A New Watermarking Technique for Multimedia Protection

Chun-Shien Lu, Shih-Kun Huang, Chwen-Jye Sze, and Hong-Yuan Mark Liao*

Institute of Information Science,
Academia Sinica, Taipei, Taiwan.
E-mail: {lcs, liao}@iis.sinica.edu.tw

Abstract

A robust watermarking scheme for hiding binary or gray-scale watermarks in digital images is proposed in this chapter. Motivated by the fact that a detector response (a correlation value) only provides a soft evidence for convincing jury in courtroom, embedded watermarks are designed to be visually recognizable after retrieval. To strengthen the existence confidence of a watermark, visually significant transformed components are selected. In addition, a relocation technique is presented to tackle geometric-distortion-based attacks without using any registration scheme. Finally, a semi-public watermark detector which does not require use of the original source is proposed for the purpose of authentication. Experimental results demonstrate that our approach satisfies the common requirements of image watermarking, and that the performance is superb.

Keywords: Human visual system   Wavelet transform   Watermarking   Modulation   Attacks

*Corresponding author

1 INTRODUCTION

1.1 WATERMARKING

Owing to the popularity of the Internet, the use and transfer of digitized media are increasing. However, this frequent use of Internet has created the need for security. Therefore, it is imperative to protect information to prevent intentional or unwitting use of information belonging rightful owners. A commonly used method is to insert watermarks into original information to declare rightful ownership. This is the so-called watermarking technique. A watermark can be a visible or invisible text, binary stream, audio, image or video. It is embedded in an original source and is expected to tolerate attacks of any kind. A valid watermarking procedure enables one to judge the owner of media contents via a retrieved watermark even if it is attacked and is, thus, fragmentary.

An effective watermarking procedure should satisfy or consider the following requirements:

1. **Transparency:** The inserted watermark should be perceptually invisible. This demand is most challenging for images with large homogeneous areas.

2. **Robustness:** A secure watermark should be difficult to remove or destroy, or at least the watermarked image must be severely degraded before the watermark is lost. Typical intentional or unwitting attacks include:

    - Common digital processing: A watermark should survive after image blurring, compression, dithering, printing and scanning, etc.

    - Subterfuge attacks (collusion and forgery) [4]: A watermark should be resistant to combinations of the same image watermarked with different watermarks (collusion). In addition, a watermark should be robust to repeatedly watermarking (forgery).

    - Geometric distortions: A watermark should be able to survive attacks which use general geometric transformation, such as cropping, rotation, translation, and scaling.

3. **Capacity:** Capacity [23, 27] is the issue allowing to embed the maximum number of distinguishable watermarks. Cox *et al.* [4] discovered that the significant components of an image have a perceptual capacity that allows watermark insertion without perceptual degradation. In other words, any attack trying to remove or destroy embedded watermarks will influence the significant components of an image and thus lead to fidelity degradation.

4. **Public watermarking:** Authentication without using original sources is necessary to two reasons [35]: (1) searching for the original image in large digital libraries is time-consuming; (2) application of "web-crawling" detection. Source-based and destination-based approaches are two major watermarking schemes depending on its usage [23]. The source-based approach focuses on ownership authentication/identification. A unique watermark is detected or extracted to determine the owner of data. It is desireable to confirm ownership by retrieving the watermark without the original image. On the other hand, the destination-based method can be used to trace the end-user when illegal use, such as reselling occurs. Under these circumstances, the existence of original images is allowed.

5. **Resolving rightful ownership deadlock:** A watermark should unambiguously certify the true occupant. Craver *et al.* [5] took the initiative in presenting and solving this problem. It is in fact very important and is usually ignored in most watermarking schemes. Qiao and Nahrstedt also solved the problem of rightful ownership and were the first ones to provide protection of customers' rights [24].

1.2 OVERVIEW

In the literature, Koch and Zhao [13] transformed an image by using block-DCT transform and then utilized a pseudorandom number generator to select a subset of blocks. A triplet of blocks with midrange frequencies was slightly revised to yield a binary sequence watermark. This seems reasonable because low frequency components are perceptually important but easy to sense after modification, and high frequency components are easy to tamper with. Macq and Quisquater [18] suggested hiding data in the least significant bits such that the embedded data is imperceptible. Their watermark is easy to destroy using attacks such as low-pass filtering. Cox *et al.* [4] proposed a global DCT-based spread spectrum approach to hide watermark. They believe that signal energy present in any frequency is undetectable if a narrowband signal is transmitted over a much broader bandwidth. Ideally, this will lead to a watermark which spreads over all frequencies so that the energy in any single frequency is very small and, thus, undetectable. Their watermark is of fixed length and is produced using *Gaussian* distribution with zero mean and unit variance. They distributed as fairly as possible the watermark to the first 1000 largest AC coefficients. An objective measurement was proposed to evaluate the similarity between the original and extracted watermarks. Hsu and Wu [11] used multiresolution representations for the host image and the binary watermark. The middle frequencies in the transformed wavelet domain were selected for modification using a residual mask. Their method has been shown to be effective for JPEG-based compression at higher bit rates. A similar work was proposed by Hsu and Wu using discrete cosine transform [12]. Some commercially available watermarking software programs, such as SysCoP and EikonaMark [20], also embed an identification word of finite length into specified positions (determined by a secret key) in an image. However, there are limitations in the above mentioned methods: (i) the length of a watermark is short and bounded; (ii) it is unclear where the watermark can be hid and to what extent modification can be done to meet the transparency and robustness requirements.

In order to improve the above-mentioned drawbacks, the characteristics of the human visual system (HVS) have been incorporated into some schemes [6, 23, 32]. It is very meaningful and reasonable to take HVS into consideration because of its inherent features. If one can modify an image based on rules taken from the human visual system, then it will be easier to generate an imperceptible watermark with maximum capacity, and the length and strength of a watermark can be adaptive to the host image. Many existing watermarking techniques generate binary sequences or small texts as watermarks. The former are visually meaningless while the latter might be easily removed or destroyed. For example, the SysCoP watermarking technique hides an eight-character watermark and uses an eight-digit secret key. Cox *et al.*'s watermark [4] is composed of a binary sequence which is statistically undetectable but visually meaningless. Hsu and Wu's watermark [11] is a recognizable binary text, but it is simple and easy to destroy. Basically, a watermarking scheme that does not sufficiently utilize the capacity of a host image may cause the potential length and strength of a watermark to be bounded. Podilchuk and Zeng [23] proposed a perceptual model based watermarking scheme, but their watermark is image dependent. In other words, their watermarks cannot be specified in advance.

One of the important issues in watermarking is the need to access the original image to reliably extract the embedded watermarks [4, 11, 12, 23]. Without the original image, the extracted watermarks may be degraded [2, 14]. Some researches [25, 33] have pointed out that the original image may help to overcome geometric transformation attacks. However, for security reasons, a watermarking technique which does not use the original source is always preferable.

Another important issue in watermarking is to the need to design an effective watermarking technique such that embedded watermarks will survive attacks. StirMark [20] and unZign [34] are two softwares available on the WWW which can be used to verify the robustness of watermarking. StirMark and unZign are powerful because watermarks generated by most existing watermarking techniques cannot resist their attacks. To the best of our knowledge, there is still no research report in the watermarking literature that provides any results of robustness evaluation under attacks from StirMark and unZign.

Some surveys regarding watermarking methods can be found in [4, 8, 10, 22, 32, 38]. In this chapter, we propose an HVS-based watermarking algorithm with both gray-scale and binary watermarks taken into consideration. Our method will attain the following goals:

- the watermark is as large as possible and as strong as possible;
- embedded watermarks are meaningful and recognizable;
- watermarks are resistant to common attacks, in particular from StirMark and unZign;
- the watermark recovery process does not use the original source.

2 HUMAN VISUAK SYSTEM BASED MODULATION

To satisfy the demand for maximum perceptual capacity, a model based on the human visual system is introduced here. Some previously proposed systems [1, 36] that base their designs on the human perceptual model have played an important role in the field of image compression. Basically, these systems take into account the structures of complex natural images. More specifically, masking, the effect of a visual model, refers to the fact that a component in a given visual signal may become imperceptible in the presence of another signal called a masker. This refers to a situation where a signal raises the visual *threshold* for other signals around it. Three commonly encountered masking types are frequency masking, luminance masking, and contrast masking. The first one is image-independent but depends on the visual environment. The other two are image-dependent. Frequency masking specifies the sensitivity of human eyes to sine wave gratings at various frequencies. For a given visual distance and display resolution, it can determine the just noticeable distortion (JND) for each spatial frequency from specified wave functions. Psychologists have experimented with several contrast sensitivity functions (CSF) from some specific wave functions, such as the DCT basis function [19] and wavelet [37]. Since wavelet transform is very powerful in image representation, we will use the wavelet-based frequency masking model [37] for watermarking. The frequency masking map with a four-level wavelet transform and display visual resolution (DVR) 32 [37] is illustrated in Fig. 1. Whiter gray values imply higher JND values.

Two very popular watermarking techniques which employed perceptual significance were presented in [4, 23]. Cox *et al.* [4] used spread spectrum embedding to hide a watermark:

$$I_i^* = I_i(1 + \alpha \cdot n_i), \tag{1}$$

where I_i and I_i^* are DCT coefficients before and after modulation, respectively, and n_i is the watermark sequence. α is a weight that controls the trade-off between transparency and robustness. In [23], Podilchuk and Zeng presented two watermarking schemes based on human visual model: image adaptive-DCT (IA-DCT) and image adaptive wavelet (IA-W) schemes. The watermark embedder for both the IA-DCT and IA-W approaches can be described in general as

$$I_{u,v}^* = \begin{cases} I_{u,v} + J_{u,v} \cdot n_{u,v}, & \text{if } I_{u,v} > J_{u,v}, \\ I_{u,v}, & otherwise \end{cases} \tag{2}$$

where $J_{u,v}$ is the masking value of a DCT or wavelet based visual model, and $n_{u,v}$ is the sequence of watermark values. It is found from both in embedding schemes that modulations take place in the perceptually significant coefficients with the modification quantity specified by a weight. The weight is heuristic [4] or depends on visual model [23]. Cox *et al.* [4] and Podilchuk and Zeng [23] both adopted a similar detector response measurement described by

$$Sim(n, n^*) = \frac{n \cdot n^*}{\sqrt{n^* \cdot n^*}}, \tag{3}$$

where n^* is the extracted watermark sequence. If the signs of a pair of elements in n and n^* are the same, then they contribute to the detector response. A higher value of $Sim(n, n^*)$ means a higher probability that n^* is a genuine watermark. High correlation values can only be achieved if the transformed coefficients are modulated and distorted along the same direction during the embedding and attacking processes, respectively. This is very important if a watermark detector is to get a higher similarity value. However, we find from Eqs. (1) and (2) that the directions of modulation are random. A positive coefficient can be updated with a positive or negative quantity, and a negative coefficient can be changed with a positive or negative quantity. Furthermore, the works of [4, 23] didn't consider the relationship between the signs of correlation pairs (the transformed coefficients and the watermark values). This is the reason why many attacks can successfully defeat the above mentioned watermarking schemes.

In this chapter, we shall seriously treat the modulation problem. The modulation strategies described in [4, 23] are called random modulation here in contrast to our attack-adaptive modulation mechanism. In [17], we mentioned that if a modulation strategy operates by adding a negative quantity to a positive coefficient or by adding a positive quantity to a negative coefficient, then we call it negative modulation. Otherwise, it is called positive modulation if the sign of the added quantity is the same as that of the transformed coefficient.

3 THE PROPOSED WATERMARKING ALGORITHMS

We shall develop two watermarking algorithms based on the assumption that the original image (host image) is gray-scale. Our watermark can be either a binary watermark or a gray-scale watermark, and its maximum size can be as large as that of the host image. The wavelet transform adopted in this work has constrained such that the size of the lowest band is 16×16. A four-level decomposition via wavelet transform is shown in Fig. 1(a).

3.1 WATERMARK STRUCTURES

A gray-scale watermark with "ACADEMIA SINICA" and its corresponding Chinese text "中央研究院" in a pictorial background is shown in the middle left part of Fig. 4. The bottom left part of Fig. 4 shows a binary watermark with the Chinese text "中央研究院", which means "ACADEMIA SINICA." For embedding binary watermarks, we do not take the irrelevant background [11, 12, 35] into account; rather we hide the watermark pixels (on the foreground) only. That is, the watermark pixels or the foreground pixels constituting the texts are embedded. For gray-scale watermarks, the characteristic we adopt is that a recognizable watermark is extracted for subjective judgement instead of an objective decision determined by some similarity measurements. Another benefit of using a gray-scale watermark is that it can avoid checking the existence of a single watermark pixel. Under different attacks, a gray-scale watermark has more chance to survive. This is because a gray-scale watermark can always preserve a certain degree of *contextual information* even after attacks. It is well known that to embed a gray-scale watermark with its original intensities is extremely difficult since the transparency requirement is easily violated. Therefore, we shall apply the human visual model to make the embedded gray-scale watermark look like the original, but it is in fact "compressed."

3.2 THE HIDING PROCESS

Our watermarking technique will be detailed in this section. First, the host image and a visually recognizable gray-scale watermark are transformed by discrete wavelet transform. It is noted that the binary watermark does not have to be transformed.

3.2.1 IMAGE-INDEPENDENT AND IMAGE-DEPENDENT PERMUTATIONS

In order to make the transformed gray-scale watermark and the non-transformed binary watermark statistically undetectable, they are spatially transformed using a chaotic system called “toral automorphism” [35]. After the chaotic mixing process, the watermark is converted into a noise-like form, which can guarantee undetectability. Basically, the toral automorphism is a kind of image-independent permutation. Next, an image-dependent permutation is executed to increase the security level and to select places for hiding the two types of watermarks. The image-dependent permutation used in our approach is in the form of a mapping function. The host image and the gray-scale watermark in the wavelet domain are a one-to-one mapping. We design such a mapping function based on the significance of the wavelet coefficients. The security level is raised because the watermarks are embedded into those components which have larger coefficients. These significant coefficients are more secure than the insignificant coefficients especially when compression is performed. The reason is that under compression attack, the significant coefficients have are more likely to survive. That is, the amount of modulation is proportional to the scalogram of wavelet transform. The concept of significance is inspired by Shapiro’s EZW compression scheme [28]. The larger the magnitude of a wavelet coefficient, the more significant it is. After sorting the wavelet coefficients of the host image and the gray-scale watermark, the mapping function $m(.,.)$ is defined as

$$m(x_h, y_h) = (x_m, y_m), \tag{4}$$

where (x_h, y_h) represents the position of a wavelet coefficient, $H_{s,o}^{(k)}(x_h, y_h)$, in the host image; and (x_m, y_m) represents the position of a wavelet coefficient, $M_{s,o}^{(k)}(x_m, y_m)$, obtained from the watermark. Both (x_h, y_h) and (x_m, y_m) correspond to the k-th largest wavelet coefficients (where $1 \leq k \leq N \times M$ and $N \times M$ is the image size) with the same scale (s) and orientation (o), respectively.

In order to obtain better security for a binary watermark, we propose to doubly insert the active pixels of the binary watermark into the wavelet domain of the host image. Double insertion has the merit of cross supporting the existence of a watermark pixel, in particular when watermarked images are attacked. Therefore, one can consider the mapping function for hiding binary watermarks a two-to-one function:

$$\begin{aligned} m(x_p, y_p) &= (x_m, y_m), \\ m(x_n, y_n) &= (x_m, y_m), \end{aligned} \tag{5}$$

where (x_p, y_p) and (x_n, y_n) are the positions of the host image's positive and negative wavelet coefficients, respectively. (x_m, y_m) is the position of a foreground pixel in a binary watermark.

Without the mapping function(s), it is hard to guess the location of the embedded watermark. Embedding watermarks into the significant frequency components implies automatic adaptation to the local scalogram of wavelet transformed host images.

3.2.2 Compression-adaptive hiding

After completing the secret mapping process, we then consider how to modify the host image's wavelet coefficients and consider the extent of modification. The most important problem in watermarking is that the watermarked image should have no visual artifact. Previously, Bender *et al.* [3] altered the intensities of a host image within a small range and hoped the update would be perceptually unnoticed. However, they did not address clearly what the range of modification should be in order to obtain perceptual invisibility. Here, we shall take the human visual system into account.

Since images (or other media) need to be transmitted via networks, the compression procedure has to be used so that the traffic jam problem can be avoided. Therefore, any watermarking technique has to take the compression-style "attack" into consideration. In this paper, we shall embed the watermark by modulating the coefficients in the wavelet transformed domain; therefore, the modulation quanity of wavelet coefficients should be able to adapt to a general compression process. Usually, when an image is compressed at a specific ratio, the absolute values of its transformed insignificant coefficients are reduced to small values or zero. On the other hand, the absolute values of the significant coefficients are also decreased by a certain amount. We have observed that if the sign of the modulated quantity is the same as that of a wavelet coefficient to be updated, then the corresponding watermarked image will preserve better image quality after compression. However, the robustness of the compressed watermarked image will be degraded under the same conditions. This is because the robustness issue, which is closely related to detector response, has been violated. Hence, we can say that there is a trade-off between image quality and robustness. Fig. 2 illustrates this phenomenon using EZW-based compression (SPIHT) at a ratio of $80 : 1$. Figs. 2(a) and (b) show two compressed Lena images which were positively and negatively modulated, respectively. It is obvious that the face portion (including the eyes, mouth, and nose) shown in Fig. 2(a) is clearer than that in Fig. 2(b). Although the image quality of Fig. 2(a) is better than that of Fig. 2(b), we also find some apparent edges around the hat areas. This is because the number of positive and negative wavelet coefficients are not the same.

To embed a watermark safely, the negative modulation strategy is adopted. That is, the sign of a wavelet coefficient and that of its corresponding modulation quantity should be different. The positive modulation strategy that makes both the wavelet coefficient and its corresponding modulation quantity the same sign is advantageous for image quality but sacrifices robustness [16]. Another study regarding the robustness issue can be found in [17].

In what follows, we shall show how the concept of JNDs [37] can be realized and used in developing a watermarking technique. The JND-based watermarking techniques will be discussed in the subsequent sections, gray-scale watermarking first and then binary watermarking. We will divide the hiding process into two parts, the lowest frequency (LL_4) part and the part for the remaining frequencies. It is noted that the lowest frequency wavelet coefficient corresponds to the largest portion of decomposition.

JND-based Modifications on A Gray-scale Watermark: The purpose of using a gray-scale watermark instead of a binary one it to make the watermark much more easily "understood" due to its stronger contextual correlation. The watermarking process for a gray-scale watermark is as follows:

$$H^m_{s,o}(x_h, y_h) = \begin{cases} H_{s,o}(x_h, y_h) + sgn(H_{s,o}(x_h, y_h)) \times J_{s,o}(x_h, y_h) \times \frac{|M_{s,o}(x_m, y_m)|}{max(M_{s,o}(.,.))} \times w, \\ \qquad\qquad\qquad\qquad \text{if } |H_{s,o}(x_h, y_h)| > J_{s,o}(x_h, y_h) \\ H_{s,o}(x_h, y_h), \\ \qquad\qquad\qquad\qquad \text{otherwise}, \end{cases} \tag{6}$$

where

$$sgn(H_{s,o}(x_h, y_h)) = \begin{cases} -1, & \text{if } H_{s,o}(x_h, y_h) \geq 0.0 \\ 1, & \text{if } H_{s,o}(x_h, y_h) < 0.0. \end{cases} \tag{7}$$

The function $sgn(\cdot)$ is designed for negative modulation. $H^m_{s,o}(x_h, y_h)$ is the wavelet coefficient of the watermarked image to be determined; $H_{s,o}(x_h, y_h)$ and $J_{s,o}(x_h, y_h)$ represent the host image's wavelet coefficient and the JND value at position (x_h, y_h), scale s, and orientation o, respectively. $M_{s,o}(x_m, y_m)$ here represents the wavelet coefficient of the gray-scale watermark. $max(M_{s,o}(.,.))$ represents the maximum $M_{s,o}(.,.)$ value obtained among different locations. The relationship between (x_m, y_m) and (x_h, y_h) has been defined in Eq. (4). w is a weight used to control the maximum possible modification that will lead to the least image quality degradation. It is defined as

$$w = \begin{cases} w_L, & \text{if } H_{s,o}(x_h, y_h) \in LL_4 \\ w_H, & \text{if } H_{s,o}(x_h, y_h) \notin LL_4. \end{cases} \tag{8}$$

When hiding a gray-scale watermark, the embedded watermark, M^e, is expressed as (according to Eq. (6))

$$M^e_{s,o}(x_h, y_h) = \begin{cases} sgn(H_{s,o}(x_h, y_h)) \times J_{s,o}(x_h, y_h) \times \frac{|M_{s,o}(x_m, y_m)|}{max(M_{s,o}(.,.))} \times w, \\ \qquad \qquad \qquad \qquad \text{if } |H_{s,o}(x_h, y_h)| > J_{s,o}(x_h, y_h) \\ 0, \\ \qquad \qquad \qquad \qquad \text{otherwise.} \end{cases} \tag{9}$$

JND-based Modification of A Binary Watermark: Each foreground pixel $s(x, y) \in \mathcal{S}$ is doubly embedded by modifying the corresponding positive and negative wavelet coefficients of the host image according to the mapping function depicted in Eq. (5). For a positive wavelet coefficient, $H_{s,o}(x_p, y_p)$, subtraction operation is triggered while addition operation is inhibited:

$$Subtraction : H^m_{s,o}(x_p, y_p) = H_{s,o}(x_p, y_p) + sgn(H_{s,o}(x_p, y_p))J_{s,o}(x_p, y_p) \times |w|. \tag{10}$$

For a negative wavelet coefficient, $H_{s,o}(x_n, y_n)$, the addition operation is triggered while the subtraction operation is inhibited:

$$Addition : H^m_{s,o}(x_n, y_n) = H_{s,o}(x_n, y_n) + sgn(H_{s,o}(x_n, y_n))J_{s,o}(x_n, y_n) \times |w|. \tag{11}$$

$sgn(\cdot)$ is defined as in Eq. (7). $H^m_{s,o}(x_i, y_i)(i = p, n)$ is the wavelet coefficient of the watermarked image; $H_{s,o}(x_i, y_i)$ and $J_{s,o}(x_i, y_i)$ represent the host image's wavelet coefficient and the JND value at position (x_h, y_h), scale s, and orientation o, respectively. The relationship between (x, y) and $(x_i, y_i)(i = p, n)$ has been defined in Eq. (5). w is a weight used to control the maximum possible modification without degrading the image quality. It can be defined as in Eq. (8) or can be generated as a sequence of random numbers for the reason discussed in Sec. 5.

After the negative modulation stage, the absolute magnitudes of the modified wavelet coefficients become smaller. The absolute value of an attacked (or modified) wavelet coefficient should retain the above relation if it corresponds to a valid binary watermark pixel. If this absolute value increases after an attack, then we conclude that its corresponding pixel does not belong to a binary watermark.

Finally, the inverse wavelet transform is applied to the modulated wavelet coefficients (in gray-scale or binary watermark hiding) to generate the watermarked image.

3.3 SEMI-PUBLIC AUTHENTICATION

Original source protection is considered extremely important in watermarking. In some watermarking schemes, if one wishes to extract watermarks from a watermarked image, the original source is required.

However, it is always preferable to detect/extract watermarks without accessing the original sources since it is dangerous to retrieve them through the Internet. In the literature, if a watermarking technique does not need the original source to extract the watermark, then the extracted watermark will be somewhat degraded [2, 14]. Here, we will present a technique to extract watermarks without using the original images. In the proposed method, an extra set of secret parameters (in fact, it is a secret image) instead of a secret key only is required. Our assertion is that it is more secure to use a secret image rather than the original source. This is because even if the secret image is intercepted, there will be no *computationally feasible* way to figure out its contents.

In the watermark recovery process, image-dependent permutation mapping is also required to locate the watermark. Hsu and Wu [11, 12] obtained this information by either saving it as a file during the embedding step or recomputing it from the original image and the watermark. In our scheme, the original image is not directly used. What we do is to combine the wavelet coefficients of the original image and the results of image-dependent permutation mapping to obtain the secret image. Under these circumstances, it is difficult for one to guess the contents correctly, and the watermark can be reconstructed without any degradation, even if the original source is absent.

Our fusion process is depicted in Fig. 3. Each pixel of these images consists of 4 bytes (32 bits). In order to get a secure combination, the least significant 16 bits of the image-dependent permutation are replaced by the least significant 16 bits of the wavelet transformed image to form the first part of the secret image. Similarly, the least significant 16 bits of the wavelet transformed original image are replaced by the least significant 16 bits of the image-dependent permutation to form the second part of the secret image. Certainly, other interleaved combinations may be used, too. After the replacement process, the shaded line areas shown in Fig. 3(a) form the first part of a secret image, and the white areas form the second part of a secret image. Finally, the toral automorphism is imposed on the individual part of the fused image. There are two reasons why an unauthorized person has difficulty interpreting our secret image. First, one has to know the number of iterations as well as the parameter used in toral automorphism. Second, one has to know how the two parts of the secret image have been combined. To reconstruct the hidden watermarks, the secret image is "decompressed" such that the actual wavelet coefficients can be used for the purpose of comparison, and the results can be understood in the real world. There exists a relation between the quality of the extracted watermark and whether or not the original image is needed for authentication. It is well known that if a watermarking algorithm does not use the original image, then the quality of the extracted watermark will be somewhat degraded. When a secret image is adopted, use of the original image is no longer necessary; thus, the quality and authentication problems are simultaneously solved. Fig. 4 illustrates the whole process of our watermark hiding technique.

4 WATERMARK DETECTION/EXTRACTION

In this section, we shall describe in detail how watermarks can be extracted from a watermarked image. Usually, the watermarked image (possibly distorted) and a secret parameter or a set of parameters are the necessary components for watermark extraction. In our method, the original image is not required, but a secret image is needed. The first step in watermark extraction is to perform wavelet transform on the watermarked image. The corresponding wavelet coefficients of the host image and the positions where the extracted watermark (M^e) should be located are both retrieved by decompressing the secret image, $\mathcal{SI}$. Next, the wavelet coefficients of the host image and of the distorted watermarked image are subtracted to obtain the information of the embedded watermark.

4.1 GRAY-SCALE WATERMARK EXTRACTION

In gray-scale watermark extraction, the subtracted results are simply regarded as the wavelet coefficients of the extracted watermark. They are expressed as

$$M^e_{s,o}(m(x_h, y_h)) = H^m_{s,o}(x_h, y_h) - H^{\mathcal{SI}}(x_h, y_h), \tag{12}$$

where $H^m_{s,o}(x_h, y_h)$ represents the wavelet coefficient of the watermarked image at position (x_h, y_h) with scale s and orientation o. $m(x_h, y_h)$ is the retrieved location of the gray-scale watermark, and $H^{\mathcal{SI}}(x_h, y_h)$ is the retrieved wavelet coefficient of the host image. The relationship between (x, y) and (x_h, y_h) is a one-to-one mapping function which has been defined in Eq. (4). Finally, the inverse image-independent permutation and inverse wavelet transform are executed to obtain the reconstructed watermark.

4.2 BINARY WATERMARK EXTRACTION

The binary watermark detection process has two major steps, i.e., determining the subtracted results and designing the decision-making mechanism. In what follows, we shall describe these two steps in detail.

Subtracted results:

$$\begin{aligned} Diff_{Subtraction}(x_p, y_p) &= H^m_{s,o}(x_p, y_p) - H^{\mathcal{SI}}(x_p, y_p), \\ Diff_{Addition}(x_n, y_n) &= H^m_{s,o}(x_n, y_n) - H^{\mathcal{SI}}(x_n, y_n), \end{aligned} \tag{13}$$

where $Diff_{Subtraction}(x_p, y_p)$ and $Diff_{Addition}(x_n, y_n)$ denote the subtracted results corresponding, respectively, to the positive and negative wavelet coefficient sequences; $H^m_{s,o}(x_i, y_i)(i = p, n)$ denotes the watermarked image's positive and negative coefficients; and $H^{\mathcal{S}I}(x_p, y_p)$ and $H^{\mathcal{S}I}(x_n, y_n)$ are the retrieved positive and negative wavelet coefficients of the host image derived from the secret image ($\mathcal{S}I$).

Decision:

$$M^e(m(x_p, y_p)) = \begin{cases} s(m(x_p, y_p)), & \text{if } Diff_{Addition}(x_n, y_n) > 0 \text{ or } Diff_{Subtraction}(x_p, y_p) < 0 \\ None, & \text{otherwise,} \end{cases} \quad (14)$$

where $m(x_p, y_p)$ is the retrieved location of the binary watermark, and $s(m(x_p, y_p))$ is the value of a foreground pixel in the original binary watermark. Notice that $m(x_p, y_p)$ is the same as $m(x_n, y_n)$. The relationship between (x, y), (x_p, y_p), and (x_n, y_n) is a two-to-one mapping as indicated in Eq. (5). The decision operation shown in Eq. (14) shows the merit of alternating support of double hiding. That is, the "*OR*" operation is adopted to determine the final result from the two detected watermarks. Finally, the inverse image-independent permutation is executed to obtain the reconstructed watermark.

4.3 DEALING WITH ATTACKS INCLUDING GEOMETRIC DISTORTION

In this section, we shall present a relocation strategy to tackle with attacks generating asynchronous phenomena. StirMark [20] and unZign [34] are two very strong attackers against many watermarking techniques. From analysis of StirMark [20], it is known that StirMark introduces an unnoticeable quality loss in the image with some simple geometrical distortions. In addition, Jitter attack [21] is another type of attacker, which leads to spatial errors in images that are perceptually invisible. Basically, these attackers cause asynchronous problems. Experience tells us that an embedded watermark which encounters these attacks is often severely degraded than those encountering other attacks [16]. Moreover, the behaviors of other unknown attacks are also not predictable. It is important to deal with the encountering attack by a clever way so that a demage caused by a StirMark attack can be minimized. This is because the orders of wavelet coefficients are different before and after an attack and might be varied extremely for attacks with inherent asynchronous property. Consequently, in order to recover a "correct" watermark, the attacked watermarked image's wavelet coefficients should be relocated to proper positions before watermark detection. The relocation operation is described in the following. First, the wavelet coefficients of the watermarked image (before attacks) and those of the attacked watermarked image are sorted, respectively. The wavelet coefficients of the watermarked image after attacks are re-arranged into the same order as are those of the watermarked image before attacks. Generally speaking, by preserving the orders, demage to the extracted watermark can always be reduced. Owing to the similarity between StirMark and unZign, it is expected that unZign can be dealt with the same way. The improved results are especially remarkable for attacks including geometric distortions, such as flip, rotation, jitter, StirMark, and unZign. Fig. 5 shows the whole process of our watermark detection/extraction technique.

5 ANALYSIS OF ATTACKS DESIGNED TO DEFEAT HVS-BASED WATERMARKING

Although the human visual model helps to maximize the capacity of an embedded watermark, it is not entirely understood whether or not it provides a higher degree of security because the watermarked image carries a clue about the strength and location of the watermark [7]. For example, Cox *et al.* [4] hid their watermarks in the first *1000* AC coefficients in the DCT domain. Podilchk and Zeng's watermark [23] was embedded according to masking effects of human perception. In order to prevent the embedded watermark from being successfully attacked, we adopt the same visual model but use a different modulation function. It is noted that the positions where the watermark is hid may be blabbed in the above mentioned schemes [4, 23]. Fortunately, the amount of modification is a random number and is difficult to predict. Futhermore, the length of the hidden recognizable watermark is not fixed and is also difficult to predict. In this section, we shall start by examining the effect of a proposed reverse operation which is imposed on the HVS-based watermarked image. This operation is similar to scenario 1 in Carver *et al.*'s interpretation attack [5]. The pirate may not have to contest with the actual owner for the watermarked resources, but he/she can destroy the watermark instead. Therefore, the pirates still accomplish the goal of peculating the watermarked resources. The above mentioned attack is the so-called removal attack [10]. In what follows, we shall introduce a way to prevent a pirate from using the HVS-based concept to execute a valid attack.

Let H be the host image and M be the gray-scale watermark to be hidden, the watermarked image denoted as H^m is expressed as in Eq. (6). Suppose now that a pirate constructs a counterfeit watermark $\bar{M}$ and obtains a faked host image $\bar{H}$; they are related to the watermarked image H^m by means of a removal operation $\ominus$:

$$H^m \ominus \bar{M} = \bar{H}. \tag{15}$$

Usually, pirates will seek to fulfill the following conditions, i.e.,

$$M \subset \bar{M}$$

and

$$Sim(H, \bar{H}) \approx 1.$$

Assume that the embedding process is known to the pirates except for the secret keys. If a forged watermark is made based on knowledge of the human visual model, then we can check whether the watermarked image is vulnerable to an HVS-based removal attack. According to our watermark hiding procedure (Eq. (6)), the positive wavelet coefficients become smaller, and the negative ones larger after modulation. Therefore, a forged watermark can be made based on the concept of JNDs of the human visual model [37]:

$$\bar{M}_{s,o}(x,y) = sgn(H^m_{s,o}(x,y))J_{s,o}(x,y) \times ratio \times \bar{w}, \tag{16}$$

where $sgn(\cdot)$ has been defined in Eq. (7) and $\bar{w}$ represents the predicted weight, which can be defined as

$$\bar{w} \;=\; \begin{cases} \bar{w}_L, & \text{if } H^m_{s,o}(x,y) \in LL_4, \\ \bar{w}_H, & \text{if } H^m_{s,o}(x,y) \notin LL_4. \end{cases}$$

$\bar{w}_L$ and $\bar{w}_H$ are the predicted weights corresponding to the low and high frequencies, respectively. According to Eq. (15), the counterfeit watermark can be expressed as

$$\bar{H}_{s,o}(x,y) = H^m_{s,o}(x,y) - \bar{M}_{s,o}(x,y)$$

$$= \begin{cases} H_{s,o}(x,y) + sgn(H_{s,o}(x,y))J_{s,o}(x,y) \times \frac{|M_{s,o}(x,y)|}{max(M_{s,o}(.,.))} \times w - sgn(H^m_{s,o}(x,y))J_{s,o}(x,y) \times ratio \times \bar{w}, \\ \qquad \text{if } |H_{s,o}(x,y)| > J_{s,o}(x,y) \\ H_{s,o}(x,y), \\ \qquad \text{otherwise.} \end{cases} \tag{17}$$

In what follows, we shall analyze the positive wavelet coefficients of Eq. (17) to check the similarity between the host image and the counterfeit image. The analysis on the negative wavelet coefficients is the same and is, thus, omitted. The positive wavelet coefficients of $\bar{H}$ can be expressed as

$$\bar{H}_{s,o}(x,y) = H_{s,o}(x,y) + (\bar{w} \times ratio - \frac{|M_{s,o}(x,y)|}{max(M_{s,o}(.,.))} \times w) \times J_{s,o}(x,y). \tag{18}$$

Note that $sgn(H_{s,o}(x,y))$ is the same as $sgn(H^m_{s,o}(x,y))$. If $\bar{w} \times ratio$ can be precisely predicted and its value happens to be $\frac{|M_{s,o}(x,y)|}{max(M_{s,o}(.,.))} \times w$, then the faked image is equivalent to the original image, and the watermark may be entirely removed. But the above situation is only an ideal case. It is noted that the term $\frac{|M_{s,o}(x,y)|}{max(M_{s,o}(.,.))}$ originates in the gray-scale watermark which is designed to be hidden. As described in Sec. 3.1, only the compressed version of the designed watermark is embedded. Hence, pirates will not have any *a priori* knowledge they can use to predict the originally designed watermark. Furthermore, the number of modified coefficients (watermark length) should also be guessed approximately. Our experiments demonstrate that it is very difficult to predict a *ratio* (which is random) and then use it to remove or to degrade the embedded watermark even when $\bar{w}$ is very close to w. For a binary watermark, the random term is the weight, w, defined in Eqs. (10) and (11).

6 EXPERIMENTAL RESULTS

We have conducted a series of experiments to corroborate the effectiveness of the proposed method. Different kinds of attacks, including some digital processing and two attackers, StirMark and unZign, were used to check whether the embedded watermark was transparent and robust. StirMark and unZign were considered to be two very powerful watermark attackers because they have successfully destroyed many watermarks made by some commercially available watermarking softwares such as Digimac, SysCoP, JK_PGS, EikonaMark, and Signnum Technologies [20]. A basic requirement for a watermark attacker is that it should "destroy" the watermark while preserve the watermarked image to some extent. A watermark attacker that destroys both the watermark and the watermarked image is in fact useless. The two watermarks (one binary and one gray-scale) used in our experiments are shown on the left side of Fig. 4. The watermarked image shown in the middle right part of Fig. 4 was watermarked using the gray-scale watermark. The experimental results are reported in the following.

6.1 RESULTS OF HIDING A GRAY-SCALE WATERMARK

The popular Lena image 256 × 256 in size was used in the experiments and the watermarked image had $35dB$ PSNR with no visual artifacts.

blurring attack: The watermarked image was strongly blurred using a low-pass filter with a Gaussian variance 7 (window size of 15 × 15). Basically, the high-frequency components of the watermarked image were removed after this processing. The result is shown in Fig. 6, in which the retrieved watermark is very recognizable.

median filtering attack: Similarly, the watermarked image was median filtered using a 7 × 7 mask.

The results after median filtering attack are shown in Fig. 7, in which the retrieved watermark is again very recognizable.

image rescaling: The watermarked image was scaled to one-quarter of its original size and upsampled to its original dimensions. We can see from Fig. 8(a) that many details were lost. Fig. 8(b) shows the retrieved watermark.

JPEG compression attack: JPEG compression algorithm is one of the most attacks that the watermark shouls be resistant to. The effect of JPEG compression was examined using a very low quality factor of 5%. Fig. 9(a) shows the watermarked image after JPEG compression. Visually, the watermarked image is severely damaged because it contains apparent blocky effects. The watermark shown in Fig. 9(b) is the extracted result. It is apparent that the retrieved watermark is very clear.

EZW compression attack (by SPIHT [26]): Embedded zero tree (EZW) compression algorithm is also another one of the most attacks that the watermark shouls be resistant to. The watermarked image was attacked by SPIHT compression (a member of the EZW compression family). The compressed result is shown in Fig. 10(a) (compression ratio 128 : 1). It is obvious that the degradation is quite significant. However, the reconstructed watermark shown in Fig. 10(b) is again recognizable.

jitter attack: Jitter attack [20] is a kind of attack that introduces geometric distortions into a watermarked image. Fig. 11(a) shows the result after jitter attack with 4 pairs of columns deleted and duplicated. It is obvious that the damage caused by this attack is invisible. Fig. 11(b) shows the retrieved watermark without introducing any synchronization processing. Again, the retrieved result is still recognizable.

StirMark attack: The StirMark attacked watermarked image is shown in Fig. 12(a). All the default parameters of StirMark software were used. It is perceived that the watermarked image before and after the attack is very much the same, but it is, in fact, geometrically distorted. These distortions are illustrated in Fig. 12(b) using meshes. From the distorted meshes, it is easy to see the power of StirMark. Asynchronization is the reason why many commercially available watermarking algorithms [20] fail under its attack. Fig. 12(c) shows the extracted watermark. Fig. 12(d) shows the watermarked image attacked by applying StirMark 5 times. The extracted watermark shown in Fig. 12(e) is surprisingly good. Apparently, the proposed relocation technique broke down the effects caused by StirMark attack.

unZign attack: unZign acts like StirMark and a report comparing them can be found in [20]. Many commercial watermarking algorithms [20] have also failed under this attack. However, the retrieved results illustrated in Fig. 13(b) shows that both Chinese and English texts were well recovered.

Combination attacks: Combination attacks were also conducted to test our approach. The watermarked image after StirMark attack, JPEG compression (5%), and blurring (7 × 7) is shown in Fig. 14(a). The retrieved watermark is shown in Fig. 14(b). Fig. 14(a) rotated by 180° is shown in Fig. 14(c). The corresponding retrieved watermark is shown in Fig. 14(d). It is apparent that the extracted watermark is still a good condition.

Collusion: Several watermarks could be inserted aiming at making the original mark unreadable. Here, five different watermarks were embedded into the same host image, separately, and averaged. The watermarked image after collusion attack is shown in Fig. 15(a). Fig. 15(b) shows the retrieved watermark.

In this experiment, the strength of a attack is always set to be strong enough if the parameters of the attack are available to change. Though the image degradation is so heavy that it may not be accepted in practical applications, the mark is still easily recovered.

6.2 RESULTS OF HIDING A BINARY WATERMARK

The Rainbow image 256 × 256 in size was used in this experiment. The contents of the binary watermark were the Chinese text “中央研究院” with 1497 foreground pixels. The generated watermarked image has $27.23dB$ PSNR but again there are no visual artifacts even when the PSNR value is relatively low. Owing to our double hiding strategy, there are in total 2994 modulated coefficients. The overall performance under different kinds of attacks is summarized in Fig. 16. The sizes of the masks used for blurring and median filtering were 15 × 15 and 11 × 11, respectively. The rescaling attack was carried out in the same way as in the gray-scale watermark case. The variance of the Gaussian noise was 16, and the jitter attack delete/update 5 pair of columns. The numbers 5% and 128 : 1 below the JPEG and SPIHT compressions denote the quality factor and the compression ratio, respectively. The numbers 1 and 5 below the StirMark attacks represent the number of times StirMark attack was executed. Fig. 16 also shows a combination of attacks which include StirMark, flip, and jitter attacks. The normalized correlation value (between 0 and 1) was calculated using the similarity measurement [11]. Higher values represent better matching between the retrieved watermark and the original watermark. The highest correlation value, 1, was obtained for all the attacks described in Fig. 16. This series of experiments demonstrates that the double hiding scheme is indeed a super mechanism for embedding a binary watermark.

7 CONCLUSION

A robust watermarking scheme has been developed in this chapter to protect digital images. The JND threshold values provided by the human visual model has been introduced to decide the maximum perceptual capacity allowed to embed a watermark. The proposed scheme has the following characteristics: (1) it uses the contextual information of a gray-scale watermark; (2) applies multiple modulations to enhance security for a binary watermark hiding; (3) it adapts to the local scalogram of a wavelet transformed host image; (4) it defends against attacks with geometrical distortions using a relocation technique; (5) it authenticates without (indirectly) using the host image. Experiments have demonstrated that our watermarking scheme can achieve not only transparency, but also robustness, and that authentication can be done without the original source.

For a binary watermark considered in this chapter, the foreground pixels in the spatial domain are directly embedded in the space/frequency domain of a host image. If the number of watermark pixels is large or if the number of embedded binary watermarks is more than one, then the capacity required to hide these data should be considered. One solution is to use binary wavelet transform [29] to transform a binary watermark since the number of foreground pixels will be reduced in the wavelet domain.

In this chapter, the negative modulation strategy has been adopted. In practice, some attacks still cannot inadequately be described by negative modulation. We have dealt with the modulation problem, and a novel watermarking approach has been proposed [17]. In [17], two watermarks were embedded in a host image and played complementary roles such that at least one watermark survived after different attacks. This explains why either positive modulation [16] or negative modulation alone is not enough for multimedia protection.

References

[1] A. J. Ahumada and H. A. Peterson, "Luminance-Model-Based DCT Quantization for Color Image Compression", *Proc. SPIE*, Vol. 1666, pp. 365-374, 1992.

[2] M. Barni, F. Bartolini, V. Cappellini, and A. Piva, "Copyright Protection of Digital Images by Embedded Unperceivable Marks", *Image and Vision Computing*, Vol. 16, pp. 897-906, 1998.

[3] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for Data Hiding", *IBM Systems Journal*, Vol. 25, pp. 313-335, 1996.

[4] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, “Secure Spread Spectrum WaterMarking for Multimedia”, *IEEE Trans. Image Processing*, Vol. 6, pp. 1673-1687, 1997.

[5] S. Craver, N. Memon, B.-L. Yeo, and M. M. Yeung, “Resolving Rightful Ownerships with Invisible Watermarking Techniques: Limitations, Attacks, and Implications”, *IEEE Journal on Selected Areas in Communications*, Vol. 16, pp. 573-586, 1998.

[6] J. F. Delaigle, C. De Vleeschouwer, and B. Macq, “Watermarking Algorithms based on a Human Visual Model”, *Signal Processing*, Vol. 66, pp. 319-336, 1998.

[7] J. Fridrich, A. C. Baldoza, and R. J. Simard, “Robust Digital Watermarking Based on Key-Dependent Basis Functions”, *Second Inter. Workshop on Information Hiding, USA*, pp. 143-157, 1998.

[8] J. Fridrich, “Applications of Data Hiding In Digital Images”, *Tutorial for The ISPACS Conference*, 1998.

[9] F. Hartung and B. Girod, “Watermarking of uncompressed and compressed Video”, *Signal Processing*, Vol. 66, pp. 283-302, 1998.

[10] F. Hartung, J. K. Su, and B. Girod, “Spread Spectrum Watermarking: Malicious Attacks and Counterattacks”, *Proc. SPIE: Security and Watermarking of Multimedia Contents*, Vol. 3657, 1999.

[11] C. T. Hsu and J. L. Wu, “Multiresolution Watermarking for Digital Images”, *IEEE Trans. CAS II: Analog and Digital Signal Processing*, Vol. 45, pp. 1097-1101, 1998.

[12] C. T. Hsu and J. L. Wu, “Hidden Digital Watermarks in Images”, *IEEE Trans. Image Processing*, Vol. 8, pp. 58-68, 1999.

[13] E. Koch and J. Zhao, “Toward Robust and Hidden Image Copyright Labeling”, *Proc. Nonlinear Signal and Image Processing Workshop*, Greece, 1995.

[14] M. Kutter, F. Jordan, and F. Bossen, “Digital Signature of Color Images using Amplitude Modulation”, *Journal of Electronic Imaging*, Vol. 7, pp. 326-332, 1998.

[15] S. H. Low and N. F. Maxemchuk, “Performance Comparison pf Two Text Marking Methods”, *IEEE Journal on Selected Areas in Communications*, Vol. 16, pp. 561-572, 1998.

[16] C. S. Lu, S. K. Huang, C. J. Sze, and H. Y. Mark Liao, "Robust Image Watermarking Based on Human Perceptual Model", *submitted to Inter. Conf. Image Analysis and Processing*, 1999.

[17] C. S. Lu, S. K. Huang, C. J. Sze, and H. Y. Mark Liao, "Complementary Watermarks Hiding Using Attack-Adaptive Modulation", *manuscript in preparation*, 1999.

[18] B. M. Macq and J. J. Quisquater, "Cryptology for Digital TV Broadcasting", *Proceedings of the IEEE*, Vol. 83, pp. 944-957, 1995.

[19] H. A. Peterson, "DCT basis Function Visibility Threshold in RGB Space ", *SID Inter. Symposium Digest of Technical Papers, Society of Information Display, CA*, pp. 677-680, 1992.

[20] F. Petitcolas and M. G. Kuhn, "StirMark 2.3 Watermark Robustness Testing Software", *http://www.cl.cam.ac.uk/~fapp2/watermarking/stirmark/*, 1998.

[21] F. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on Copyright Marking Systems", *Second Workshop on Information Hiding*, USA, pp. 218-238, 1998.

[22] F. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information Hiding: A Survey", *to appear in Proc. IEEE special issue on Protection of Multimedia Content*, 1999.

[23] C. I. Podilchuk and W. Zeng, "Image-Adaptive Watermarking Using Visual Models", *IEEE Journal on Selected Areas in Communications*, Vol. 16, pp. 525-539, 1998.

[24] L. Qiao and K. Nahrstedt, "Watermarking Schemes and Protocals for Protecting Rightful Ownership and Customer's Rights", *J. Image Comm. and Image Representation*, Vol. 9, pp. 194-210, 1998.

[25] J. J. K. Ruanaidh and T. Pun, "Rotation, Scale, and Translation Invariant Spread Spectrum Digital Image Watermarking", *Signal Processing*, Vol. 66, pp. 303-318, 1998.

[26] A. Said and W. A. Pearlman, "A New, Fast, and Efficient Image Codec based on Set Partitioning in Hierarchical Trees", *IEEE Trans. Circuit and Systems for Video Technology*, Vol. 6, pp. 243-250, 1996.

[27] S. D. Servetto, C. I. Podilchuk and K. Ramchandran, "Capacity issues in Digital Image Watermarking", *5th IEEE Conf. Image Processing*, 1998.

[28] J. M. Shapiro, "Embedded Image Coding Using Zerotrees of Wavelet Coefficients", *IEEE Trans. Signal Processing*, Vol. 41, pp. 3445-3462, 1993.

[29] M. D. Swanson and A. H. Tewfik, “A Binary Wavelet Decomposition of Binary Images”, *IEEE Trans. Image Processing*, Vol. 5, 1996.

[30] M. D. Swanson, B. Zhu, and A. H. Tewfik, “Multiresolution Scene-Based Video Watermarking Using Perceptual Models”, *IEEE Journal on Selected Areas in Communications*, Vol. 16, pp. 540-550, 1998.

[31] M. D. Swanson, B. Zhu, A. H. Tewfik, and L. Boney, “Robust Audio Watermarking Using Perceptual Masking”, *Signal Processing*, Vol. 66, pp. 337-356, 1998.

[32] M. D. Swanson, M. Kobayashi and A. H. Tewfik, “Multimedia Data-Embedding and Watermarking Technologies”, *Proc. of the IEEE*, Vol. 86, pp. 1064-1087, 1998.

[33] A. Z. Tirkel, C. F. Osborne and T. E. Hall, “Image and Watermark Registration”, *Signal Processing*, Vol. 66, pp. 373-384, 1998.

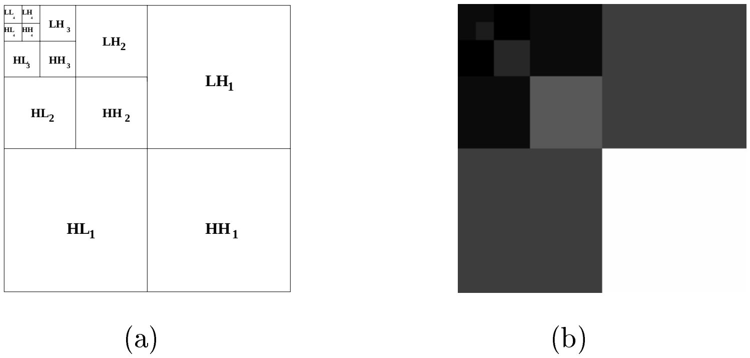
[34] “unZign Watermark Removal Software”, *http://altern.org/watermark/*, 1997.

[35] G. Voyatzis and I. Pitas, “Digital Image Watermarking Using Mixing Systems”, *Computers & Graphics*, Vol. 22, pp. 405-416, 1998.

[36] A. B. Watson, “DCT Quantization Matrics Visually Optimized for Individual Images”, *Proc. SPIE Conf. Human Vision, Visual Processing, and Digital Display IV*, Vol. 1913, pp. 202-216, 1993.

[37] A. B. Watson, G. Y. Yang, J. A. Solomon, and J. Villasenor, “Visibility of Wavelet Quantization Noise”, *IEEE Trans. Image Processing*, Vol. 6, pp. 1164-1175, 1997.

[38] J. Zhao and E. Koch, “A General Digital Watermarking Model”, *Computers & Graphics*, Vol. 22, pp. 397-403, 1998.



(a)

(b)

Figure 1: (a) Four-level wavelet transform and (b) frequency masking map corresponds to four-level wavelet transform [37].

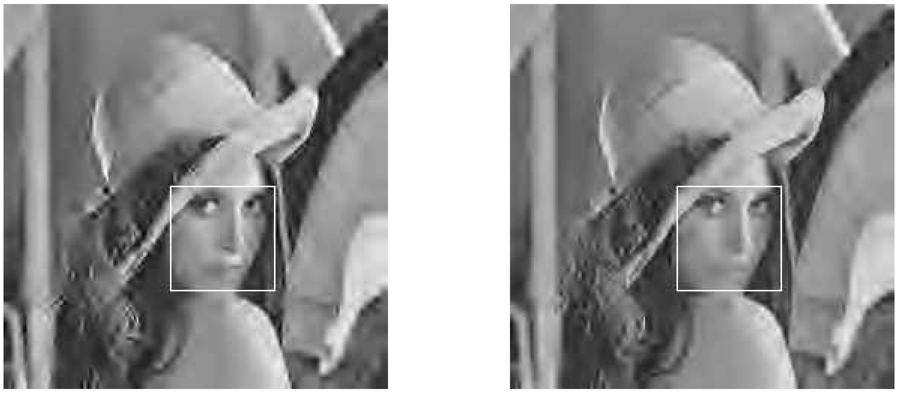


Figure 2: The trade-off between image quality (at a compression ratio of 80 : 1) and robustness: (a) positively modulated image after compression; (b) negatively modulated image after compression.

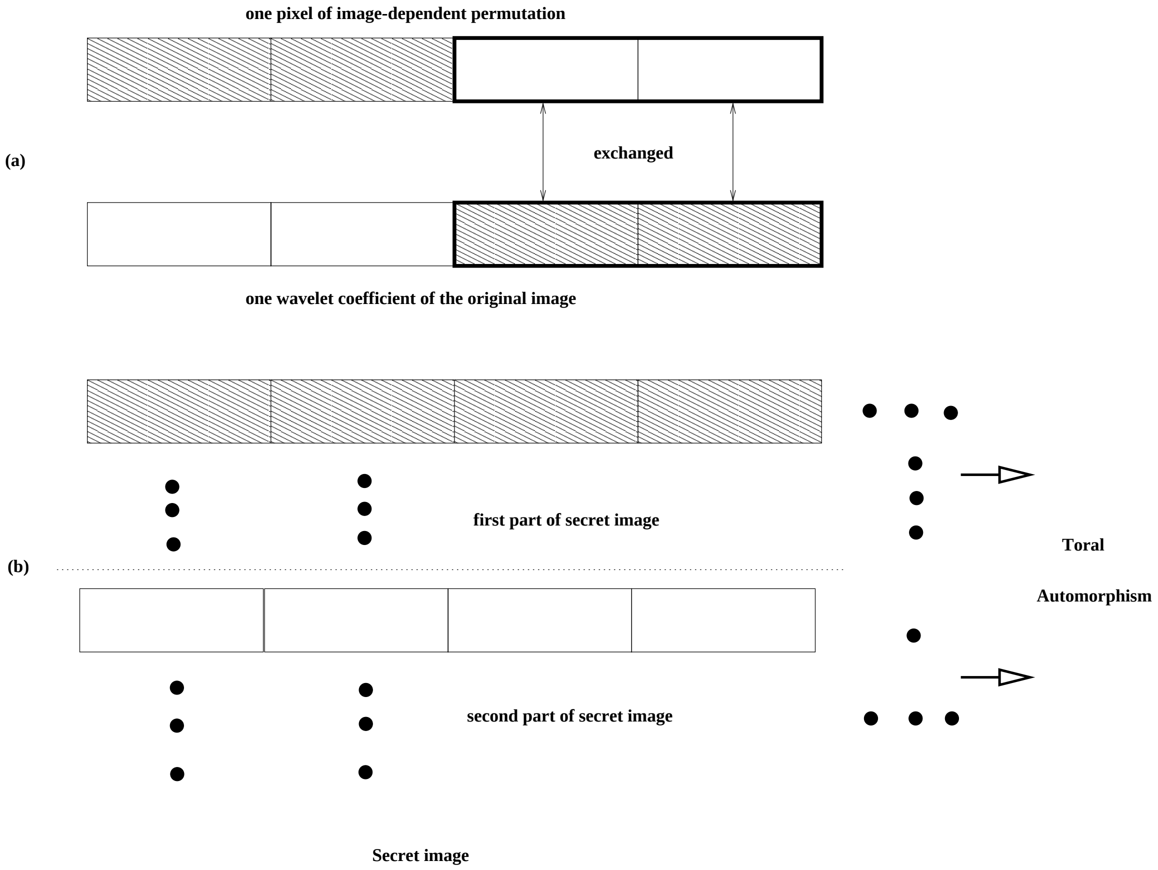

one pixel of image-dependent permutation
(a)
exchanged
one wavelet coefficient of the original image
first part of secret image
(b)
second part of secret image
Toral
Automorphism
Secret image

Figure 3: Secret image: Integrating the mapping image of the image-dependent permutation and the wavelet coefficients of the original image in the watermark embedding process.

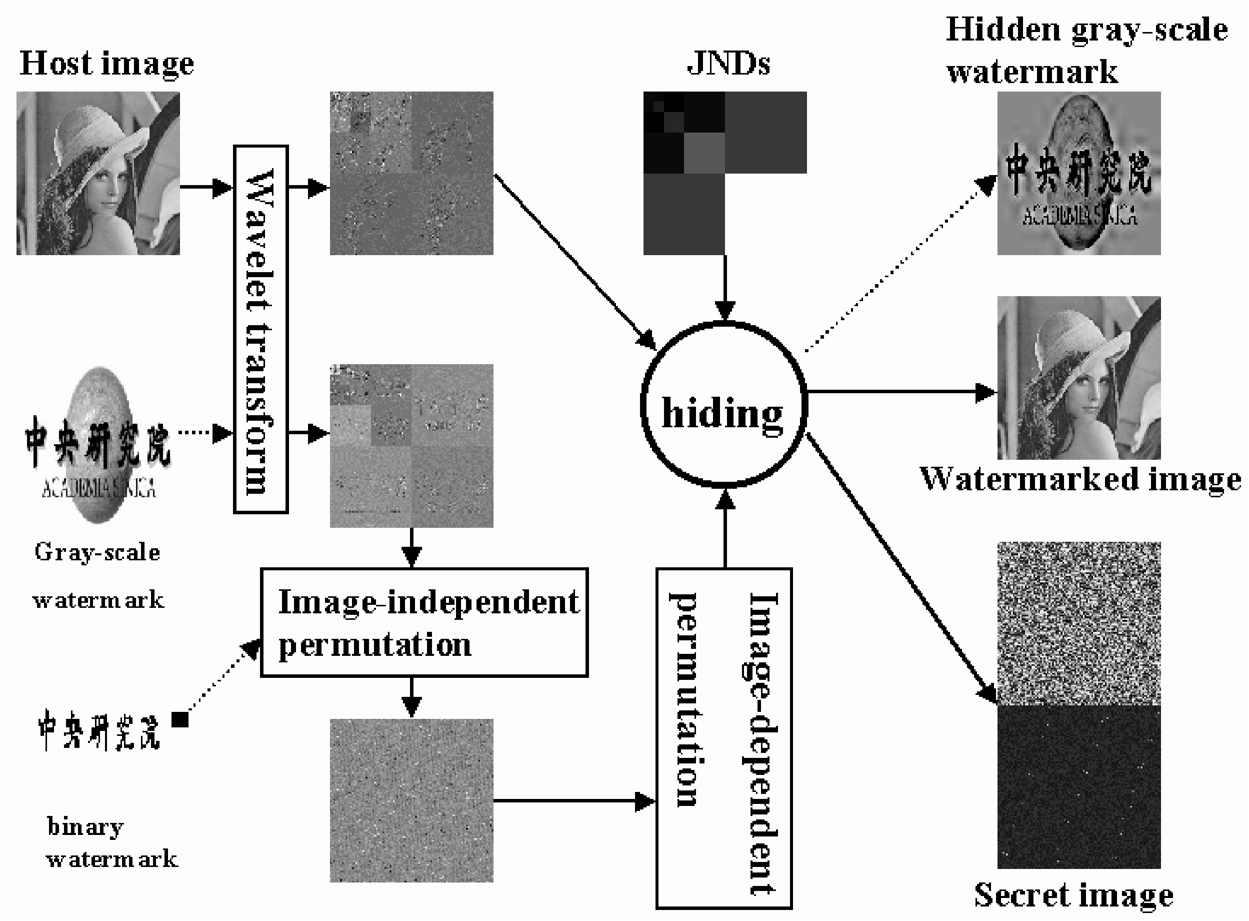


Figure 4: The flow chart of our watermark hiding process.

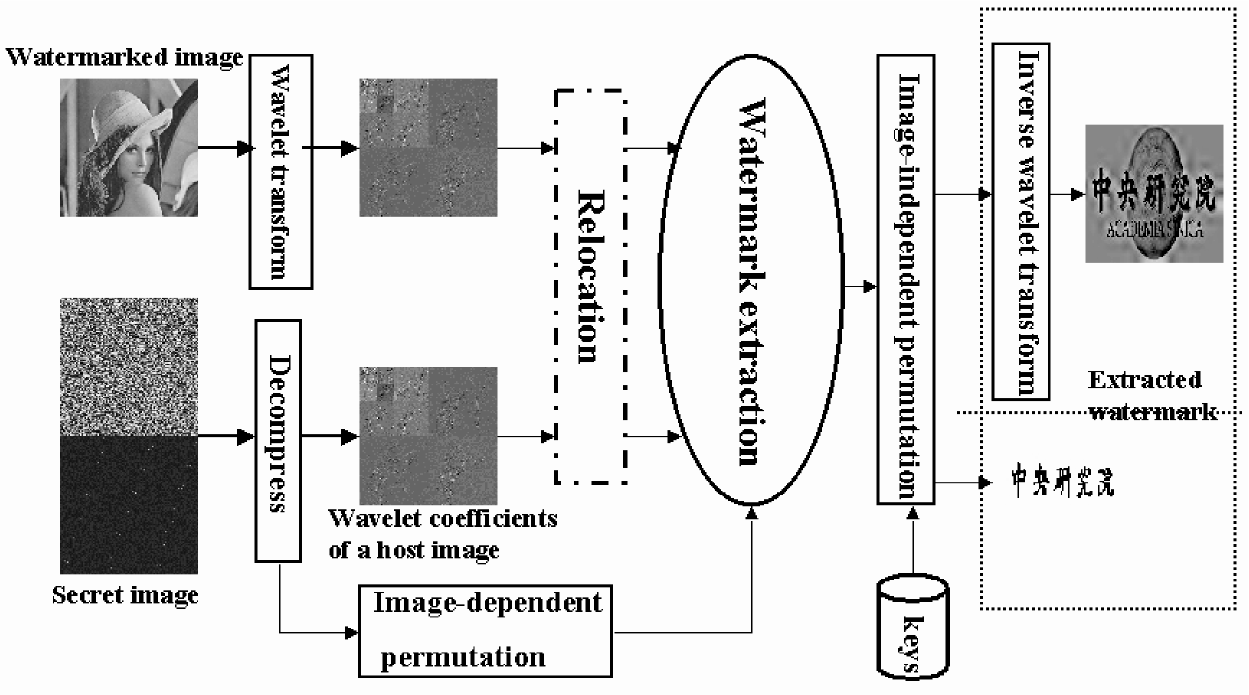
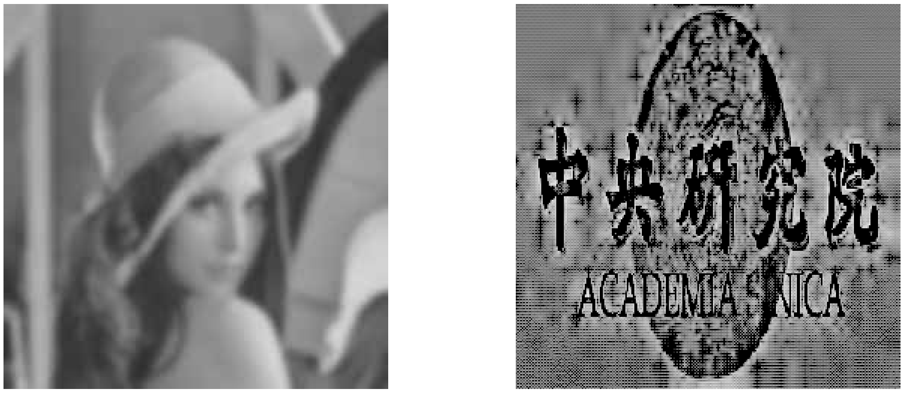
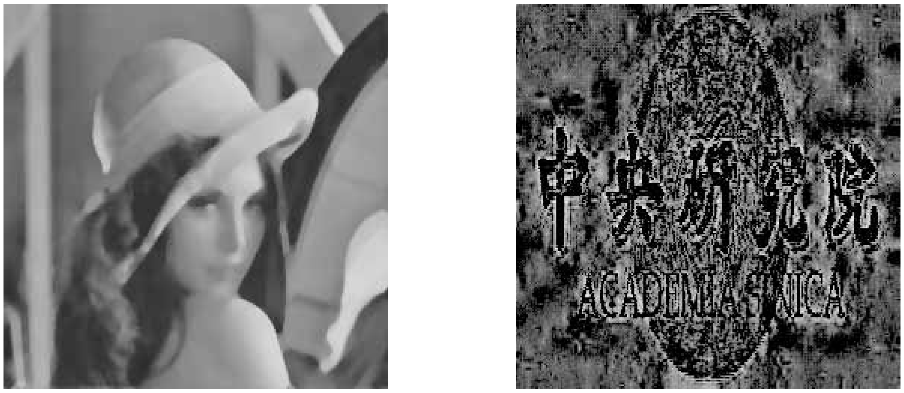


Figure 5: The flow chart of our watermark detection/extraction process.



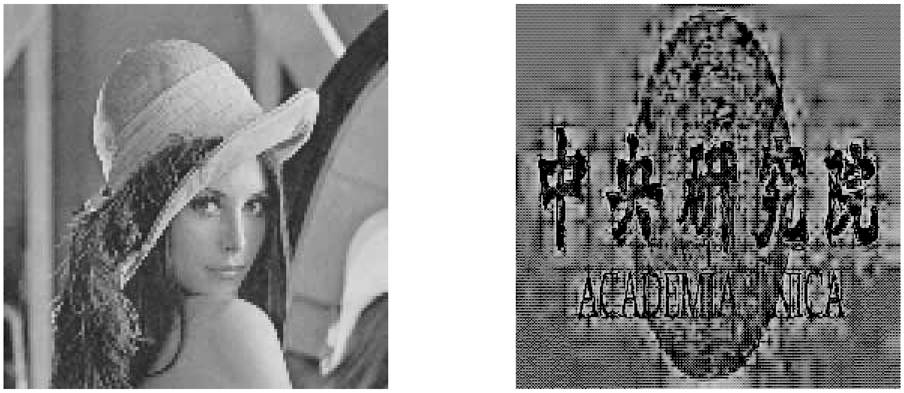
(a) Blurred watermarked image (b) Retrieved watermark

Figure 6: Blurring attack.



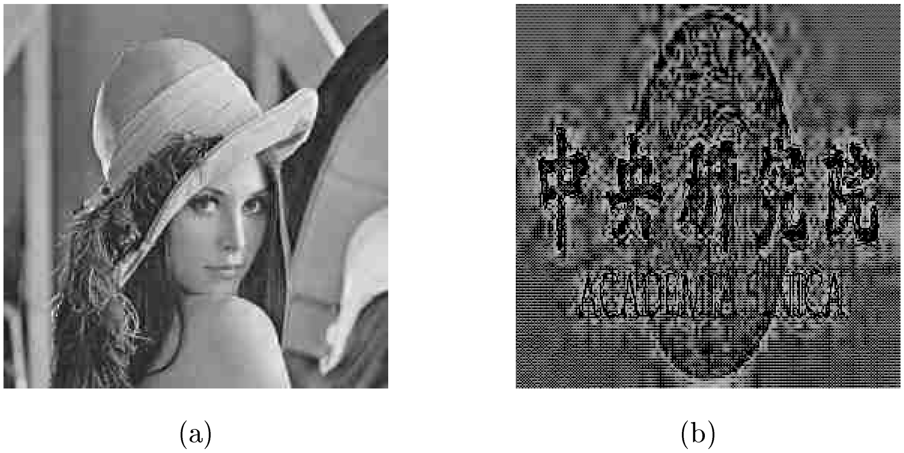
(a) Median filtered watermarked image (b) Retrieved watermark

Figure 7: Median filtering attack.



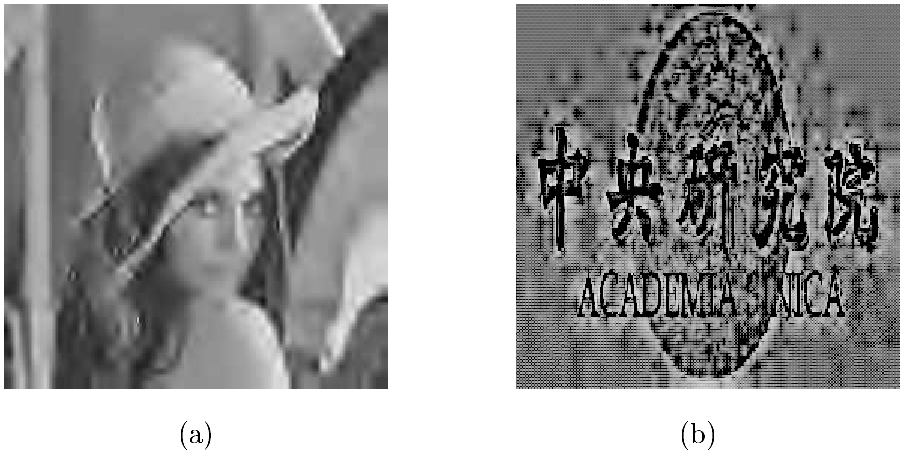
(a) Watermarked image after rescaling (b) Retrieved watermark

Figure 8: Rescaling attack.



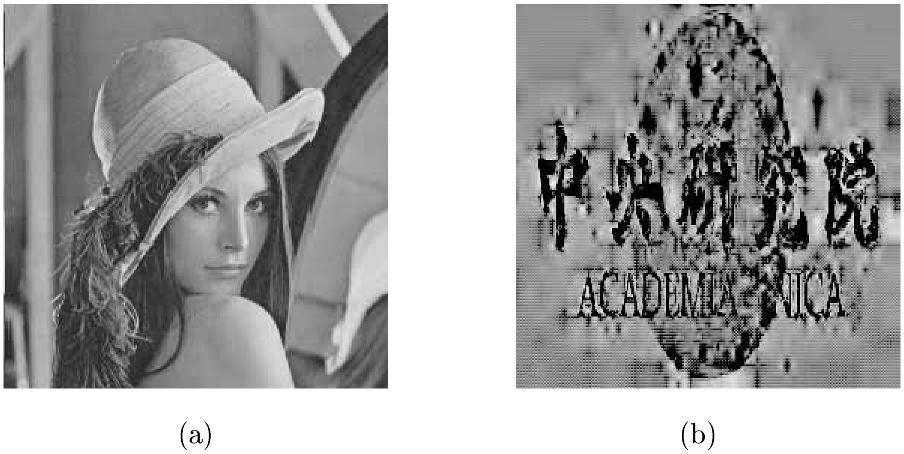
(a) (b)

Figure 9: JPEG attack: (a) watermarked image attacked by JPEG compression with a quality factor of 5% (without smoothing); (b) watermark retrieved from (a).



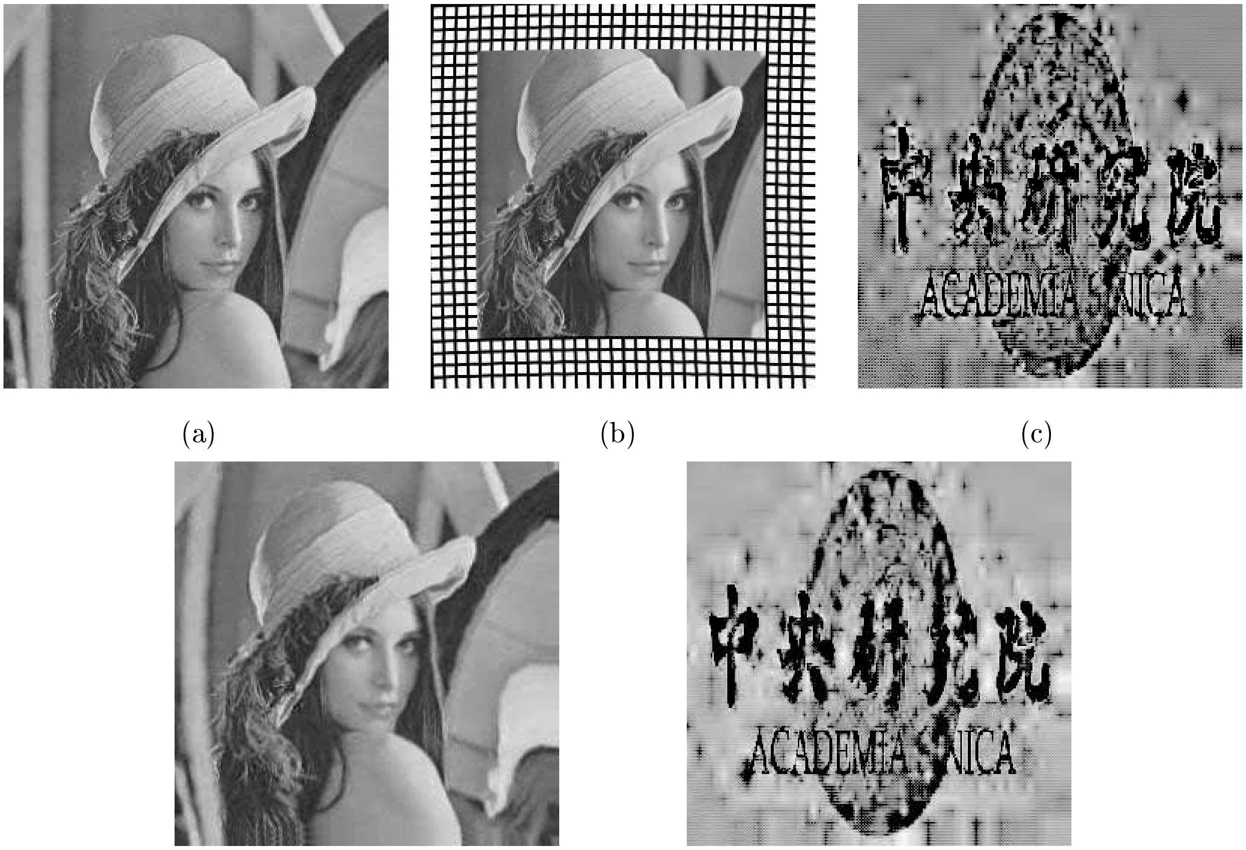
(a) (b)

Figure 10: EZW attack: (a) watermarked image degraded by SPIHT with a compression ratio of 128 : 1 (without smoothing); (b) retrieved watermark.



(a) (b)

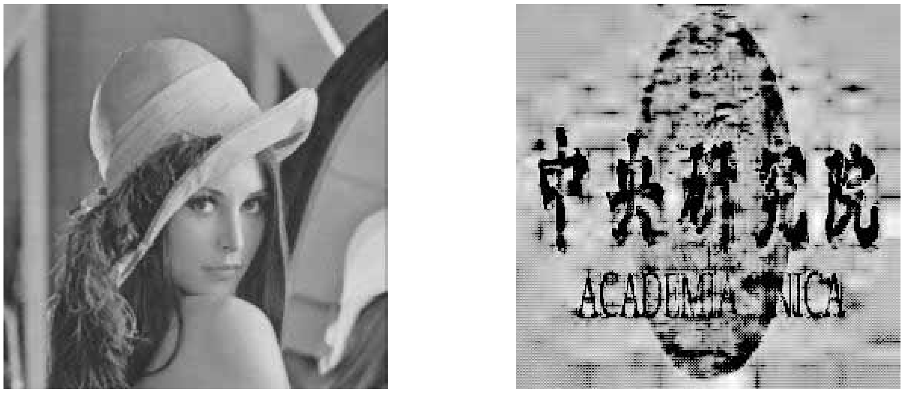
Figure 11: Jitter attack: (4 pairs of columns (90, 200), (50, 150), (120, 110), and (1, 3) were deleted/duplicated): (a) jitter attacked watermarked image; (b) retrieved watermark.



(a) (b) (c)

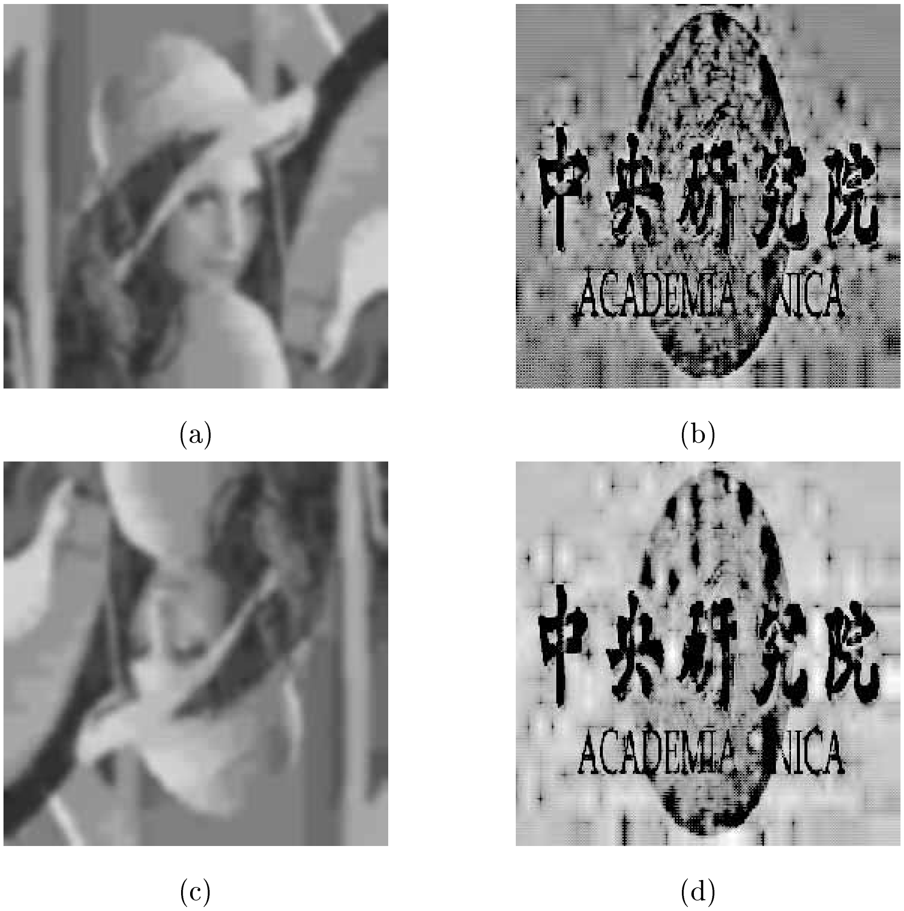
(d) (e)

Figure 12: StirMark attack (all default parameters): (a) StirMark attacked watermarked image (1 time); (b) distorted mesh caused by StirMark attack; (c) watermark retrieved from (a); (d) watermarked image attacked by applying StirMark 5 times; (e) watermark retrieved from (d).



(a) (b)

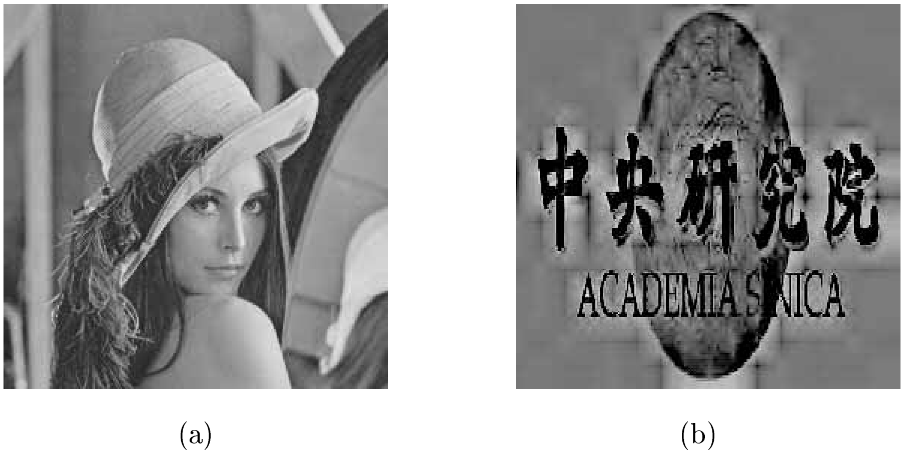
Figure 13: unZign attack: (a) unZign attacked watermarked image; (b) retrieved watermark.



(a) (b)

(c) (d)

Figure 14: Combination attacks (StirMark+JPEG (5%)+blurring(7 × 7)): (a) attacked watermarked image; (b) watermark retrieved from (a); (c) Fig. 14(a) rotated by 180◦; (d) watermark retrieved from (c).



(a) (b)

Figure 15: Collusion attack (five watermarked images were averaged): (a) collusioned attacked image; (b) retrieved watermark.

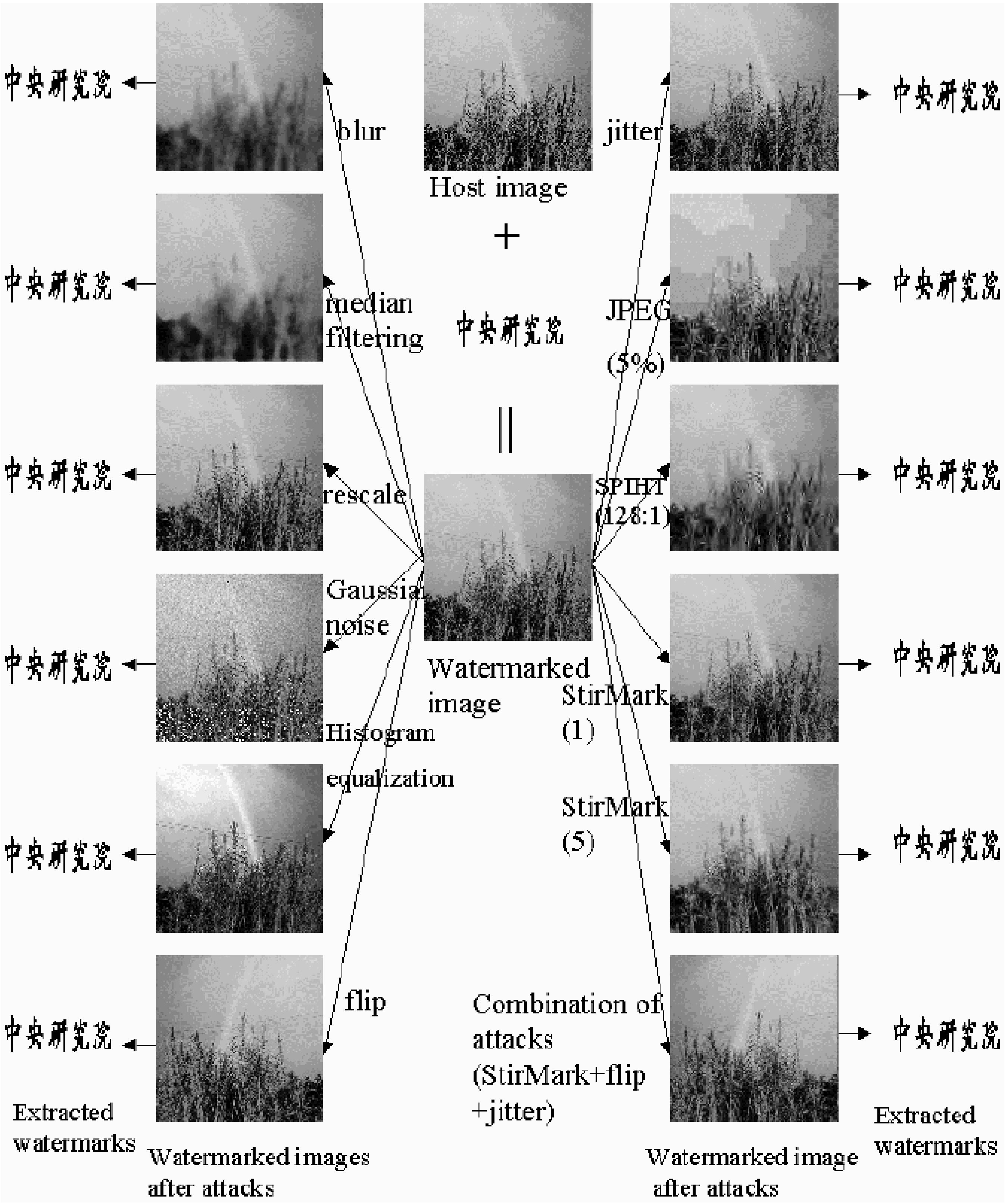


Figure 16: Performance of our binary watermark hiding/detection under various attacks.