A Semi-blind Watermarking Based on Discrete Wavelet Transform

Chin-Chen Chang, Yung-Chen Chou, and Tzu-Chuen Lu

Department of Information Engineering and Computer Science, Feng Chia University, Taichung 40724, Taiwan
ccc@cs.ccu.edu.tw
Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi 62102, Taiwan
jackjow@cs.ccu.edu.tw
Department of Information Management, Chaoyang University of Technology, Taichung41349, Taiwan
tclu@cyut.edu.tw

Abstract. This paper proposed a robust watermarking scheme based on discrete wavelet transform to hide a grayscale watermark in a digital image for image authentication. The proposed scheme employed toral automorphism to scramble the host image and the watermark so as to enhance the security and fidelity of the embedded watermark. Later, the permuted watermark and the permuted host image were transformed by discrete wavelet transform. Next, the transformed watermark was concealed in the low frequency coefficient of the transformed image by using the concept of codebook matching. Simulation results showed that the required extra storage of the proposed scheme for extracting the watermark was lower than that of Lu et al.'s scheme. In addition, the extracted watermark image quality of the proposed methods was better than that of Shieh et al.'s scheme. According to the experimental results, the proposed scheme indeed outperformed Shieh et al.'s and Lu et al.'s schemes. Moreover, the proposed scheme was robust to various attacks, such as JPEG compression, Gaussian blurred, sharpening, cropping, brightness, contrast enhancement, rotation, and so on.

Keywords: Digital Watermark, discrete wavelet transformation, semi-blind watermarking, toral automorphism.

1 Introduction

With the recent growth of the information techniques, digital images are easy to create, edit, adjust, and share. The digital image can be accurately copied and arbitrarily distributed via the Internet, Intranet or other types of networks within seconds. However, these convenient techniques also bring forth several challenging problems that need to be resolved, such as illegal copying, non-authenticated invasion or tampering. For this reason, many image protection mechanisms such

S. Qing, H. Imai, and G. Wang (Eds.): ICICS 2007, LNCS 4861, pp. 164–176, 2007.

as cryptography, watermarking, or data hiding have been proposed to establish the authenticity and integrity of a digital image. Watermarking is one of the popular digital image protection mechanisms that have been widely used in various applications such as intellectual property right, copyright protection, forgery detection, authorship inference, content identification, or image authentication and so on. In a watermarking scheme, a digital signal, called watermark, is embedded in a host image to generate a watermarked image. The watermark is extracted from the watermarked image to prove the ownership of the image when necessary.

Cox et al. [4] classified watermarking techniques as robust watermarking, fragile watermarking [1, 3], and semi-fragile watermarking. In a robust watermarking scheme [5], the watermark is invisibly embedded in the host image. The embedded watermark must be robust enough to resist any regular image processing or malicious attacks [2]. A robust watermarking scheme must satisfy the following requirements: imperceptibility, robustness, unambiguousness, capacity, security, and multiple watermarks. The robust watermarking schemes are used to protect copyright or to verify the ownership.

Different from the robust watermarking, a fragile watermarking scheme concerns the completeness of image content. Any slightest alternation may destroy the embedded watermark. The fragile watermarking schemes are used to ensure the received image is exactly the authorized one, and to verify the image content is selfsame to the original. The semi-fragile watermarking schemes, like fragile watermarking schemes, concern the integrity of the image content. Moreover, the semi-fragile watermarking schemes allow regular image processing such as transmission error, image compression, noise, and so on.

The watermarking schemes can also be divided into three categories: non-blind watermarking scheme, semi-blind watermarking, and blind watermarking. If the original host image is required to reliably extract the embedded watermark, the scheme is non-blind. The practicality of the non-blind watermarking scheme is limited, since it needs extra storage to maintain the source image. Semi-blind watermarking scheme uses the watermark or side information instead of the host image to extract the embedded watermark. In contrast, the blind watermark scheme does not need the host image or extra information.

In this paper, we shall propose a robustness semi-blind watermark scheme for image authentication and copyright protection. The proposed scheme is based on discrete wavelet transformation. In order to increase the security of the watermarked image, the proposed scheme adopts toral automorphism to permute the host image and the digital watermark. Further, the permuted host image and the permuted watermark are transformed by using discrete wavelet transformation. The lower coefficients of the transformed host image are used to train a codebook. The transformed watermark is concealed into the lower coefficients of the transformed host image by using the concept of codebook matching. In order to reliably extract the embedded watermark, the proposed scheme needs some extra information. However, the amount of the required extra information is less than that required by other semi-blind schemes.

The rest of this paper is organized as follows. In Section 2, we briefly review related literatures. Section 3 details the proposed scheme, and Section 4 presents the experimental results. Finally, the conclusions are proposed in Section 5.

2 Literature Review

In the past decade, many semi-blind watermarking techniques have been proposed in various literatures. For example, Voyatzis and Pitas [14] proposed a watermarking scheme in 1996, in which toral automorphisms was applied to scramble the digital watermark. Then, the permuted watermark was inserted into the host image. The toral automorphism was a permutation function that transformed two dimensional data into irregular data. Let us consider an image of size $h \times w$. The value of the coordinates (x, y) of the image is denoted as $P = \begin{bmatrix} x \\ y \end{bmatrix}$. Then, the image is iteratively transformed by toral automorphism t times. Let $P_t = \begin{bmatrix} x_t \\ y_t \end{bmatrix}$ be the value of coordinates (x, y) in t period, where $P_t = \begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ K & K+1 \end{bmatrix} \begin{bmatrix} x_{t-1} \\ y_{t-1} \end{bmatrix} mod \begin{bmatrix} h \\ w \end{bmatrix}$. The image after K iterations, each pixel will be back to its original position.

Lu et al. [9] proposed a cocktail watermarking scheme in 2000. Two complementary watermarks, positively modulated watermark and negatively modulated watermark, were embedded in the wavelet coefficients of the host image. In their scheme, the positions of the watermarks were needed for watermarking extraction. A random mapping function was used to distinguish the positions of the watermarks. Then, the extracted watermark was compared with the original embedded watermark for image authentication. Afterwards, Lu et al. [8] proposed a semi-blind watermarking scheme based on the human visual system for image protection. In their scheme, the host image and a grayscale watermark were transformed by discrete wavelet transform. Next, the host image and the watermark were permuted by using toral automorphism. Then, they used one-to-one mapping function to embed the watermark into larger coefficients of the host image. The mapping function was used to indicate the location of the embedded watermark.

In 2001, Solachidis et al. [11] embedded a circularly symmetric watermark in a host image by using discrete Fourier transformation. In 2001, Lin et al. [6] concealed the watermark in the frequency domain by using Fourier-Mellin transformation. Stankovic et al. [12] embedded a two dimensions watermark with a variable spatial frequency in the host image. The watermark was extracted by using 2-D space/spatial-frequency Radon-Wigner distributions. All these schemes required the original watermark for watermark detection.

In 2005, Shieh et al. [10] proposed a semi-blind watermarking scheme based on singular value decomposition. In their scheme, a grayscale watermark was concealed into a digital image. The first step of their scheme was to divide the watermark and the host image into several blocks. In the second step, each block was transformed by using singular value decomposition (SVD). Next, they found a similar block for each block of the watermark from the host image. The singular value of the block was used to replace that of the similar block of the host image.

However, the image quality of the extracted watermark image of Shieh et al.'s scheme was low. In addition, the computation complexity of their scheme was too heavy, since the scheme used SVD to compute the singular value for each block. In order to solve these problems, this paper will propose a low computation watermarking scheme based on discrete wavelet transformation (DWT). The image quality of the extracted watermark of the proposed scheme is better than that of Shieh et al.'s scheme.

3 Proposed Method

The main idea of our proposed method is to generate the relationship between the host image and watermark to be the right ownership information. Thus, the proposed method uses DWT to transform the host image and watermark from the spatial domain into frequency domain, respectively. After the relationship is constructed, we register the information to a trustworthy third party for further usage. Briefly, this watermarking method can be divided to watermark embedding phase and watermark extraction phase. The details of the proposed method are described as follows.

3.1 Watermark Embedding

Fig. 1 shows the watermark embedding procedure of the proposed scheme. In the figure, the symbol H is a host image and W is a watermark. Both of H and W contain $H \times W$ pixels. The scheme first uses toral automorphism with a secret key [13] to permute H and W into two noise-like images H' and W', respectively. The permutation operation makes the embedded watermark robust for malicious cropping operations.

After that, the scheme applies DWT to transform H' and W' from the spatial domain into frequency domain. DWT decomposes an image into high and low frequency components. The low frequency components compose the base of an image, and the high frequency components refine the outline of the image. The human eye is relatively insensitive to the high-frequency components. Hence, many researchers conceal information in the high-frequency components. However, most perceptual coding techniques, such as JPEG, affect the high-frequency components during image compression. In order to avoid the embedded information from being filtered out, the scheme conceals the information in low-frequency components.

In DWT, each level of decomposition creates four sub-bands of an image, LL, LH, HL, and HH. The LL sub-band can be continually decomposed to obtain another level of decomposition. The scheme performs the DWT twice to obtain two levels of decomposition. The obtained sub-bands are LL_2, HL_2, LH_2, HH_2, HL_1, LH_1, and HH_1. Let H^* and W^* be the transformed images of H' and W', respectively. Next, the scheme consults the sub-band LL_2 of H^* to generate a codebook for embedding the watermark. In the embedding process, the sub-band LL_2 of W^* is divided into several non-overlapping blocks. For each

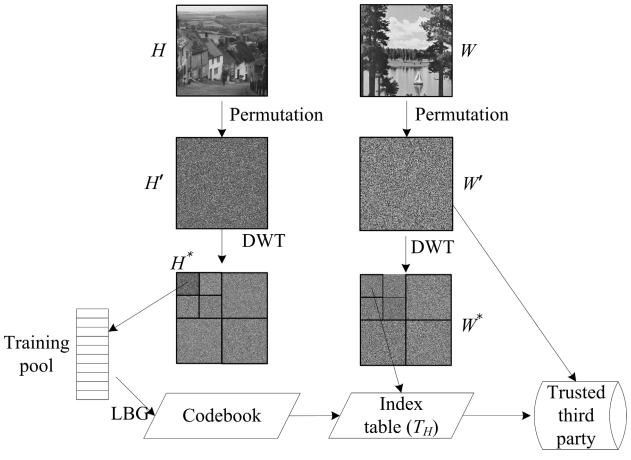


Fig. 1. The diagram of the watermark embedding procedure

block, the scheme searches a similar pattern from the codebook by using the concept of vector quantization. Next, the indices of the patterns are stored for further verification. In the following subsection, we will describe how to generate the codebook as well as how to construct the indices for image authentication.

3.2 Codebook Generation

Before generating a codebook, the scheme uses a normalization function to normalize the coefficients of the sub-band LL_2 of H^* and W^* to ensure that the coefficients are ranging in the same scale. In the other words, all coefficients in LL_2 are ranging from 0 to 255. The normalization function is defined as follows:

$$c' = (c - \min(LL2)) \times r, \tag{1}$$

where c is the coefficient, c' is the normalized coefficient, and r is the normalized ratio computed by

$$r = \frac{255}{\max(LL_2) - \min(LL_2)}. \tag{2}$$

The symbols $\max(LL_2)$ and $\min(LL_2)$ are the maximum function and the minimum function used to find the maximum and minimum values from sub-band LL_2.

Next, the scheme uses a sliding window to move over the normalized LL_2 of H^* one coefficient at a time and generate a set of patterns. The size of the sliding window is $k \times k$, and the set of patterns is called a training pool (denoted as TP). Let $TP = \{tp_i | i = 1, 2, ..., N_{TP}\}$ be the training pool, where tp_i is the i-th pattern and N_{TP} is the number of patterns in TP.

Fig. 2 illustrates an example for constructing a training pool. Fig. 2(a) shows a part of the normalized LL_2 coefficients of an image, and Fig. 2(b) is a diagram

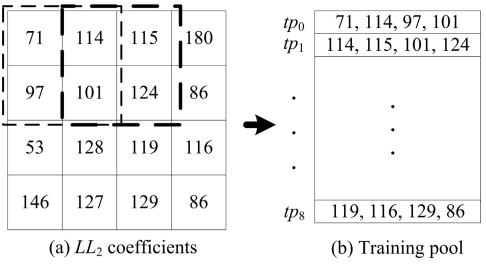


Fig. 2. An example for constructing a training pool

for the training pool. In Fig. 2(a), the sliding window is sized two by two. For instance, first pattern is $tp_0 = \{71, 114, 97, 101\}$, and the second pattern is $tp_1 = \{114, 115, 101, 124\}$.

After the training pool is constructed, we employ the LBG algorithm [7] to train a coefficient codebook. The first step of the LBG algorithm is to randomly choose N_B patterns from the training pool and set them as the initial coefficient codebook. Further, the classification operation classifies the patterns into N_B classes according to the initial coefficient codebook. After the classification, a new coefficient codebook will be constructed by computing the central value of each class. The training of the codebook is terminated when the change between the newly trained codebook and previous iteratively trained codebook is smaller than a pre-defined threshold.

Next, the scheme constructs an indices table to indicate the relationship between H^* and W^*'s LL_2 coefficients. In this stage, the scheme divides the subband LL_2 of W^* into several blocks, and matches a most similar pattern from the codebook for each block. The indices of the most similar pattern are collected to form an indices table. The indices table and the permuted watermark W' are stored for further watermark verification.

3.3 Watermark Extracting

The watermark extracting procedure is used to prove the ownership of an image. Fig. 3 shows the diagram of the watermarking extracting procedure. In the figure, the symbol V denotes a controversial image. The scheme permutes V by using toral automorphism with the secret key, and uses the DWT to transform the permuted image from the spatial domain into frequency domain. The normalized coefficients are used to generate a codebook. Further, the corresponding indices table of V and the permuted watermark W' are retrieved from the trustworthy third party to reconstruct the watermark. The inverse DWT (IDWT) is used to transform the watermark from the frequency domain into spatial domain. Then, the scheme de-permutes the transformed watermark to construct an extracted watermark image.

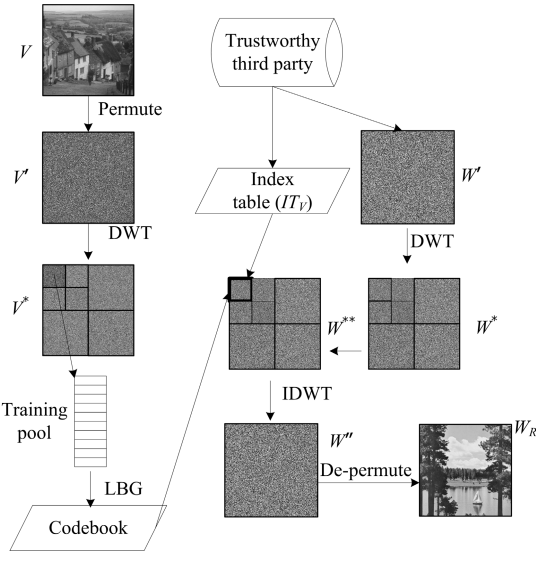


Fig. 3. A diagram of the watermark extracting procedure

4 Experiments and Experimental Results

This section demonstrates the performance of the proposed scheme. In addition, Shieh et al.'s method [10] and Lu et al.'s method [8] are employed to be the benchmarks. Fig. 4 shows six commonly used test images. The size of each image is 256 × 256 pixels. We use the test images to be watermarks and the host images for our experiments.

This paper adopts the peak-signal-to-noise ratio (PSNR) to measure the image quality of the extracted watermark. The PSNR in decibels (dB) is computed by

$$PSNR = 10 \log \frac{255^2}{MSE}(dB), \text{where} \quad MSE = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{i=1}^{W} (I_{ij} - I'_{ij})^2, \quad (3)$$

where MSE is the mean square error between the extracted watermark and the original watermark.

The PSNR is not meaningful, but it is a useful measurement for comparing the differences between the extracted watermark and the original one. The high PSNR value means that the extracted watermark has less distortion from original watermark. On the contrary, low PSNR means that the extracted watermark has more distortion from the original watermark.

4.1 Experimental Results

Tables 1 and 2 show the PSNR of the extracted watermark. The size of the sliding window is a critical factor that influences the performance of image quality.

(a) Baboon (b) Barbara (c) Goldhill

(d) Lena (e) Sailboat (f) Zelda

Fig. 4. The testing images of size 256 × 256

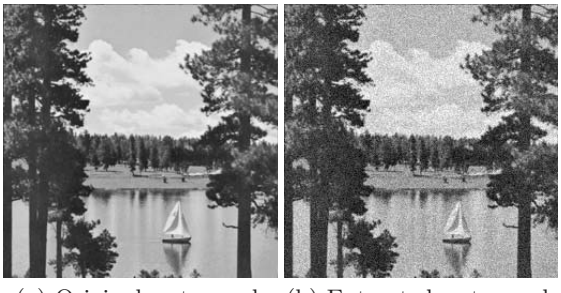
Table 1. The visual quality of the extracted watermark by using the proposed method (size of sliding window = 4 × 4)

	Host images					
Watermarks	Baboon	Barbara	Goldhill	Lena	Sailboat	Zelda
Baboon	-	26.3294	27.4068	27.6417	24.7498	29.6612
Barbara	29.7243	-	27.47	27.6137	24.7539	29.6618
Goldhill	26.158	29.6332	-	27.437	24.637	29.5769
Lena	26.3497	29.7531	27.3282	-	24.7412	29.6427
Sailboat	26.3081	29.7163	27.4058	27.6096	-	29.6241
Zelda	26.3391	29.7603	27.4717	27.6377	24.7578	-

According to the experimental results shown in Table 1 and Table 2, we can see that larger sliding window size has lower visual quality results. For example, the sliding window in Table 1 is sized four by four, and that in Table 2 is two by two. The average PSNR in Table 1 is 27.5633 dB while that in Table 2 is 29.6022 dB. The different sliding window sizes have different benefits. We will discuss the effects of different sliding window size in next sub-section. In this paper, the sliding window is sized four by four to test the performance of the proposed method.

Table 2. The visual quality of the extracted watermark by using the proposed method (size of sliding window = 2 × 2)

Watermarks	Host images					
	Baboon	Barbara	Goldhill	Lena	Sailboat	Zelda
Baboon	-	28.3201	29.5233	29.7928	26.6956	31.5927
Barbara	31.7595	-	29.5151	29.7909	26.6945	31.594
Goldhill	31.7169	28.2465	-	29.7161	26.6629	31.5657
Lena	31.7539	28.3174	29.4524	-	26.6894	31.581
Sailboat	31.7612	28.3178	29.5197	29.791	-	31.5928
Zelda	31.7696	28.319	29.5265	29.7922	26.696	-



(a) Original watermark (b) Extracted watermark

Fig. 5. The original watermark and the extracted watermark

In the experimental results, the worst case is to embed the watermark "Sailboat" into the host image "Goldhill". In Table 1, the PSNR value of the extracted watermark is 24.637 dB. The original watermark is shown in Fig. 5(a) and the extracted watermark is shown in Fig. 5(b). Obviously, Fig. 5(b) is meaningful and recognizable.

For the robustness evaluation, we apply the lossy image compression and commonly used image processing to evaluate the integrity and recognizability of the extracted watermark. In this experiment, JPEG and JPEG2000 are used to compress the host image "Goldhill" with compression ratios of 41:1 and 62:1, respectively. The extracted watermarks are shown in Figs. 7(a) and Fig. 7 (b).

Figs. 6(c)-(i) are the modified images which use the Gaussian blurring with 5 as the radius, sharpening, Gaussian noise with 10%, cropping, brightness adjustment with 50%, contrast enhancement with 50%, and rotating the image with degree = 40°. The extracted watermarks corresponding to the modified images are shown in Figs. 7(c)-(i). All the extracted watermarks are meaningful and recognizable. Even though the watermarked image had been cropped into a quarter of original image, the PSNR value of the extracted watermark is 20.6521 dB.

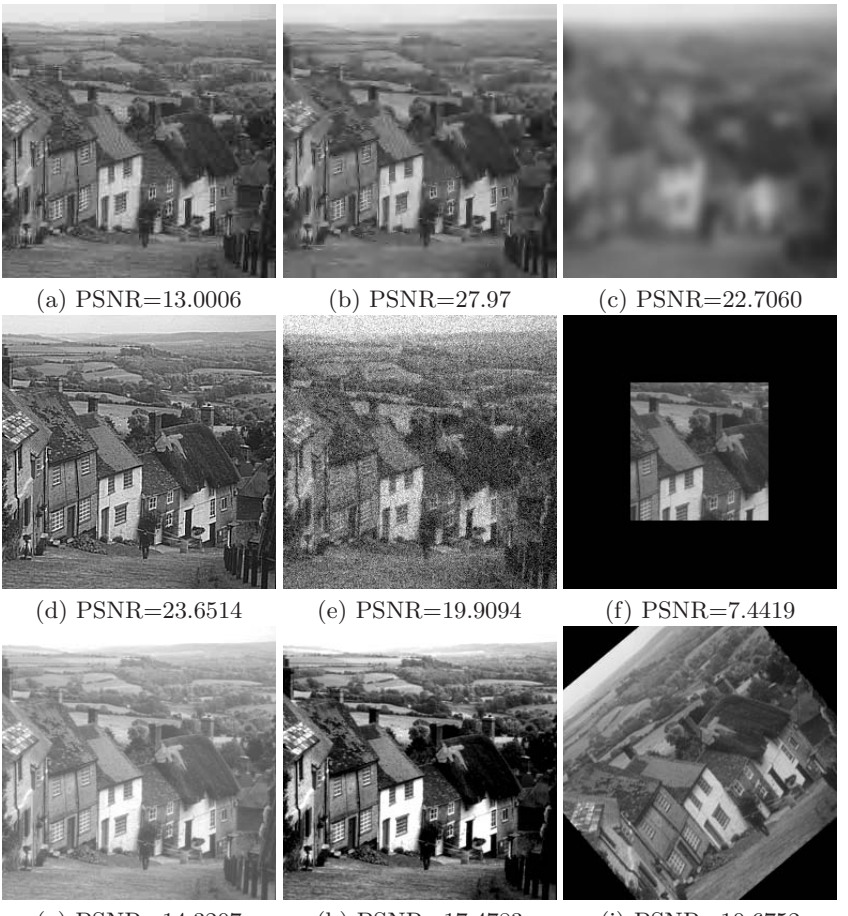
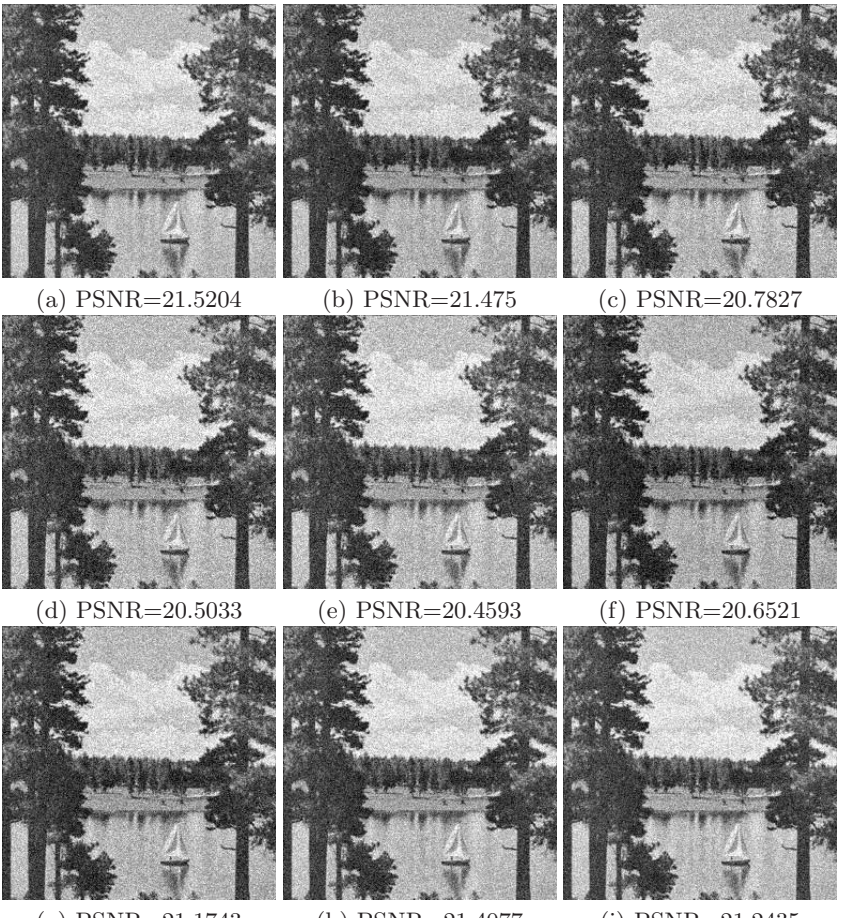


Fig. 6. The attacked images; (a) JPEG compression, (b) JPEG2000 compression, (c) Gaussian blurred (radius=5), (d) Sharpening, (e) Gaussian noise (10%), (f) Cropping, (g) Brightness, (h) Contrast enhancement, (i) Rotation

Table 3 shows the comparing results of the extracted watermark between the proposed method and Shieh et al.'s method. Generally speaking, the visual quality of the watermark extracted by using the proposed method is higher than that by Shieh et al.'s method. Lu et al. proposed a semi-blind watermarking method in 2001 [8]. Lu et al.'s method transforms both host image and watermark into the frequency domain by DWT. They applied the just noticeable distortion (JND) to decide how watermark's coefficients are embedded to host image's coefficients. The comparisons are summarized in Table 4.



(a) PSNR=21.5204 (b) PSNR=21.475 (c) PSNR=20.7827

(d) PSNR=20.5033 (e) PSNR=20.4593 (f) PSNR=20.6521

(g) PSNR=21.1743 (h) PSNR=21.4077 (i) PSNR=21.2435

Fig. 7. Extracted watermark from modified images; (a) JPEG compression, (b) JPEG2000 compression, (c) Gaussian blurred (radius=5), (d) Sharpening, (e) Gaussian noise (10%), (f) Cropping, (g) Brightness, (h) Contrast enhancement, (i) Rotation

Table 3. The results (PSNR) for comparing to Shieh et al.'s method (watermark = "Sailboat")

Methods	Host images				
	Baboon	Barbara	Goldhill	Lena	Zelda
Proposed method	29.7163	26.3081	27.4058	27.6096	29.6241
Shieh et al.'s method	19.3061	28.3829	22.5033	23.7203	15.4935

Table 4. The comparison of the proposed method and existing method

Items	Lu et al. [5]	Shieh et al. [7]	Proposed
Original image size (byte)	256 × 256	256 × 256	256 × 256
Watermark size (byte)	256 × 256	256 × 256	256 × 256
Required extra storage (byte)	524,288	65,792	65,792
Retrieval strategy	Semi-blind	Semi-blind	Semi-blind
Domain (host/watermark)	DWT/DWT	SVD/SVD	DWT/DWT
Multiple watermarking	No	Yes	Yes

4.2 Discussions

The size of the sliding window is corresponding to that of a pattern used to train a codebook. Different size of pattern will affect the visual quality of the extracted watermark and the extra storage of extracting information. The pattern with large size leads worse visual quality of extracted watermark and little storage needed for storing the extra information. On the contrary, the pattern with small size can obtain better visual quality of the extracted watermark. However, it requires more storage to keep the extra information. This is a trade-off problem.

In the proposed method, two-level DWT transform was performed to embed and extract watermarks. In level one DWT transform, the number of lower frequency coefficients is a quarter of the number of pixels of the host image so that it will result in a large number of patterns in the training pool. A large training pool affects the computation cost of training a suitable codebook. On the other hand, more than two levels of DWT transform can not provide enough patterns for constructing the training pool. Thus, based on our experimental experiences, it is suggested that the two-level DWT transform be suitable for the proposed method. However, the number of transformation level is not fixed because it corresponds to the size of images. In other words, an image can be transformed by more than two levels when the particular coefficients of the transformed image produce enough training patterns.

A user may bring up an un-registered image and apply the watermark retrieving procedure to exploit an image that belongs to a certain company. In our proposed watermark retrieving procedure, a verifier will request the corresponding secret watermark W^* and index table data from the trusted unit. Thus, only the righteous owner of the image can ask to verify the watermark.

5 Conclusions

In this paper, we have demonstrated a semi-blind watermark technology based on discrete wavelet transformation. In the embedding process, the host image and the watermark were permuted by toral automorphism to increase the security, and fidelity of the embedded watermark and to resist the counterfeiting

attack. The proposed scheme embedded the watermark in the frequency domain that could provide a greater control in terms of the robustness and fragility of the watermark. The benefits have been demonstrated in our experiments which indicated that our proposed scheme outperformed Lu et al.'s and Shieh et al.'s schemes in terms of the quality of extracted watermarks and the amount of the required storage.

References

1. Chang, C.C., Hu, Y.S., Lu, T.C.: A Watermarking-Based Image Ownership and Tampering Authentication Scheme. Pattern Recognition Letters 27, 439–446 (2006)
2. Chang, C.C., Tsai, P.Y., Lin, M.H.: SVD-based Digital Image Watermarking Scheme. Pattern Recognition Letters 26(10), 1577–1586 (2005)
3. Chang, C.C., Wu, W.C., Hu, Y.C.: Public-Key Inter-Block Dependence Fragile Watermarking for Image Authentication Using Continued Fraction. Informatica 28, 147–152 (2004)
4. Cox, I.J., Kilian, J., Leighton, F.T., Shamoon, T.: Secure Spread Spectrum Watermarking for Multimedia, IEEE Transactions on Image Processing, Vol. IEEE Transactions on Image Processing 6(12), 1673–1687 (1997)
5. Licks, V., Jordan, R.: On Digital Image Watermarking Robust to Geometric Transformation. In: Proceedings of IEEE International Conference on Image Processing, vol. 3, pp. 690–693 (2000)
6. 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 Transactions on Image Processing 10, 767–782 (2001)
7. Linde, Y., Buzo, A., Gray, R.M.: An Algorithm for Vector Quantizer Design. IEEE Transactions on Communications 28, 84–95 (1980)
8. Lu, C.S., Huang, S.K., Sze, C.J., Liao, H.Y.: A New Watermarking Technique for Multimedia Protection, Multimedia Image and Video Processing, Chap. 18, pp. 507–530. CRC Press, Boca Raton, USA (2001)
9. Lu, C.S., Huang, S.K., Sze, C.J., Liao, H.Y.: Cocktail Watermarking for Digital Image Protection. IEEE Transactions on Multimedia 2(4), 209–224 (2000)
10. Shieh, J.M., Lou, D.C., Chang, M.C.: A Semi-blind Digital Watermarking Scheme Based on Singular Value Decomposition. Computer Standards & Interfaces 28(4), 428–440 (2006)
11. Solachidis, V., Pitas, I.: Circularly Symmetric Watermark Embedding in 2-D DFT Domain. IEEE Transactions on Image Processing 10, 1741–1753 (2001)
12. Stankovic, S., Djurovic, I., Pitas, I.: Watermarking in the Space/Spatial-Frequency Domain Using Two-Dimensional Radon-Wigner Distribution. IEEE Transactions on Image Processing 10, 650–658 (2001)
13. Voyatzis, G., Pitas, I.: Digital Image Watermarking Using Mixing Systems. Computers & Graphics 22(4), 405–416 (1998)
14. Voyatzis, G., Pitas, I.: Applications of Toral Automorphisms in Image Watermarking. In: IEEE Conference on Image Processing, pp. 237–240 (1996)