), 1123–1129 (2000)
4. Digimarc Corporation, US patent 5,822,436, Photographic Products and Methods Employing Embedded Information
5. Herrigel, A., Voloshynovskiy, S., Rytsar, Y.B.: Watermark Template Attack. In: Proc. of SPIE Security and Watermarking of Multimedia Contents III, vol. 4314, pp. 394–405 (2001)
6. O'Ruanaidh, J.J.K., Pun, T.: Rotation, Scale, and Translation Invariant Digital Image Watermarking. In: Proc. of IEEE Int. Conf. on Image Processing, pp. 536–539 (1997)
7. O'Ruanaidh, J.J.K., Pun, T.: Rotation, Scale, and Translation Invariant Spread Spectrum Digital Image Watermarking. Signal Processing 66, 303–317 (1998)
8. Zheng, D., Zhao, J., El Saddik, A.: RST-Invariant Digital Image Watermarking Based on Log-Polar Mapping and Phase Correlation. IEEE Trans. on Circuits and Systems for Video Technology. 13(8), 753–765 (2003)
9. Lin, C.Y., Wu, M., Bloom, J.A., Cox, I.J., Miller, M.L., Lui, Y.M.: Rotation, Scale, and Translation Resilient Watermarking for Images. IEEE Trans. on Image Processing. 10(5), 767–782 (2001)
10. Alghoniemy, M., Tewfik, A.H.: Geometric Distortion Correction Through Image Normalization. In: Proc. of IEEE Int. Conf. Multimedia Expo., vol. 3, pp. 1291–1294 (2000)
11. Alghoniemy, M., Tewfik, A.H.: Image Watermarking by Moment Invariants. In: Proc. of IEEE Int. Conf. Image Processing, vol. 2, pp. 73–76 (2000)
12. Alghoniemy, M., Tewfik, A.H.: Geometric Invariance in Image Watermarking. IEEE Trans. on Image Processing. 13(2), 145–153 (2004)
13. Kim, H.S., Lee, H.K.: Invariant Image Watermark Using Zernike Moments. IEEE Trans. on Circuit and Systems for Video Technology. 13(8), 766–775 (2003)
14. Xin, Y., Liao, S., Pawlak, M.: Geometrically Robust Image Watermarking Via Pseudo-Zernike Moments. In: Proc. of Canadian Conf. Electrical and Computer Engineering, vol. 2, pp. 939–942 (2004)
15. Bas, P., Chassery, J.M., Macq, B.: Geometrically Invariant Watermarking Using Feature Points. IEEE Trans. on Image Processing. 11(9), 1014–1028 (2002)
16. Seo, J., Yoo, C.: Localized Image Watermarking Based on Feature Points of Scale-Space Representation. Pattern Recognition 37(7), 1365–1375 (2004)
17. Seo, J., Yoo, C.: Image Watermarking Based on Invariant Regions of Scale-Space Representation. IEEE Trans. on Signal Processing. 54(4), 1537–1549 (2006)
18. Lee, H., Kim, H., Lee, H.: Robust Image Watermarking Using Local Invariant Features. Optical Engineering 45(3), 1–11 (2006)
19. Tang, C.W., Hang, H.M.: A Feature-Based Robust Digital Image Watermarking Scheme. IEEE Trans. on Signal Processing. 51(4), 950–959 (2003)
20. Wang, X., Wu, J., Niu, P.: A New Digital Image Watermarking Algorithm Resilient to Desynchronization Attacks. IEEE Trans. on Information Forensics Security. 2(4), 655–663 (2007)

21. Harris, C., Stephen, M.: A Combined Corner and Edge Detector. In: Proc. of the 4th Alvey Vision Conf., pp. 147–151 (1988)
22. Pickholtz, R.L., Schilling, D.L., Milstein, L.B.: Theory of Spread Spectrum Communications – A Tutorial. IEEE Trans. on Communications. COM 30, 855–884 (1982)
23. Hwang, M.S., Chang, C.C., Hwang, K.F.: A Watermarking Technique Based on One-Way Hashing Functions. IEEE Trans. on Consumer Electronics 25, 286–294 (1999)
24. Hartung, F., Girod, B.: Watermarking of Uncompressed and Compressed Video. Signal Processing 66, 283–301 (1998)
25. Pranata, S., Guan, Y.L., Chua, H.C.: BER Formulation for the Blind Retrieval of MPEG Video Watermark. In: Petitcolas, F.A.P., Kim, H.-J. (eds.) IWDW 2002. LNCS, vol. 2613, pp. 91–104. Springer, Heidelberg (2003)
26. Barni, M., Podilchuk, C.I., Bartolini, F., Delp, E.J.: Watermark Embedding: Hiding a Signal within a Cover Image. IEEE Communications Magazine 39, 102–108 (2001)
27. Bertin, E., Marchand-Maillet, S., Chassery, J.M.: Optimization in Voronoi Diagrams. Kluwer, Dordrecht (1994)
28. Barber, C.B., Dobkin, D.P., Huhdanpaa, H.T.: The Quickhull Algorithm for Convex Hulls. ACM Trans. on Mathematical Software. 22(4), 469–483 (1996)
29. Hsieh, M.S., Tseng, D.C.: Perceptual Digital Watermarking for Image Authentication in Electronic Commerce. Electronic Commerce Research 4, 157–170 (2004)