A DWT-DFT Composite Watermarking Scheme Robust to Both Affine Transform and JPEG Compression

Xiangui Kang, Jiwu Huang, *Senior Member, IEEE*, Yun Q. Shi, *Senior Member, IEEE*, and Yan Lin

***Abstract*—Robustness is one of the crucial important issues in watermarking. Robustness against geometric distortion and JPEG compression at the same time with blind extraction remains especially challenging. In this paper, a blind discrete wavelet transform-discrete Fourier transform (DWT-DFT) composite image watermarking algorithm that is robust against both affine transformation and JPEG compression is proposed. This algorithm improves the robustness via using new embedding strategy, watermark structure, 2-D interleaving, and synchronization technique. A spread-spectrum-based informative watermark with a training sequence are embedded in the coefficients of the LL subband in the DWT domain while a template is embedded in the middle frequency components in the DFT domain. In watermark extraction, we first detect the template in a possibly corrupted watermarked image to obtain the parameters of affine transform and convert the image back to its original shape. Then we perform translation registration by using the training sequence embedded in the DWT domain and finally extract the informative watermark. Experimental works have demonstrated that the watermark generated by the proposed algorithm is more robust than other watermarking algorithms reported in the literature. Specifically it is robust against almost all affine transform related testing functions in StirMark 3.1 and JPEG compression with quality factor as low as 10 simultaneously. While the approach is presented for gray-level images, it can also be applied to color images and video sequences.**

***Index Terms*—Affine transformation, geometric attacks, image watermarking, robustness, template matching.**

I. Introduction

Digital watermarking has emerged as a potentially effective tool for multimedia copyright protection, authentication and tamper proofing [1]. Robustness of watermarking is one of the key issues for some applications. A serious problem constraining some practical exploitations of watermarking technology is the insufficient robustness of existing watermarking algorithms against geometrical distortions such as translation, rotation, scaling, cropping, change of aspect ratio and shearing. These geometrical distortions cause the loss of geometric synchronization that is necessary in watermark detection and decoding [2]. Vulnerable to geometric distortion is a major weakness of many watermarking methods [3].

Although some significant progresses have been made recently, these new schemes are normally not robust to JPEG compression at the same time as shown below. There are two different types of solutions to resisting geometrical attacks: nonblind and blind methods [4]. With the nonblind approach, due to availability of the original image, the problem can be resolved with a good solution by effective search between the geometrically attacked and unattacked image [5], [6]. The blind solution, which does not use the original image in watermark extraction, has wider application but is obviously more challenging. Three major approaches to the blind solution have been reported in the literature. The first approach hides a watermark signal in the invariants of a host signal (invariant with respect to scaling, rotation, shifting and etc.), and is somewhat awkward due to the theoretical and practical difficulties in constructing invariants with respect to combinations of the above-mentioned operations. Specifically, Ruanaidh *et al.* [7] first proposed a watermarking scheme based on transform invariants via applying Fourier-Mellin transform to the magnitude spectrum of an original image. However, the resulting stego-image quality is poor due to interpolation errors [2]. In [3], instead of a "strong invariant" domain, the watermark is embedded into the magnitudes of the DFT coefficients resampled by the LPM. The detection process involves a comparison of the watermark with all cyclic shifts of the extracted watermark to cope with rotation. To deal with scaling, the correlation coefficient is selected as the detection metric. Only one bit information is hidden in the image, and the watermark cannot resist the general transformations.

The second approach exploits the self-reference principle based on an auto-correlation function (ACF) or the Fourier magnitude spectrum of a periodical watermark [8], [9]. In [8], the watermark is replicated in the image in order to create four repetitions of the same watermark such that the experienced geometrical transformation can be detected by applying autocorrelation to the investigated image. Experimental results showed that the algorithm could resist generalized geometrical transformations. However, the watermarking is vulnerable to the lossy coding scheme such as JPEG compression. In [9], the usage of a periodical block allocation of a watermark pattern for recovering from geometrical distortions is proposed. It is

Manuscript received December 16, 2002; revised March 20, 2003. This work was supported by the National Science Foundation (NSF) of China (69975011, 60172067, 60133020), the "863" Program (2002AA144060), the NSFGD (013164), Funding of China National Education Ministry, the New Jersey Commission of Science and Technology via NJWINS, the New Jersey Commission of High Education via NJ-ITOWER, and the NSF via IUCRC.

X. Kang, J. Huang, and Y. Lin are with the Department of Electronics and Communication Engineering, Sun Yat-Sen University, Guangzhou 510275, China (e-mail: isskxg@zsu.edu.cn; isshjw@zsu.edu.cn; ly.king@263.net).

Y. Q. Shi is with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102 USA (e-mail: shi@njit.edu).

Digital Object Identifier 10.1109/TCSVT.2003.815957

noticed that, however, the watermark estimation is key-independent, the periodical watermark results in an underlying regular grid. Hence, the watermark may be detected and may be destroyed [10], [11]. It has been proposed recently that the resulting watermark can be slightly pre-distorted in such a way that this problem may be resolved [12]. In [12], it is reported that the watermark is robust against local or nonlinear geometrical distortions, such as random bending attack.

The third approach utilizes an additional template (e.g. a sinusoid or a cross [13], [14]). In [15], Pereira and Pun proposed an affine resistant image watermark in DFT domain. The informative watermark (the message to be conveyed to the detector/receiver) together with a template is embedded in the middle frequency components in DFT domain. The template consists of intentionally embedded peaks in the Fourier spectrum. That is, the watermark embedder casts extra peaks in some locations of the Fourier spectrum. The locations of the peaks are predefined or selected by a secret key. The strength is adaptively determined by using local statistics of the Fourier spectrum. In watermark detection, all local maxima are extracted by using small windows. While resistant to affine transform, the watermark generated with the scheme is not robust enough. In particular, it is not robust to JPEG compression with the quality factor below 75.

In this paper, a new blind image watermarking algorithm robust against both affine transformations and JPEG compression is proposed. The proposed DWT-DFT composite watermarking scheme embeds a message and a training sequence in LL_4 subband in the DWT domain, and embeds a template in magnitude spectrum in DFT domain. The watermarking incorporates DWT/DFT, concatenated coding of direct sequence spread spectrum (DSSS) and BCH, 2-D interleaving, resynchronization based on the template and training sequence. Experimental results have demonstrated that the watermark is robust against both affine transformations and JPEG compression when the quality factor is as low as 10.

The rest of this paper is organized as follows. In Section II, we introduce the watermark embedding. Section III describes the watermark extraction based on resynchronization. The experimental results are presented in Section IV. A conclusion is drawn in the last section.

II. Watermark Embedding

This algorithm achieves enhanced robustness via improving embedding strategy and watermark structure, and using a new effective synchronization technique. DWT is playing an increasingly important role in watermarking, due to its good spatial-frequency characteristics and its wide applications in the image/video coding standards. According to Cox *et al.* [16] and Huang *et al.* [17], watermark should be embedded in the DC and low frequency AC coefficients in DCT domain due to their large perceptual capacity. The strategy can be extended to DWT domain. We embed informative watermark into LL_4 subband in the DWT domain to make it more robust while keeping the watermark invisible. When the marked image undergoes affine transformation $\left(\begin{bmatrix}x'\\y'\end{bmatrix} = \boldsymbol{B}\begin{bmatrix}x\\y\end{bmatrix} + \begin{bmatrix}t_x\\t_y\end{bmatrix}\right)$,, the matrix $\boldsymbol{B}$ can be determined by using a template as reference. The template is embedded into the middle frequency components in the magnitude spectrum to avoid interfering with the informative watermark. To determine the translation parameter, we embed a training sequence in the DWT domain. To survive all kinds of attacks, we use the concatenated coding of BCH and DSSS method to encode the message $\boldsymbol{m}\ \{m_i;\ i = 1,\ldots,L, m_i \in \{0,1\}\}(L = 60$ in our work). To cope with bursts of errors which possibly occurred with watermark, a newly developed 2-D interleaving [18], [19] is exploited. The watermark embedding process is shown in Fig. 1.

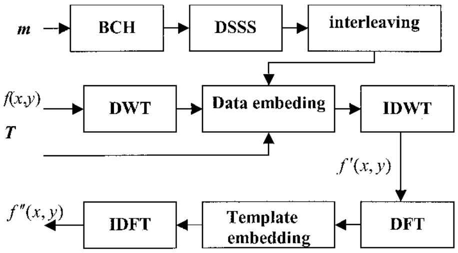


Fig. 1. Watermark embedding process.

A. Message Encoding and Training Sequence Embedding in the DWT Domain

The watermark embedding in the DWT domain is implemented through the following procedures.

1) By using the Daubechies 9/7 bi-orthogonal wavelet filters, we apply a four-level DWT to an input image $f(x,y)$ ($512 \times 512 \times 8$ bits, in our work), generating 12 subbands of high frequency (LH_i, HL_i, HH_i, $i = 1 \sim 4$) and one low-frequency subband (LL_4).
2) The message $\boldsymbol{m}$ is first encoded using BCH (72, 60) to obtain the message $\boldsymbol{m}_c$ of length $L_c = 72$. Then each bit m_{ci} of $\boldsymbol{m}_c$ is DSSS encoded using an N_1-bit bipolar PN-sequence $\boldsymbol{p} = \{p_j;\ j = 1,\ldots,N_1\}$, where "1" is coded spreadly as $\{+1 \times p_j;\ j = 1,\ \ldots,N_1\}$, "0" as $\{-1 \times p_j;\ j = 1,\ldots,N_1\}$, thus obtaining a binary string $\boldsymbol{W}$

$$m_{ci} \xrightarrow{\text{DSSS coding}} W_i\{w_{ij};\ w_{ij} \in \{-1,+1\},\ 1 \leq j < N_1\}, 1 \leq i < L_c.$$

3) The training sequence $\boldsymbol{T}\{T_n;\ n = 1\ldots 63\}$, $T_n \in \{-1,1\}$, which is a key-based sequence, should be distributed all over the image in order to survive all kinds of attacks, especially cropping. To this purpose, in our work, the training sequence $\boldsymbol{T}$ (63 bits) is embedded in row 16 and column 16 of the LL_4 subband. It is noted that a different combination of row and column corresponding to the important image portion can also be chosen. The informative watermark $\boldsymbol{W}$ is 2-D interleaved and embedded in the leftover portion of the LL_4 subband (Fig. 2).

In implementation, we allocate the 63 bits of the training sequence $\boldsymbol{T}$ into row 16 and column 16 of a 32 × 32 2-D array, we then de-interleave [18] it, resulting in a new 32 × 32 array. The binary string $\boldsymbol{W}$ is embedded into the remaining portion of the above-mentioned array.

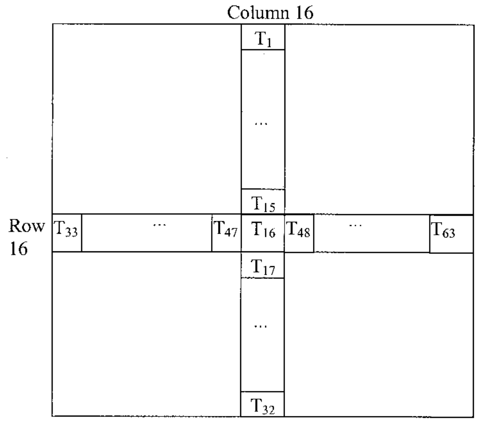


Fig. 2. Training data set.

By applying 2-D interleaving technique [19] to this array, we obtain another 2-D array. Scanning this 2-D array, say, row by row, we convert it into a 1-D array $\boldsymbol{X}$.

Coefficients of the LL_4 subband are scanned in the same way as in the embedding, resulting in a 1-D array, denoted by $\boldsymbol{C}$. Similar to our nonblind watermarking technique [6], we embed the binary data $\boldsymbol{X}$ into $\boldsymbol{C}$ to obtain $\boldsymbol{C}'$ according to (1), where α is a parameter related to the watermark embedding strength, $C(i)$ and $C'(i)$ denote the amplitude of the i^{th} element in $\boldsymbol{C}$ and $\boldsymbol{C}'$, respectively. Note that the difference between $C(i)$ and $C'(i)$ is between -0.5α and $+0.5\alpha$. If $x_i = +1$, $C'(i) \bmod \alpha = (3\alpha/4)$. If $x_i = -1$, $C'(i) \mathrm{mod}\, \alpha = (\alpha/4)$. So in the extraction, if the extracted coefficient $C^*(i)$ has $C^*(i) \mathrm{mod}\, \alpha > (\alpha/2)$, then the recovered binary bit $x_i^* = +1$. Otherwise, $x_i^* = -1$. In order to extract an embedded bit correctly, the absolute error (introduced by image distortion) between $C^*(i)$ and $C'(i)$, as in (1), shown at the bottom of the page, must be less than 0.25α. The parameter α can be chosen so as to make a good compromise between the contending requirements of imperceptibility and robustness. We choose α to be 90 in our work.

This training sequence helps to achieve synchronization against translation possibly applied to stego-image. If the correlation coefficient between the training sequence $\boldsymbol{T}$ and the test sequence $\boldsymbol{S}$ obtained from a test image satisfies the following condition $\rho_{T,S} = (1/63)\sum_{n=1}^{63}(T_n \cdot S_n) \geq \mathrm{thresh}_1$, we regard $\boldsymbol{S}$ as matched to $\boldsymbol{T}$ and consider synchronization is achieved. Here, we can calculate the corresponding probability of false positive (false synchronization) as $H_{fp} = (1/2^{63}) \cdot \sum_{k=63-e}^{63}\binom{63}{k}$, where $e = \mathrm{round}(63/2(1 - \mathrm{thresh}_1))$, and $round(\cdot)$ means taking the nearest integer. In our work, we choose $\mathrm{thresh}_1 = 0.56$ (empirically value), thus have $H_{fp} = 5.56 \times 10^{-6}$, which may be sufficient low for many applications.

4) Performing IDWT on the modified DWT coefficients, we produce the watermarked image $f'(x,y)$. The PSNR of thus generated marked images $f'(x,y)$ versus the original image is higher than 42.7 dB.

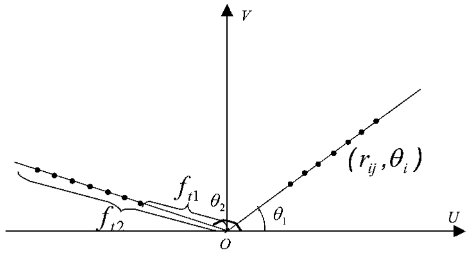


Fig. 3. Template embedding.

B. Template Embedding in DFT Domain

Since the DWT coefficients are not invariant under geometric transformation, to resist affine transform, we embed a template in DFT domain of the watermarked image inspired by [15]. The template embedding is as follows.

1) In order to have a required high resolution, the image $f'(x,y)$ is padded with zeros to a size of 1024 × 1024, the fast Fourier transform (FFT) is then applied.
2) 14 template points uniformly distributed along two lines (refer to Fig. 3) are embedded, seven points each line in the upper half plane in the DFT domain at angles θ_1 and θ_2 with radii varying between f_{t1} and f_{t2}. The angles θ_i and radii r_{ij}, where $i = 1, 2, j = 1, \ldots, 7$, may be chosen pseudo-randomly as determined by a key. We require at least two lines in order to resolve ambiguities arising from the symmetry of the magnitude of the DFT, and we choose to use only two lines because adding more lines increases the computational cost of template detection dramatically. We empirically find that seven points per line are enough to lower false positives probability to a satisfactory extent during detection. But to achieve more robustness against JPEG compression than the technique reported in [15], a lower frequency band, say $f_{t1} = 200$ and $f_{t2} = 305$, is used for embedding the template. This corresponds to 0.2 and 0.3 in the normalized frequency, which is lower than the band of 0.35~0.37 used in [15]. Since we do not embed the informative watermark in the magnitude spectrum of DFT domain, to be more robust to JPEG compression, a larger strength of the template points is chosen than in [15]. Concretely, instead of the local average value plus two times of standard deviation [15], we use the local average value of DFT points plus

$$
\begin{cases}
C'(i) = C(i) - (C(i)\mathrm{mod}\alpha) + \frac{3}{4}\alpha, & \text{if } x_i = 1 \text{ and } (C(i)\mathrm{mod}\alpha) \geq \frac{1}{4}\alpha \\
C'(i) = \left[C(i) - \frac{1}{4}\alpha\right] - \left[\left(C(i) - \frac{1}{4}\alpha\right)\mathrm{mod}\alpha\right] + \frac{3}{4}\alpha, & \text{if } x_i = 1 \text{ and } (C(i)\mathrm{mod}\alpha) < \frac{1}{4}\alpha \\
C'(i) = C(i) - (C(i)\mathrm{mod}\alpha) + \frac{1}{4}\alpha, & \text{if } x_i = 0 \text{ and } (C(i)\mathrm{mod}\alpha) \leq \frac{3}{4}\alpha \\
C'(i) = \left[C(i) + \frac{1}{2}\alpha\right] - \left[\left(C(i) - \frac{1}{2}\alpha\right)\mathrm{mod}\alpha\right] + \frac{1}{4}\alpha, & \text{if } x_i = 0 \text{ and } (C(i)\mathrm{mod}\alpha) > \frac{3}{4}\alpha
\end{cases}
\tag{1}
$$

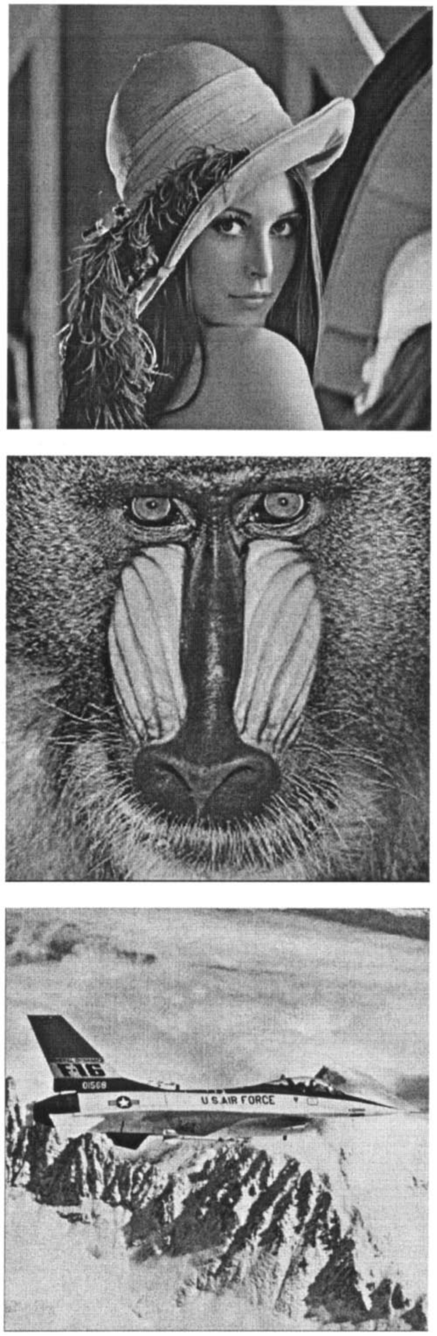


Fig. 4. Original images, *Lena*, *Baboon*, and *Plane*.

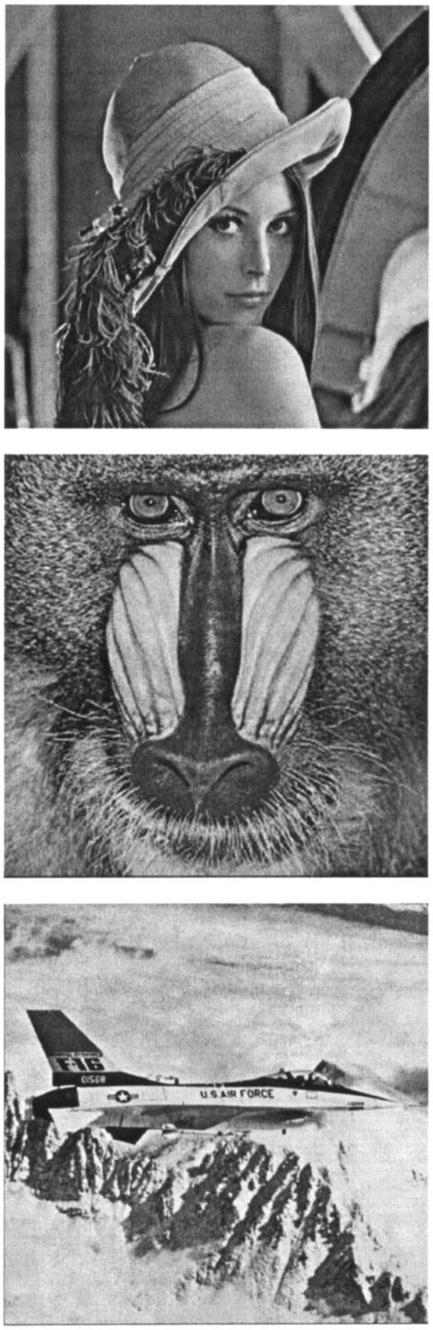


Fig. 5. Watermarked images with $\mathrm{PSNRs} > 42.5\ \mathrm{dB}$.

five times of standard deviation. According to our experimental results, this larger strength and lower frequency band have little effect on the invisibility of the embedded watermark (refer to Figs. 4 and 5).

3) Correspondingly another set of 14 points are embedded in the lower half plane to fulfill the symmetry constraint.

4) Calculate the inverse FFT, we obtain the DWT-DFT composite watermarked image $f''(x,y)$. The PSNR of $f''(x,y)$ versus the original image is 42.5 dB, which is reduced by 0.2 dB compared with the PSNR of $f'(x,y)$ versus the original image due to the template embedding. The experimental results demonstrate that the embedded data are perceptually invisible.

III. Watermark Extraction With Proposed Resynchronization

In order to extract the hidden information, we extract a data sequence $\boldsymbol{S}\{S_n;\ n = 1 \ldots 63\}$ in row 16 and column 16 in the DWT LL_4 subband of the to-be-checked image $g(x,y)$, which is rescaled to the size of the original image at first, in our work, 512×512. (We assume that the size of the original image is known to the detector.) If, $\rho_{\tau,S} \geq thresh_1$, we can then extract the informative watermark and recover the message from LL_4 subband directly. Otherwise, we need to resynchronize the hidden data before extracting the informative watermark.

In order to resynchronize the hidden data after geometric distortion, we restore affine transform according to the template embedded, and then restore translation using the training sequence. Therefore the procedure of information extraction is divided into three phases: template detection, translation registration, and decoding.

A. Template Detection

We first detect the template embedded in DFT domain. By comparing the detected template with the originally embedded template, we can determine the affine transformation possibly applied to the test image. To avoid high computational complexity, we propose an effective method to estimate the affine transformation matrix.

A linear transform applied in spatial domain results in a corresponding linear transform in DFT domain. That is, if a linear transform $\boldsymbol{B}$ is applied to an image in spatial domain as follows:

$$\begin{bmatrix} x \\ y \end{bmatrix} \rightarrow \boldsymbol{B} \begin{bmatrix} x \\ y \end{bmatrix} \tag{2}$$

then correspondingly the following transform takes place in DFT domain [15]:

$$\begin{bmatrix} u \\ v \end{bmatrix} \rightarrow (\boldsymbol{B}^{-1})^T \begin{bmatrix} u \\ v \end{bmatrix}. \tag{3}$$

The template detection is conducted in the following way, which is basically the same as in [15] except a more efficient way to estimate the linear transform matrix.

1) Apply a Bartlett window to the to-be-checked image $g(x,y)$ to produce the filtered image Iw.
2) Calculate the FFT of the image padded with zero to the size of 1024 × 1024. A higher resolution in the FFT is expected to result in a more accurate estimation of the undergone transformations. However, as we increase the amount of zero-padding, we also increase the volume of calculations. We find that using a size of 1024 × 1024 yields a suitable compromise.
3) Extract and record the positions of all the local peaks (p_{ui}, p_{vi}) in the image. Sort the peaks by angle and divide them into N_b equally spaced bins in terms of angle.
4) For both lines in the template, perform the following.

   For each of the equally spaced bins search for a K where $k_{\min} < K <_{\max}$ such that at least N_m local peaks having radial coordinate r_{Ii}, where $i \in 1 \ldots N_b$, match the peaks in the original template having radial coordinate r_{Tj} along line j where $j \in 1, 2$. Here, by match it is meant that $|r_{Ii} - Kr_{Tj}| < thresh_2$. If at least N_m points match, we store the set of matched points. (In our work, we choose $K_{\min} = 0.5$ and $K_{\max} = 2$. The corresponding scaling factor considered is hence between 2 and 0.5.)
5) From all sets of matched points, choose one set matching to template line 1 and another to template line 2 such that the angle between two sets deviates from the difference between θ_1 and θ_2 as shown in Fig. 3 within a threshold θ_{diff}. Calculate a transformation matrix $\boldsymbol{A}$ such that the mean square error (mse) defined in (4) is minimized as follows:

$$\frac{1}{\text{nummatches}} \left\| A \begin{bmatrix} u_{11} & v_{11} \\ \vdots & \vdots \\ u_{1l_1} & v_{1l_1} \\ u_{21} & v_{21} \\ \vdots & \vdots \\ u_{2l_2} & v_{2l_2} \end{bmatrix}^T - \begin{bmatrix} u'_{11} & v'_{11} \\ \vdots & \vdots \\ u'_{1l_1} & v'_{1l_1} \\ u'_{21} & v'_{21} \\ \vdots & \vdots \\ u'_{2l_2} & v'_{2l_2} \end{bmatrix}^T \right\|^2 \tag{4}$$

where (u, v) with subscripts represent a peak point's coordinate, (u', v') with subscripts represent the original (known) template's coordinate, $\text{nummatches} = l_1 + l_2$ is the number of the matching points. We note that $\boldsymbol{A}$ is a 2 × 2 transformation matrix $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ and the matrix inside $\| \cdots \|$ is of $2 \times (l_1 + l_2)$. The notation $\| \cdots \|^2$ denotes the sum of all of these squared error elements in the error matrix. The rows contain the errors in estimating the u and v from the original (known) template positions u' and v' after applying the linear transformation.

In the following, we propose a method to estimate the transformation matrix $\boldsymbol{A}$. It is noted that (5) links the set of matched points $(u_{i_1j_1}, v_{i_1j_1})$ and the set of the

$$\begin{bmatrix} u_{11} & v_{11} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ u_{1l_1} & v_{1l_1} & 0 & 0 \\ u_{21} & v_{21} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ u_{2l_2} & v_{2l_2} & 0 & 0 \\ 0 & 0 & u_{11} & v_{11} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & u_{1l_1} & v_{1l_1} \\ 0 & 0 & u_{21} & v_{21} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & u_{2l_2} & v_{2l_2} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} u'_{11} \\ \vdots \\ u'_{1l_1} \\ u'_{21} \\ \vdots \\ u'_{2l_2} \\ v'_{11} \\ \vdots \\ v'_{1l_1} \\ v'_{21} \\ \vdots \\ v'_{2l_2} \end{bmatrix} \tag{5}$$

original template points $(u'_{l_1j_1}, v'_{l_1j_1})$ with $i_1 = 1, 2$; $j_1 = 1, 2, \ldots, l_1$ or l_2; $N_m \leq l_1$ or $l_2 \leq 7$.

Equation (5) can be rewritten as the following matrix-vector format:

$$\boldsymbol{M}\boldsymbol{A_c} = \boldsymbol{N} \tag{6}$$

or

$$\begin{bmatrix} \boldsymbol{M}_1 & 0 \\ 0 & \boldsymbol{M}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{N}_1 \\ \boldsymbol{N}_2 \end{bmatrix}. \tag{7}$$

We seek $\boldsymbol{A}$ that minimizes $\|\boldsymbol{E}\|^2 = \|\boldsymbol{N} - \boldsymbol{M}A_c\|^2$. The solution that satisfies the requirement is [20]

$$\boldsymbol{M}_1^T \boldsymbol{M}_1 \boldsymbol{A}_1 = \boldsymbol{M}_1^T \boldsymbol{N}_1 \tag{8}$$

$$\boldsymbol{M}_1^T \boldsymbol{M}_1 \boldsymbol{A}_2 = \boldsymbol{M}_1^T \boldsymbol{N}_2. \tag{9}$$

Since the matrix $\boldsymbol{M}_1^T \boldsymbol{M}_1$ is a positive definite 2 × 2 matrix, $\boldsymbol{M}_1^T \boldsymbol{N}_1$ and $\boldsymbol{M}_1^T \boldsymbol{N}_2$ are 2 × 1 matrices, the above two linear equation systems can

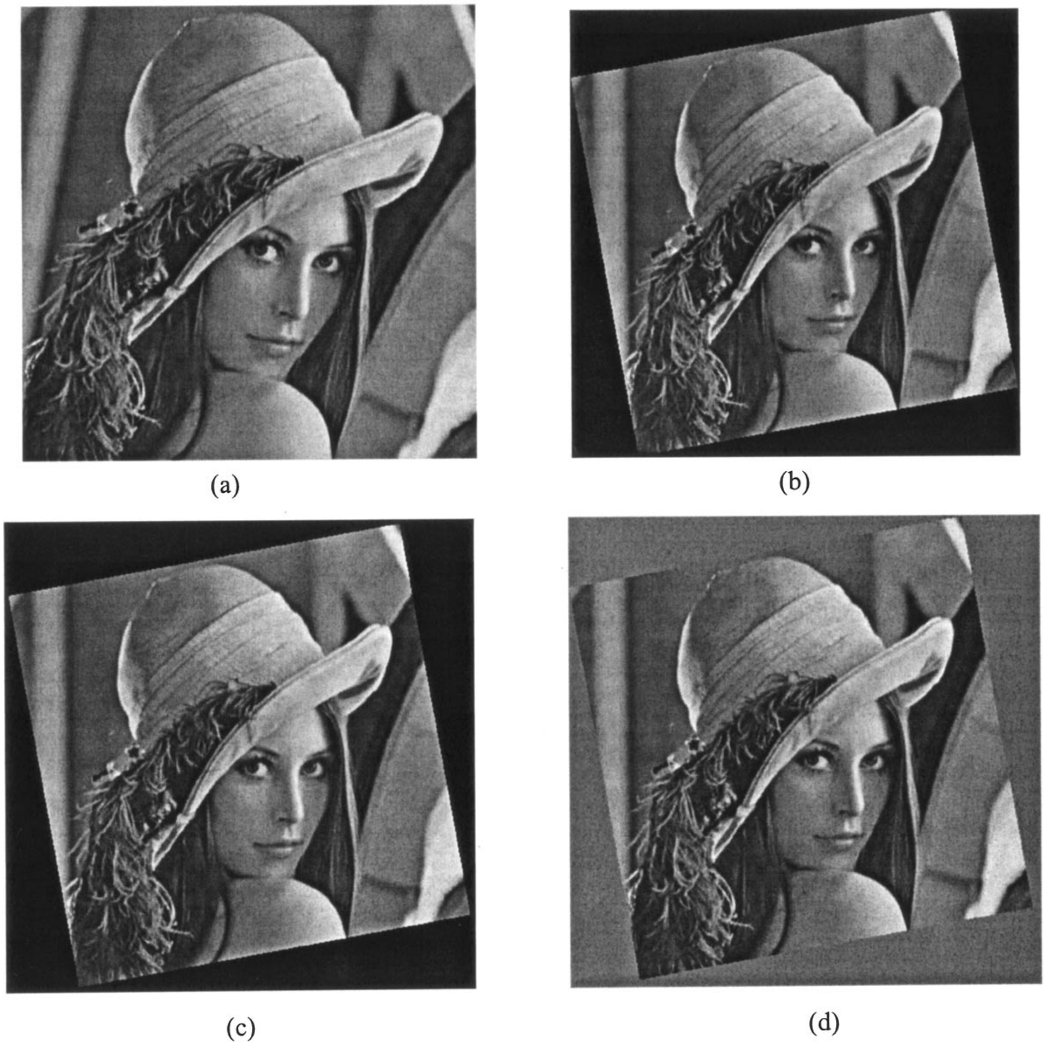


Fig. 6. Resynchronization: (a) the to-be-checked image $g(x,y)$, which is of 512 × 512 and is experienced a rotation of 10°, scaling, translation, cropping, and JPEG compression with quality factor of 50; (b) the image $g'(x,y)$, which is of 504 × 504 and has been recovered from the linear transform applied; (c) the image $I(x,y)$, which has been padded with 0s to the size of 512 × 512; and (d) the resynchronized image $g^*(x,y)$, which is of 512 × 512 and has been padded with the mean gray-scale value of the image $g(x,y)$. The embedded message was finally recovered without error.

be easily solved for $\boldsymbol{A}_1 = (\boldsymbol{M}_1^T\boldsymbol{M}_1)^{-1}\boldsymbol{M}_1^T\boldsymbol{N}_1$ and $\boldsymbol{A}_2 = (\boldsymbol{M}_1^T\boldsymbol{M}_1)^{-1}\boldsymbol{M}_1^T\boldsymbol{N}_2$, thus we obtain the candidate of transformation matrix $\boldsymbol{A} = \left[\begin{smallmatrix}\boldsymbol{A}_1^T\\ \boldsymbol{A}_2^T\end{smallmatrix}\right]$ and the corresponding mse according to (4) is

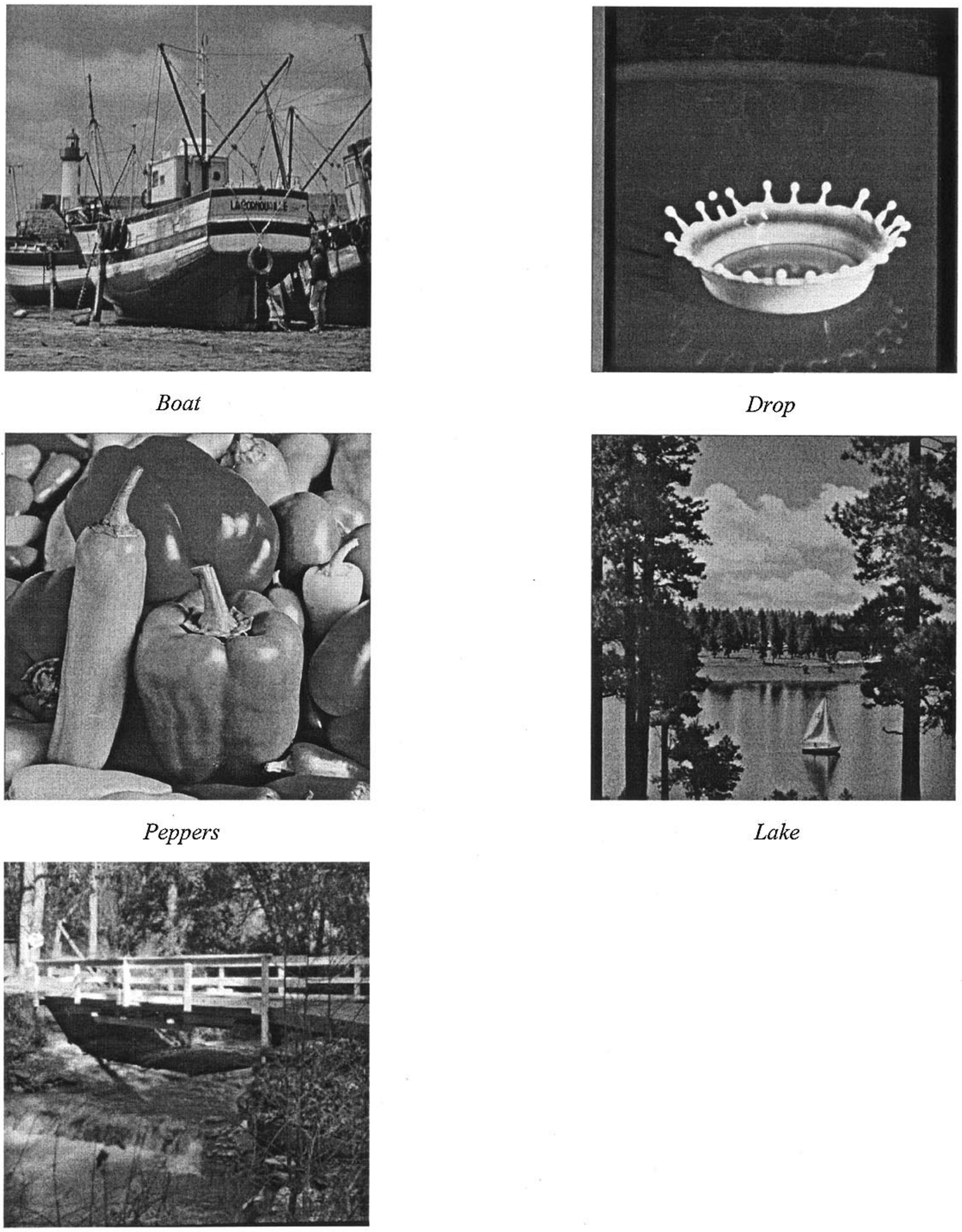
$$\begin{aligned}\text{mse} &= \frac{1}{\text{nummatches}}\|E\|^2\\ &= \frac{1}{\text{nummatches}}\\ &\quad\times\left(\|\boldsymbol{N}_1 - \boldsymbol{M}_1\boldsymbol{A}_1\|^2 + \|\boldsymbol{N}_2 - \boldsymbol{M}_1\boldsymbol{A}_2\|^2\right). \qquad (10)\end{aligned}$$

6) Since we only work on the upper half plane, in order to resolve possible ambiguities, we add 180° to the angles in the sets of matched points corresponding to line 1 of the template (either line can be used), then repeat the previous step.
7) Choose the $\boldsymbol{A}$ that results in the smallest mse.
8) If the minimized error is larger than the detection threshold T_d, we conclude that no watermark was embedded in the image. Otherwise we proceed to decoding. According to (2) and (3), we can obtain the linear transform matrix $\boldsymbol{B}$, described at the beginning of Section II, and $\boldsymbol{B} = \boldsymbol{A}^{\mathrm{T}}$. Applying the inverse linear transform, $\boldsymbol{B}^{-1}$, to the image $g(x,y)$, we obtain an image $g'(x,y)$, that has correct the applied linear transform. One example is shown in Fig. 6(a) and (b).

B. Translation Registration and Decoding

Assume the linear transform corrected image $g'(x,y)$ has size of $M \times N$. Padding $g'(x,y)$ with 0s to the size of 512 × 512 generates the image $I(x,y)$ [Fig. 6(c)].

One way to restore translation is to search by brute-force for the largest correlation coefficient between the training sequence $\boldsymbol{T}$ and the data sequence $\boldsymbol{S}$ extracted from the DWT coefficients in row 16 and column 16 of LL_4 subband corresponding to the



Boat

Drop

Peppers

Lake

Bridge

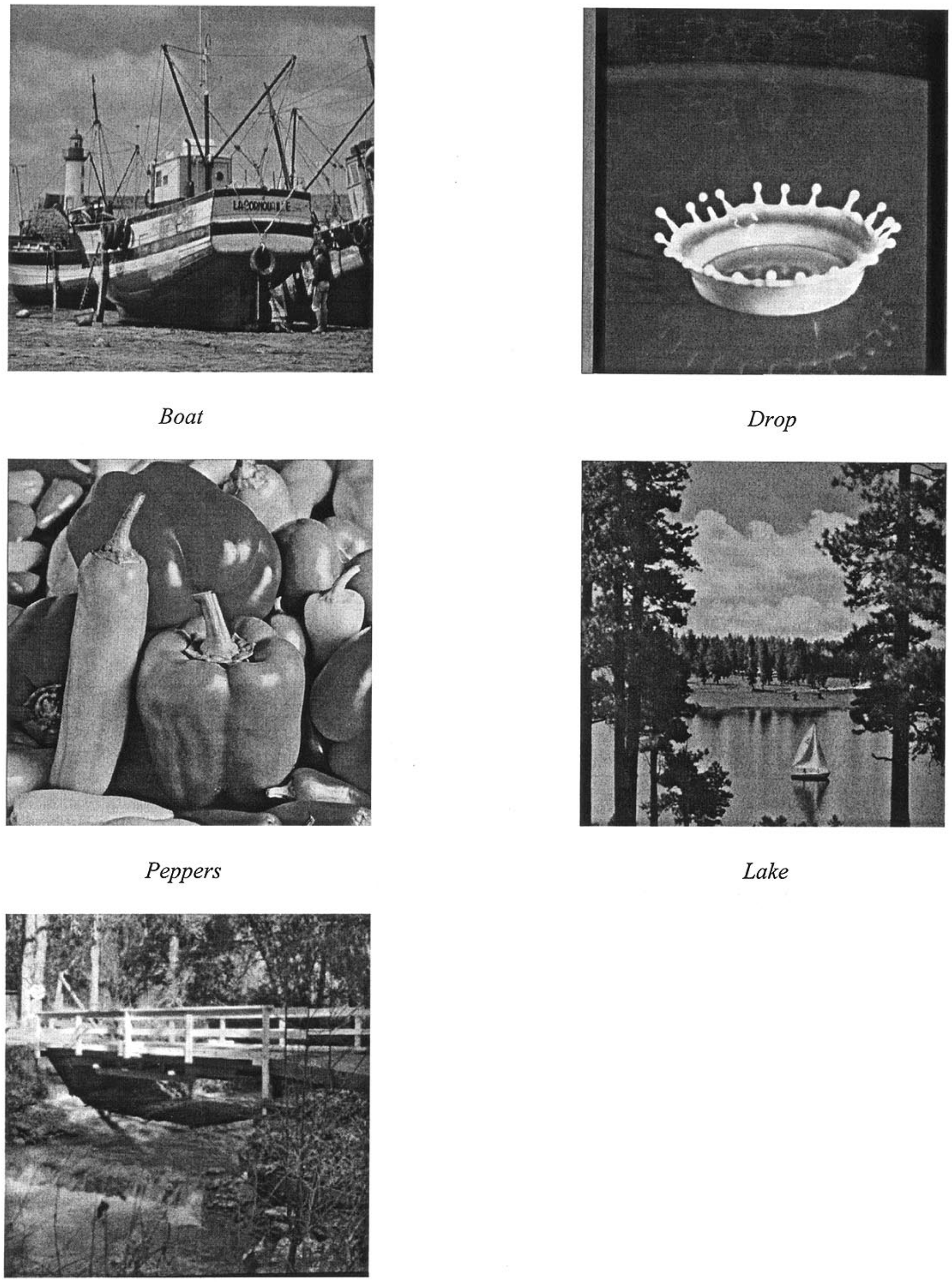
Fig. 7. Some original images used in our test.

following set of all possible translated images. (Note that we construct $\boldsymbol{S}$ sequence in the way as we embed the $\boldsymbol{T}$ sequence in the LL_4 subband, refer to Fig. 2)

$$\begin{aligned} I_t(x,y) = I\left((x-t_x)\mathrm{mod}512,(y-t_y)\mathrm{mod}512\right) \\ \left\{-\frac{1}{2}(512-M) \leq t_x < \frac{1}{2}(512-M)\right. \\ \left. -\frac{1}{2}(512-N) \leq t_y < \frac{1}{2}(512-N)\right\} \end{aligned} \quad (11)$$

where t_x and t_y are the translation parameters in the spatial domain. This method demands a heavy computational load when $(512-M) > 16$ and/or $(512-N) > 16$. We dramatically reduce the required computational load by performing DWT for at most 256 cases according to the dyadic nature of the DWT. That is, if an image is translated by $16x_{t1}$ rows and/or $16y_{t1}$ columns (x_{t1}, $y_{t1} \in \mathrm{Z}$), then the LL_4 subband coefficients of the image are translated by x_{t1} rows and/or y_{t1} columns accordingly. This property is utilized to efficiently handle translation synchronization in our algorithm. That is, we have $t_x = 16x_{t1} + x_t$, $t_y = 16y_{t1} + y_t$, where $-8 \leq x_t, y_t < +8$. In each of the 256 pairs of (x_t, y_t), we perform DWT on the translated image $I_t(x,y)$, generating the LL_4 coefficients, denoted by $\mathrm{LL}_{4t}(x,y)$. We then perform translations on the $LL_{4t}(x,y)$

$$\begin{aligned} LL'_{4t}(x,y) = LL_{4t}\left((x-x_{t1})\mathrm{mod}\,32,(y-y_{t1})\mathrm{mod}\,32\right) \\ \{-T_1 \leq x_{t1} < T_1; -T_2 \leq y_{t1} < T_2\} \end{aligned} \quad (12)$$



Boat

Drop

Peppers

Lake

Bridge

Fig. 8. Watermarked images with $\mathrm{PSNRs} > 42.5\ \mathrm{dB}$.

where x_{t1}, y_{t1} are the translation parameters in the LL_4 subband, $T_1 = round\ ((1/2)(512 - M)/16)$, $T_2 = round\ ((1/2)(512 - N)/16)$. Each time, we extract the data sequence $\boldsymbol{S}$ in row 16 and column 16 in the $LL'_{4t}(x, y)$. The synchronization is achieved when $\rho_{T,S} \geq \mathrm{thresh}_1$ or $\rho_{T,S}$ is largest. For example, for the image in Fig. 6(c), the maximum correlation coefficient $\rho_{T,S}(= 0.87)$ is achieved when $x_t = -3$, $y_t = -4$ and $x_{t1} = 0$, $y_{t1} = 0$. At last, we obtain the translation parameters ($t_x = 16 \times x_{t1} + x_t$, $t_y = 16 \times y_{t1} + y_t$).

After restoring affine transform and translation, and padding with the mean gray-scale value of the image $g(x, y)$, we obtain the resynchronized image $g^*(x, y)$ [Fig. 6(d)]. The LL_4 coefficients of $g^*(x, y)$ are scanned using the same way as in the data embedding, resulting a 1-D array, $\boldsymbol{C}^*$. The extracted hidden binary data, denoted by $\boldsymbol{X}^* = \{x_i^*\}$, are extracted as follows:

$$x_i^* = \begin{cases} +1, & C^*(i) \mathrm{mod} \alpha > \frac{\alpha}{2} \\ -1, & \mathrm{otherwise} \end{cases} \tag{13}$$

De-interleaving [18] the 32 × 32 2-D array, constructed from $\boldsymbol{X}^*$, we can obtain the recovered binary data $\boldsymbol{W}^*$. We segment $\boldsymbol{W}^*$ by N_1 bits per sequence, correlate the obtained sequence

TABLE I
EXPERIMENTAL RESULTS WITH STIRMARK 3.1

StirMark functions	*Lena*	*Baboon*	*Plane*	*Boat*	*Drop*	*Pepper*	*Lake*	*Bridge*
JPEG 10~100	1	1	1	1	1	1	1	1
Scaling	1	1	1	1	1	1	1	1
Jitter	1	1	1	1	1	1	1	1
Cropping_25	1	1	1	1	1	1	1	1
Aspect ratio	1	1	1	1	1	1	1	1
Rotation (auto-crop, scale)	1	1	1	1	1	1	1	1
General linear transform	1	1	1	1	1	1	1	1
Shearing	1	1	1	1	1	1	1	1
Gauss filtering	1	1	1	1	1	1	1	1
Sharpening	1	1	1	1	1	1	1	1
FMLR	1	1	1	1	1	1	1	1
2x2median_filter	1	0	1	0	1	1	1	0
3x3median_filter	1	0	1	0	1	0	0	0
4x4median_filter	0	0	0	0	1	0	0	0
Random bending	0	0	0	0	0	0	0	0

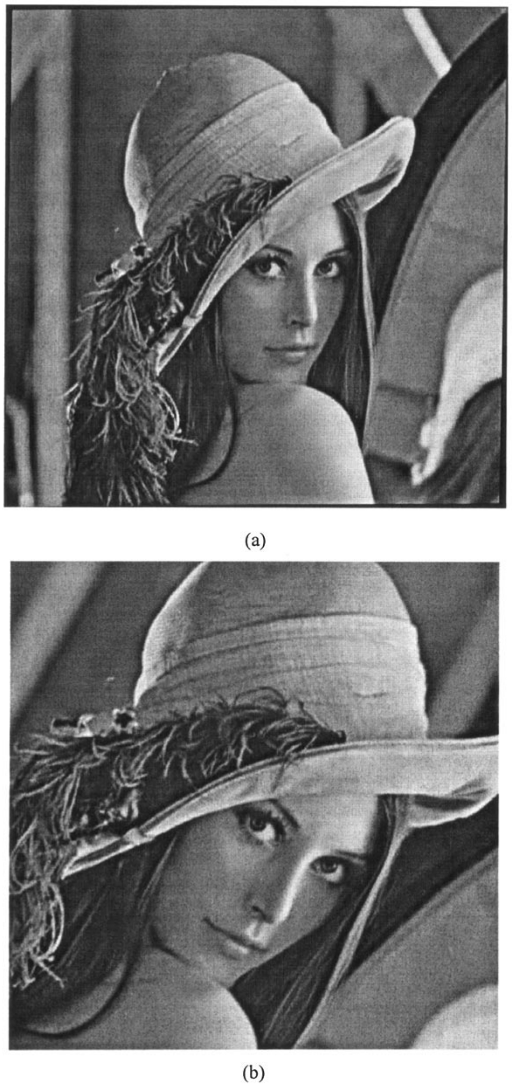
with the original PN-sequence $\boldsymbol{p}$. If the correlation value is larger than 0, the recovered bit is "1", otherwise, "0". The binary bit sequence $\boldsymbol{b}$ can thus be recovered.

The recovered bit sequence $\boldsymbol{b}$ is now BCH decoded. In our work, we use the BCH (72, 60). Hence, if there are fewer than five errors, the message $\boldsymbol{m}$ will be recovered without error, otherwise, the embedded message cannot be recovered correctly.

IV. EXPERIMENTAL RESULTS

We have tested the proposed algorithm on images shown in Fig. 4 and Fig. 7. The results are reported in Table I. In our work, we choose $L = 60$, $N_1 = 11$, $N_b = 180$, $N_m = 5$, $\text{thresh}_1 = 0.56$, $\text{thresh}_2 = 0.002$, the detection threshold $T_d = 1.0 \times 10^{-6}$. The PSNRs of the marked images are higher than 42.5 dB (see Fig. 8). The watermarks are perceptually invisible. The watermark embedding takes less than 4 s, while the extraction takes about 2∼38 s on a 1.7-GHz Pentium PC using C language.

Fig. 9 shows a marked *Lena* image that has undergone JPEG compression with quality factor 50 (JPEG_50) in addition to general linear transform [a StirMark test function: linear_ 1.010_0.013_0.009_1.011, Fig. 9(a)] or rotation 30° [auto-crop, auto-scale, Fig. 9(b)]. In both cases, the embedded message (60 information bits) can be recovered with no error. This demonstrates that our watermarking method is able to resist both affine transforms and JPEG compression. Table I shows more test results with our proposed algorithms by using StirMark 3.1. In Table I, "1" represents the embedded 60-bit message can be recovered successfully while "0" means the embedded message cannot be recovered successfully. It is observed that the watermark is robust against Gaussian filtering, sharpening, FMLR, rotation (auto-crop, auto-scale), aspect ratio variations, scaling, jitter attack (random removal of rows and/or columns), general linear transform, shearing. In all of these cases, the embedded message can be recovered. We can also see that the watermark can effectively resist JPEG compression, and cropping. It is noted that our algorithm can


(a)

(b)

Fig. 9. Watermarked image that have undergone JPEG_50 in addition to an affine transform (StirMark function). (a) JPEG_50 +linear_1.010_0.013_0.009_1.011. (b) JPEG_50+rotation_scale_30.00.

TABLE II
PERFORMANCE COMPARISON OF THE PROPOSED SCHEME WITH THAT IN [15]

Scheme	PSNR(dB)of marked image	JPEG (QF)	shearing
Our scheme	>42.5	10~100	5%
Scheme in [15]	<38	75~100	0.1%

recover the embedded message for JPEG compression with quality factor as low as 10. The watermark is recovered when up to 65% of the image has been cropped. The watermark can also resist the combination of RST (rotation, scaling, translation, cropping) with JPEG_50 (Figs. 6 and 9).

We compare the proposed scheme with the scheme in [15] on *Lena* image using StirMark 3.1. The results are shown in Table II. Both schemes hide a 60-bit message in a $512 \times 512 \times 8$ image. It is noted that our scheme performs much better in resisting JPEG compression. It is robust against JPEG with quality factor as low as 10 instead of 75 in [15]. Table II indicates that our watermark can be recovered without error against shearing of less than or equal to 5% (shearing_5.00_5.00), while the watermark in [15] can only be recovered correctly against shearing of less than or equal to against 0.1% ([15, p. 1127]). While the watermark embedded with our proposed method is more robust against JPEG and geometric distortions, the PSNR of the marked image versus the original image with our method is higher than 42.5 dB and the PSNR with [15] is not higher than 38 dB.

We have also tested the proposed algorithm using different wavelet filters, such as orthogonal wavelet filters Daubechies-N $(\mathrm{N} = 1 \sim 10)$ and other bi-orthogonal wavelet filters (for example, Daubechies 5/3 wavelet filter). The similar results have been obtained.

V. CONCLUSIONS

The main contributions presented in this paper are as follows.

1) We proposed a DWT-DFT composite watermarking scheme that is robust to affine transforms and JPEG compression simultaneously. Compared with other watermarking algorithms reported in the literature, the proposed technique is more robust and has a higher PSNR of marked image. The proposed watermarking can successfully resist almost all of the affine transform related test functions in StirMark 3.1 and JPEG compression with quality factor as low as 10.
2) We proposed to use a training sequence embedded in the DWT domain to achieve synchronization against translation. By using the dyadic property of the DWT, we reduced the number of the DWT implementation dramatically, hence lowering the computational complexity.
3) A new method to estimate the affine transform matrix, expressed in (8) and (9), is proposed. It can reduce the high computational complexity required in the iterative computation.

However, the robustness of the informative watermark against median filtering and random bending needs to be improved. Robustness against median filtering may be improved by increasing the strength of the informative watermark via adaptive embedding based on perceptual masking [21]–[24]. Moreover, it is noted that the template embedded in the DFT domain may be removed by the attacker [25]. These issues are our future research subjects.

The proposed technique for embedding character strings into gray-scale images may find applications in the intellectual property protection and anonymous communications. It is noted that although the algorithm is presented for gray-level images, it may be applied to color images and video sequences in a rather straightforward way. For a color image, we can apply the proposed algorithm to the luminance component of the image. For a video sequence, we may treat each frame as a gray-level image and hide a bit stream into the frame [26]. Another feasible way is to simply embed a few symbols into each of I-frames in an MPEG-compressed video sequence.

ACKNOWLEDGMENT

The authors are grateful to the constructed review comments, which have helped to enhance the quality of this paper.

REFERENCES

[1] F. Hartung and M. Kutter, “Multimedia watermarking techniques,” *Proc. IEEE*, vol. 87, no. July, pp. 1079–1107, 1999.

[2] F. Deguillaume, S. Voloshynovskiy, and T. Pun, “A method for the estimation and recovering from general affine transforms in digital watermarking applications,” in *Proc. SPIE: Security and Watermarking of Multimedia Contents IV*, vol. 4675, San Jose, CA, Jan. 2002, pp. 313–322.

[3] C.-Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, M. L. Miller, and Y.-M. Lui, “Rotation, scale, and translation resilient watermarking for images,” *IEEE Trans. Image Processing*, vol. 10, pp. 767–782, May 2001.

[4] J.-L. Dugelay and F. A. P. Petitcolas, “Possible counter-attackers against random geometric distortions,” in *Proc. SPIE: Security and Watermarking of Multimedia Contents II*, vol. 3971, CA, Jan. 2000.

[5] G. W. Braudaway and F. Minter, “Automatic recovery of invisible image watermarks from geometrically distorted images,” in *Proc. SPIE: Security and Watermarking of Multimedia Contents I*, vol. 3971, CA, Jan. 2000.

[6] X. Kang, J. Huang, and Y. Q. Shi, “An image watermarking algorithm robust to geometric distortion,” in *Lecture Notes in Computer Science: Proc. Int. Workshop on Digital Watermarking 2002 (IWDW2002)*, vol. 2613, Seoul, Korea, 2002, pp. 212–223.

[7] J. J. K. O’Ruanaidh and T. Pun, “Rotation, scale and translation invariant spread spectrum digital image watermarking,” *Signal Processing*, vol. 66, no. 3, pp. 303–317, 1998.

[8] M. Kutter, “Watermarking resistance to translation, rotation, and scaling,” in *Proc. SPIE: Multimedia Systems Applications*, vol. 3528, 1998, pp. 423–431.

[9] S. Voloshynovskiy, F. Deguillaume, and T. Pun, “Content adaptive watermarking based on a stochastic multiresolution image modeling,” in *Proc. 10th European Signal Processing Conf. (EUSIPCO’2000)*, Tampere, Finland, Sept. 2000.

[10] H. J. Shim and B. Jeon, “Rotation, scaling, and translation robust image watermarking using Gabor kernels,” in *Proc. SPIE: Security and Watermarking of Multimedia Contents IV*, vol. 4675, San Jose, CA, Jan. 2002, pp. 563–571.

[11] S. Voloshynovskiy, S. Pereira, A. Herrigel, N. Baumgärtner, and T. Pun, “Generalized watermark attack based on watermark estimation and perceptual remodulation,” in *Proc. SPIE: Electronic Imaging 2000, Security and Watermarking of Multimedia Content II*, vol. 3971, San Jose, CA, 2000.

[12] S. Voloshynovskiy, F. Deguillaume, and T. Pun, “Multibit digital watermarking robust against local nonlinear geometrical distortions,” in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Thessaloniki, Greece, 2001, pp. 999–1002.

[13] D. J. Fleet and D. J. Heger, “Embedding invisible information in color images,” in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, Santa Barbara, CA, 1997, pp. 532–535.

[14] W. Bender, D. Gruhl, and N. Morimoto, "Techniques for data hiding," *IBM Syst. J.*, vol. 35, no. 3/4, pp. 313–337, 1996.

[15] S. Pereia and T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE Trans. Image Processing*, vol. 9, pp. 1123–1129, 2000.

[16] I. J. Cox, J. Killian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Processing*, vol. 6, pp. 1673–1687, Dec. 1997.

[17] J. Huang, Y. Q. Shi, and Y. Shi, "Embedding image watermarks in DC components," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, pp. 974–979, 2000.

[18] G. F. Elmasry and Y. Q. Shi, "2-D interleaving for enhancing the robustness of watermark signals embedded in still image," in *Proc. IEEE Int. Conf. Multimedia and Expo*, 2000, p. **AUTHOR: PLS. PROVIDE PAGES**.

[19] Y. Q. Shi and X. M. Zhang, "A new two-dimensional interleaving technique using successive packing," *IEEE Trans. Circuits Syst. I*, vol. 49, pp. 779–789, June 2002.

[20] G. W. Stewart, *Introduction to Matrix Computations*. New York: Academic, 1973.

[21] C. I. Podilchuk and W. Zeng, "Image-adaptive watermarking using visual models," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 525–539, 1998.

[22] J. Huang and Y. Q. Shi, "An adaptive image watermarking scheme based on visual masking," *Electron. Lett.*, vol. 34, no. 8, pp. 748–750, 1998.

[23] S. Voloshynovskiy, A. Herrigel, N. Baumgärtner, and T. Pun, "A stochastic approach to content adaptive digital image watermarking," *Lecture Notes in Computer Science*, vol. 1768, pp. 212–236, 1999.

[24] S. Voloshynovskiy, F. Deguillaume, and T. Pun, "Content adaptive watermarking based on a stochastic multiresolution image modeling," in *Proc. 10th European Signal Processing Conf. (EUSIPCO'2000)*, Tampere, Finland, Sept. 2000.

[25] S. Voloshynovskiy, S. Pereira, V. Iquise, and T. Pun, "Attack modeling: toward a second generation benchmark," *Signal Processing*, vol. 81, no. 6, pp. 1177–1214, 2001.

[26] F. Hartung and B. Girod, "Watermarking of uncompressed and compressed video," *Signal Processing*, vol. 66, no. 3, pp. 283–301, 1998.

Xiangui Kang received the B.S. and M.S. degrees from Peking University, Peking, China, and Nanjing University, Nanjing, China, in 1990 and 1993, respectively. He is currently working toward the Ph.D. degree in the Department of Electronics and Communication Engineering, Sun Yat-Sen University, Guangzhou, China.

He is with the Department of Electronics and Communication Engineering, Sun Yat-Sen University. His research interests include image processing, data hiding, watermarking, and network security.

Jiwu Huang (SM'00) received the B.S. degree in electronic engineering from Xidian University, Xi'an, China, in 1982, the M.S degree in electronic engineering from Tsinghua University, Peking, China, in 1987, and the Ph.D. degree in pattern recognition and information systems from the Chinese Academy of Science, Peking, China, in 1998.

He is with the Department of Electronics and Communication Engineering, Sun Yat-Sen University, Guangzhou, China. His current research interests include image processing, image coding, data hiding, and watermarking.

Yun Q. Shi (M'90–SM'93) received the B.S. degree in electronic engineering and the M.S. degree in precision instrumentation from Shanghai Jiao Tong University, Shanghai, China, and the Ph.D. degree in electrical engineering from the University of Pittsburgh, Pittsburgh, PA.

He has been a Professor with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ, since 1987. His research interests include motion analysis from image sequences, video coding and transmission, and digital image watermarking.

Yan Lin received the B.E. degree in communication engineering in 2002 from Sun Yat-Sen University, Guangzhou, China, where he is currently working toward the Master's degree in the Department of Electronics and Communication Engineering.

His research interests include image watermarking, data hiding, and network security.