Mathematical Biosciences and Engineering

MBE, 16(5): 4788–4801.
DOI: 10.3934/mbe.2019241
Received: 08 February 2019
Accepted: 24 April 2019
Published: 28 May 2019

http://www.aimspress.com/journal/MBE

Research article

Robust digital watermarking for color images in combined DFT and DT-CWT domains

Qichao Ying [1], Jingzhi Lin [1], Zhenxing Qian [2,*] Haisheng Xu [1] and Xinpeng Zhang [2]

[1] School of Communication and Information Engineering, Shanghai University, Shanghai, P.R. China
[2] Shanghai Institute of Intelligent Electronics and Systems, School of Computer Science, Fudan University, Shanghai, China

* **Correspondence:** Email: zxqian@fudan.edu.cn; Tel: +86-13764870417.

Abstract: Image watermarking focuses on hiding secret data into the cover image imperceptibly to protect the copyright of the original image. In this paper, we propose a new framework of robust digital watermarking for color images using combined embedding techniques of Discrete Fourier Transform (DFT) and Dual Tree Complex Wavelet Transform (DTCWT). The cover image is first divided into Y, U and V channels. The Y channel is then transformed by DFT and partitioned into the ring shapes. With an embedding key, we generate pseudo-random patterns to represent the watermark. These patterns are also transformed and partitioned. The watermark represented by the selection of patterns is then embedded into the rings of the DFT coefficients. We further embed a rectification watermark into the U channel, in which DTCWT is applied to achieve a capability of geometric distortion resilience. On the recipient's side, the detection and extraction of watermark can be successfully done. Compared with previous schemes, the proposed method is better on preserving the image quality. Meanwhile, the robustness against typical attacks is also stronger.

Keywords: digital watermarking; image processing; robustness; DTCWT; image rectification

1. Introduction

Digital watermarking is becoming more and more popular with the rapid advances of data science. Internet gives the clients an open access to an explosively rising amount of data. However, these advantages have also brought the problems of unauthorized usage of the original images, such

as duplication, redistribution and tampering. It is essential to protect the legal ownership of the original data. Digital watermarking has been used in copyright protection and content authentication of images in multimedia. The technique embeds data into the host images using an imperceptible way. It can be used in various kinds of applications including copyright protection, content authentication and content description.

Robustness and fidelity are the two basic requirements for digital watermarking. On one hand, the embedded watermark must be robust enough against a variety of possible intentional and unintentional attacks. Intentional attacks, such as geometric distortion, rotating and cropping, are often used to destroy the watermark existence so as to distribute the copies without copyright. Unintentional attacks include image compression, addition of channel noise, filtering, etc. Generally, the unintentional attacks appear more often than intentional attacks. Common attacks that result in geometrical distortion usually cause synchronization error, which can dramatically deteriorate the performance of watermark detection. On the other hand, the watermarked image should be close enough to the cover image so that human eyes cannot distinguish the differences between them. Generally, there is a trade-off between robustness and fidelity, i.e., stronger robustness usually causes lower fidelity, and vice versa.

Existing watermarking schemes can roughly be divided into two categories: embedding in the spatial domain and embedding in transfer domains. Traditional digital watermarking schemes in the spatial domain [1–3] utilize contextual area or the high correlation between neighboring pixels in a host image to hide the watermark. However, watermarks embedded in the spatial domain have worse robustness than those embedded in transfer domains. In [4], the theory of Fourier-Mellin Transform (FMT) invariants is exploited to produce watermarks that are resistant to rotation, scaling and translation. Thereafter, Discrete Wavelet Transform (DWT) domain [4,5], Discrete Cosine Transform (DCT) domain [6,7] and Discrete Fourier Transform (DFT) domain [8,9] are widely referred to embed watermark. Log-polar mapping (LPM) methods [10] are also applied in image watermarking, which shows strong robustness to traditional image attacks while the computational complexities are much lower than [4]. The difficulty of extraction cause by geographical distortion is mainly alleviated in two ways. One is to resynchronize the received image before conducting watermark extraction. Another is to embed and extract watermark in a rotation and translation invariant or semi-invariant domain. For example, Niu et al. [11] extracts the stable feature points of the cover image by multi-scale SIFT detector, and the Bandelet transform is performed on the local feature regions (LFRs) to embed the digital watermark. Therefore, the detection of watermark can be conducted without synchronization error.

On the other hand, in many recent works, the contourlet coefficients of an image are widely used for watermarking, where the statistics of the contourlet coefficients have been investigated [12–14]. These methods give better embedding performance compared to the traditional methods like [11], and they require an accurate statistical characterization of images. Besides, there also appear some hybrid methods for image watermarking using the techniques in the combined domains [15–16].

While many of the above-mentioned schemes have strong resilience to rotating, scaling or compressing, the robustness against cropping is generally not good enough. However, cropping is one of the most popular attacks by the unauthorized users. Besides, blind detection of watermark can merely be promised in these schemes. So it is of great importance to develop a new framework of robust digital watermarking schemes to help addressing these issues.

In this paper, we propose a new framework of digital watermarking scheme in DFT-DTCWT domain for colored images. The framework is designed to be invariant to common attacks such as cropping, rotating, image processing, etc. The cover image is first divided into the Y, U, V channels. We embed a watermark into the DFT coefficients of the Y channel and some rectification information into the DTCWT coefficients of the U channel. Compared with previous state-of-the-art works, our scheme excels in a stronger robustness to common attacks and the transparency of the watermark.

The rest of this paper is organized as follows. Section II introduces the related words of digital watermarking. The proposed framework is depicted in Section 2. Section 3 provides the experimental results and Section 4 concludes the paper.

2. Proposed watermarking scheme

Suppose the colored cover image $\mathbf{I}_C$ is of the size $m_r \times m_c$, and the to-be-embedded watermark is denoted as $\boldsymbol{W}$ We first separate $\mathbf{I}_C$ into the Y, U, V channels. The independent channels are transformed from the typical R, G, B channels of the image. The embedding procedure includes two stages: watermark embedding in the Y channel and embedding some rectification information into the U channel. The V channel of the cover image remains unchanged. In the following, watermark embedding and extraction are respectively presented. The overview of the embedding procedure is depicted in Figure 1.

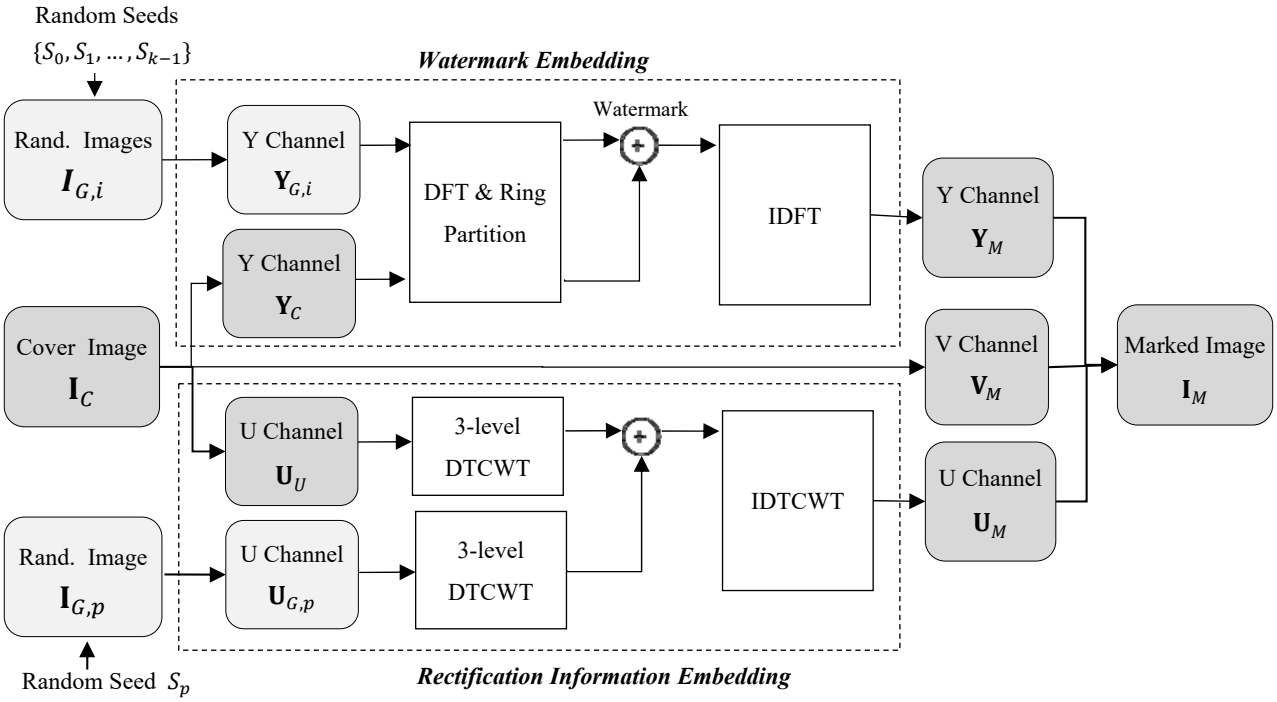

Figure 1. Overview of the embedding procedure of the proposed framework.

2.1. Watermark embedding

We first select k different pseudo-random seeds as private keys of the watermark embedding. The seed collection is denoted as $\{S_0, S_1, \dots, S_{k-1}\}$. Next, a pseudo-random sequence generator is applied to generate random sequence by using different seeds respectively. We get k pseudo-random sequences $\{Seq_0, Seq_1, \dots, Seq_{k-1}\}$, where Seq_0 is generated using S_0 as its generating seed.

Given two pseudo-random seed, the sequences which are of the same length and generated by the same seed are the same, which can also be represented as high correlation, and the correlation is close to 0 for two sequence generated by different seeds.

$$\left|Cor\left(Seq_i, Seq_j\right)\right| \approx \begin{cases} 0, i \neq j \\ 1, i = j \end{cases} \tag{2}$$

Here, $cor(x, y)$ represents the function of correlation calculation between x and y. We let the length of every pseudo-random sequence is at least $3 \times m_r \times m_c$, and the values of the elements in the sequences are integers within the range of [0,255]. Afterwards, we respectively generate an image pattern $\mathbf{I}_{G,i}$ by filling the R, G, B values of every pixel in the image with the pseudo-random values in Seq_i in a row-by-row order. Therefore, k different patterns, namely, $\mathbf{K} = \{\mathbf{I}_{G,0}, \mathbf{I}_{G,1}, \dots, \mathbf{I}_{G,k-1}\}$ can be constructed.

Afterwards, Y channel of the cover image, denoted as $\mathbf{Y}_C$, is extracted and transformed into frequency domain using Discrete Fourier Transform (DFT). Denote the Fourier coefficients of $\mathbf{Y}_C$ as $\mathbf{DFT}_C$. Obviously, the size of $\mathbf{DFT}_C$ is also $m_r \times m_c$. Also, the Y channel of every $\mathbf{I}_{G,i} \in \mathbf{K}$ is extracted and denoted as $\mathbf{Y}_{G,i}$. $\mathbf{Y}_{G,i}$ is transformed into frequency domain using DFT to construct its Fourier coefficients $\mathbf{DFT}_{G,i}$. The collection of the transformed Fourier coefficients of $\mathbf{K}$ is denoted as $\mathbf{C}_G = \{\mathbf{DFT}_{G,1}, \dots, \mathbf{DFT}_{G,k-1}\}$. The experiments show that the correlation of $\mathbf{DFT}_{G,i}$ between different patterns follows the same rule of (2).

$$\left|Cor\left(\mathbf{DFT}_{G,i}, \mathbf{DFT}_{G,j}\right)\right| \approx \begin{cases} 0, i \neq j \\ 1, i = j \end{cases} \tag{3}$$

Next, $\mathbf{DFT}_C$, together with $\mathbf{DFT}_{G,i} \in \mathbf{C}_G$, is divided into n rings. Let $\mathbf{R}_C^x$, $\mathbf{R}_{G,i}^x$ respectively be a set of the coefficients in the x_{th} ring of $\mathbf{DFT}_C$ and $\mathbf{DFT}_{G,i}$. The ring partition of the image is done by calculating the distance between each coefficient and the geographical center. Denote the radius of the n circles as $\mathbf{r} = \{r_0, r_1, \dots, r_{n-1}\}$, where r_0 and r_{n-1} are respectively the radius of the innermost and outmost circles, and $r_{n-1} = \lfloor \min(m_r, m_c) \rfloor$. Here $\lfloor a \rfloor$ stands for taking the floor integer of a, and $\min(a, b)$ is the operation of taking the minimum. We denote the area of the x_{th} circle and the x_{th} ring as S_x, μ_x, respectively. The numbers of coefficients in every ring are expected to be roughly the same. Therefore, we let $S_x = \pi {r_x}^2$, and $\mu_x = \lfloor S_x / x \rfloor$. And thus, r_x is computed by

$$r_x = \begin{cases} \sqrt{\frac{\mu_x}{\pi}}, & x = 0 \\ \sqrt{\frac{\mu_x + \pi {r_{x-1}}^2}{\pi}}, & x \neq 0 \end{cases} \tag{5}$$

Let $p(a, b)$ be the coefficient of $\mathbf{DFT}_C$ in the a_{th} row and b_{th} column. The distance between $p(a, b)$ and the image center can be calculated by the Euclidean distance.

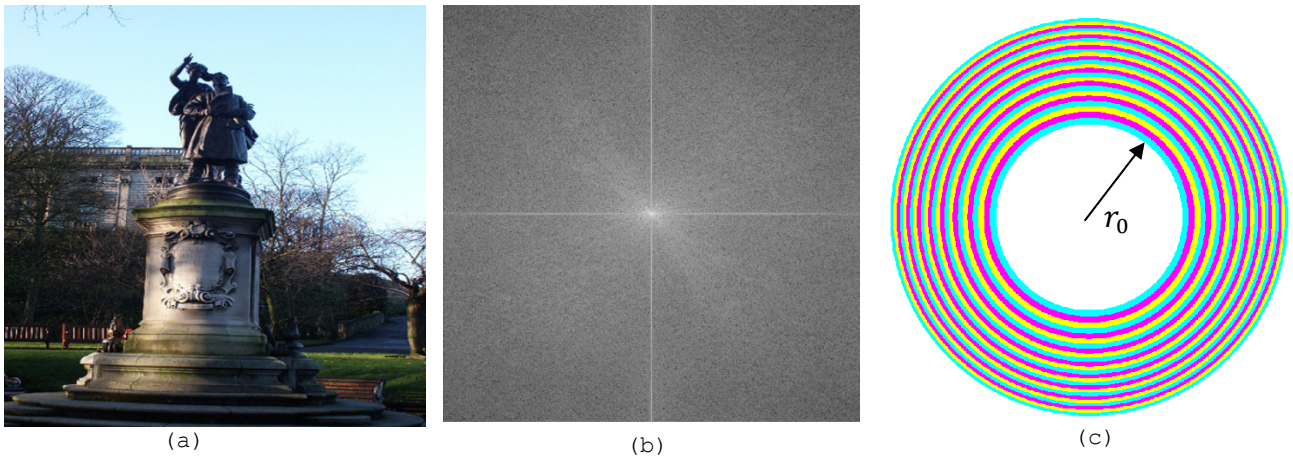

Figure 2. Illustration of DFT coefficients and Ring Partitioning.

$$d_{a,b} = \sqrt{(a - m_r/2)^2 + (b - m_c/2)^2} \tag{6}$$

According to the calculated radius of the rings and the distances, we classify the coefficients into n rings, where $\mathbf{R}_C^x$ is a collection of coefficients defined as (7)

$$\mathbf{R}_C^x = \{p(a,b) | r_{x-1} \le d_{a,b} \le r_x\} \tag{7}$$

Then, according to the same principle, $\mathbf{R}_{G,i}^x$ can also be calculated. Afterwards, the watermark sequence is represented as the selection of $\mathbf{R}_{G,i}^x$ in every ring, and the selected pattern is then added on $\mathbf{R}_C^x$ for watermark embedding. If b bits of additional data are to be embedded in each ring, we should select 2^b different pseudo-random seeds. That means the relationship between b and k is also $b = log_2 k$. Also, since the coefficients inside the innermost ring represent the low frequency of the images, data embedding in this ring will cause great modification. Thus, we do not hide bits into this ring. Therefore, a total of $b \times (n\text{-}1)$ bits can be embedded in the Y component of the cover image. Denote the code word sequence as $= \{m_1, m_2, \ldots, m_{b-1}\}$. Then the x_{th} ring of the Y channel in the watermarked image $\mathbf{Y}_O$, denoted as $\mathbf{R}_O^x$ can be calculated by (8).

$$\mathbf{R}_O^x = \begin{cases} \mathbf{R}_c^x + \boldsymbol{\alpha}_x \mathbf{R}_{G,0}^x, & \text{embed } m_0 \\ \mathbf{R}_c^x + \boldsymbol{\alpha}_x \mathbf{R}_{G,1}^x, & \text{embed } m_1 \\ \quad \ldots \ldots & \\ \mathbf{R}_c^x + \boldsymbol{\alpha}_x \mathbf{R}_{G,k-1}^x, & \text{embed } m_{b-1} \end{cases} \tag{8}$$

In (8), the parameter $\boldsymbol{\alpha}_x$ is a sequence of parameters that controls the embedding weights. A larger $\boldsymbol{\alpha}_x$ means a larger embedding strength in the ring, which generally leads to a heavier distortion in the relevant coefficients in the frequency domain. In this paper, we set

$$\boldsymbol{\alpha}_x = k \cdot \frac{E(|\mathbf{R}_c^x|)}{E(|\mathbf{R}_{G,i}^x|)} \tag{9}$$

where $E(|.|)$ is used to calculate the energy of the signal, and k is empirically set as 1/3 to decrease the impact caused by watermarking. Thus, the energy of watermark is always k times

lower than that of $\mathbf{Y}_C$. Afterwards, the invert DFT is applied on $\mathbf{DFT}_O$ and we get the Y component of marked image as $\mathbf{Y}_M$. In order to detect the watermark, the private keys $\{S_0, S_1, \ldots, S_{k-1}\}$ should be known by the recipient or transmitted under a covert communicating channel.

Figure 2(a) provides the cover image "Sculpture". Figure 2(b) shows the amplitude spectrum of the DFT coefficients of Figure 2(a). The larger values are shown by lighter colors and zeros are shown by extreme black. The center represents the low frequency of the cover image. The illustration of ring partition is shown in Figure 2(c).

The embedding rate of the proposed framework is flexible. Generally, the payload increases when more rings and seeds are used. However, the robustness against common attacks might be weaker.

2.2. Embedding Rectification Information

For geometric distortion attacks might cause the problem of de-synchronization, we further propose to embed image rectification information in the U channel using DTCWT, which can be used to identify geometric distortions and invert the detected attacks ahead of watermark extraction in Y channel.

DTCWT is composed of two discrete wavelet transforms. The real tree applies the low-pass filters and imaginary tree applies the high-pass filters. DTCWT has the advantages of approximate shift invariance, good directional selectivity. A 3-level DT-CWT decomposes an image sized $m_r \times m_c$ into 3 scales, where each scale is of size $m_r/2^{s-1} * m_c/2^{s-1}$. Here, s=1, 2, 3. Six complex coefficients can be calculated from each decimated subband, which is denoted as $\{P_H^{l,1}, P_H^{l,2}, P_H^{l,3}, P_H^{l,4}, P_H^{l,5}, P_H^{l,6}\}$, where l represents the level and for each j, $P_H^j = \rho_1 e^{i\theta_j}$. Here, the six coefficients are correspondent with the 6 subbands' orientations, and ρ and θ respectively represent the amplitude and the angle of every coefficient. In the proposed scheme, we conduct 3-level DTCWT decomposition, which produces 2 low frequency and 18 high frequent subbands. Detailed implementation of DTCWT is omitted in this paper and can be found in relevant papers.

Denote the U component of the cover image as $\mathbf{U}_C$. First, we conduct the 3-level DTCWT to $\mathbf{U}_C$, and we obtain the 6 high-frequent subband in the 3-level denoted as $\mathbf{P}=\{P_C^{3,1}, P_C^{3,2}, P_C^{3,3}, P_C^{3,4}, P_C^{3,5}, P_C^{3,6}\}$. We further select a private key S_p and generate a random sequence using the specified seed. Next, an image pattern $\mathbf{I}_{G,i}$ is generated using the same method as described in Section 2.1. The size of $\mathbf{I}_{G,i}$ is the same as the cover image. We obtain the U component of $\mathbf{I}_{G,i}$ as $\mathbf{U}_{G,i}$ and conduct on it the 3-level DTCWT. The 6 high-frequent subband of the pattern are $\{P_G^{3,1}, P_G^{3,2}, P_G^{3,3}, P_G^{3,4}, P_G^{3,5}, P_G^{3,6}\}$. The detecting watermark is embedded by linearly adding the corresponding 3-level subbands of $\mathbf{U}_{G,i}$ onto those of $\mathbf{U}_C$. Denote the new subbands as $\{P_M^{3,1}, P_M^{3,2}, P_M^{3,3}, P_M^{3,4}, P_M^{3,5}, P_M^{3,6}\}$. Then, $P_M^{3,k} = P_C^{3,k} + \boldsymbol{\alpha}_x P_G^{3,k}$, where k=1, …, 6. The definition of $\boldsymbol{\alpha}_x$ here is similar to that in (9) that lets the average amplitude of the side information k times lower than that of the original signal.

Afterwards, we get the watermarked U component of $\mathbf{I}_M$ by conducting the invert DTCWT. For accurate watermark detection, S_p should also be acquired by the recipient.

2.3. *Watermark Extraction*

When the recipient gets the doubted image $\mathbf{I}_D$, the hidden watermark can be detected and extracted. The overall flowchart of the extracting procedure is depicted in Figure 3. Note that the

recipient is convinced to know the private keys used in watermark embedding for the generation of random sequences, otherwise he cannot perform the watermark extraction. The extracting process contains two parts: watermark detection & image rectification in U channel, and watermark extraction in Y channel. The image is first decomposed into the three components $\{\mathbf{Y}_D, \mathbf{U}_D, \mathbf{V}_D\}$.

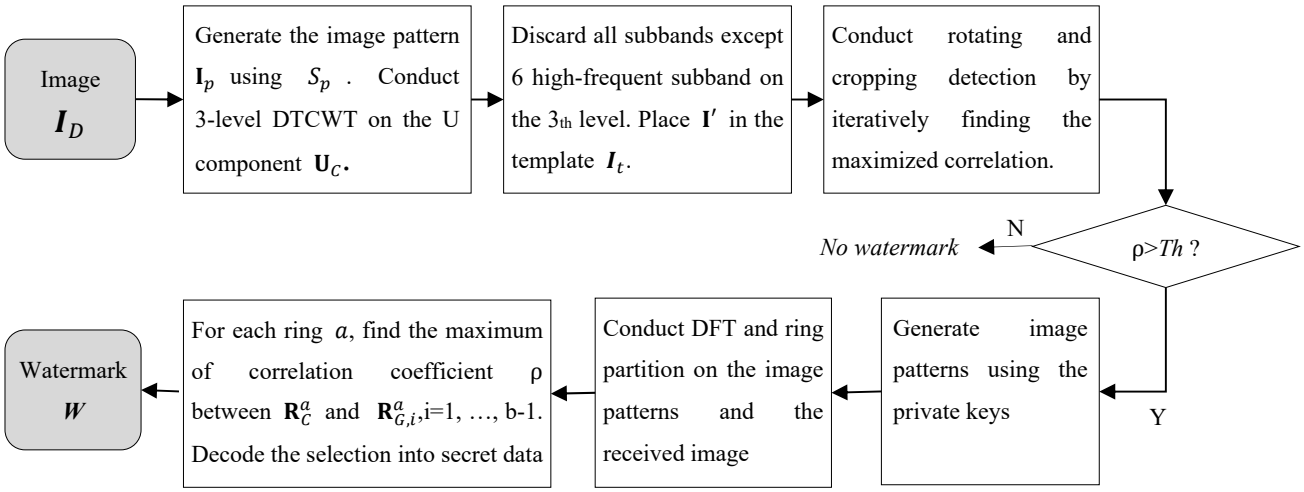

Figure 3. Flowchart of the extracting procedure.

To conduct watermark detection & image rectification in U channel, the recipient firstly conducts the 3-level DTCWT on $\mathbf{U}_D$. Then, he reserves the 6 high-frequent subband on the 3th level and lets all the coefficients in the rest of subbands equal to zero. Thereafter, by conducting the invert 3-level DTCWT, $\mathbf{U}_D'$ can be obtained which only contains the high frequent information of $\mathbf{U}_D$. Afterwards, using S_p, the recipient generates the image pattern $\mathbf{I}_p$. Also, an empty image template $\mathbf{I}_t$ sized $m_r \times m_c$ is formed to realize the image rectification.

For cropping detection, the recipient places $\mathbf{U}_D'$ at the top-left corner of the template to get the detecting model $w_c = \mathbf{I}_t + \mathbf{U}_D'$, and use the sliding-window method to find the actual location of the cropped image. Each time he moves rightward $\mathbf{U}_D'$ by γ pixels in the template, and $\mathbf{U}_D'$ is placed at the beginning of the row γ downward when it reaches the right boundary. A larger γ results in a lower computational complexity, but the accuracy may be lower too.

The recipient calculates the convolution of the model and $\mathbf{I}_p$ and get the result matrix $\mathbf{M}_{cov}$. Denote the maximum value in $\mathbf{M}_{cov}$ as $\mathbf{M}_{cov}(x'_d, y'_d)$, and therefore, the actual location the actual location of the cropped image can be found as (x'_d, y'_d).

For rotation detection, the recipient needs to check the correlation between the model and $\mathbf{I}_p$ with several different rotating angle. He determines the actual rotating angle as the angle with which the correlation coefficient between the model and $\mathbf{I}_p$ is the largest.

$$(x'_d, y'_d) = \underset{x,y}{\operatorname{argmax}}(\operatorname{cor}(\mathbf{I}_p, w_c)),\ \theta_1 = \max(\operatorname{cor}(\mathbf{I}_p, w_c)) \tag{10}$$

Here, $\max(x)$ represents taking the maximum among x, and θ_1 is the maximized correlation for location rectification. Note that the cropping detection can be conducted also by observing the

correlation between the template and each portion of the received image using the sliding window technique. However, the computational complexity of this method is much higher.

The recipient can also conduct the rotation detection by a similar method. The recipient finds the rotating angle α by iteratively finding the maximized correlation θ_2 for each rotating angle.

Afterwards, watermark in $\boldsymbol{I}_D$ can be determined by checking $\max(\theta_1, \theta_2) > Th$, in which Th is a pre-defined threshold which is normally set as 0.5. If $Th > 0.5$, the recipient rectifies $\boldsymbol{I}_D$ according to the detected position and rotating angel and moves on to extract the watermark hidden in Y channel. Otherwise he does not consider the image contains secret watermark.

Figure 4 provides a sketch of image rectification procedures for cropping and rotating detection. The order of these two kinds of detection is flexible. The recipient should try the cropping detection ahead of rotating detection, if the reversed order does not show that there is a watermark.

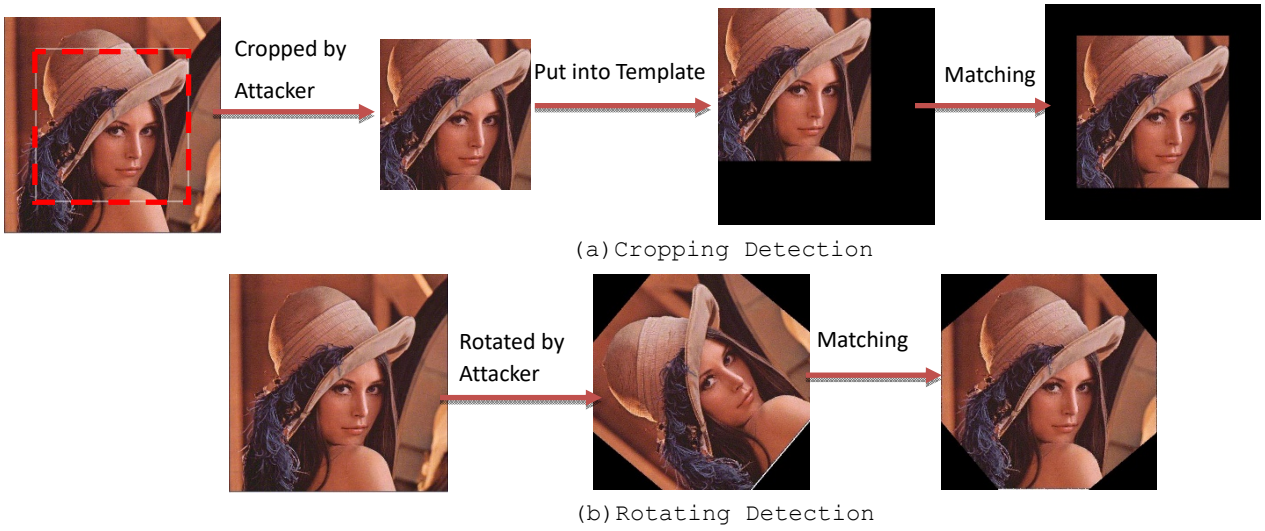

Figure 4. A sketch of image rectification procedure.

To extract the watermark hidden in the Y component, the recipient conducts DFT on $\mathbf{Y}_D$ to get $\mathbf{DFT}_D$. Then, the coefficients are further partitioned as $\{\mathbf{R}_m^0, \mathbf{R}_m^1, \ldots, \mathbf{R}_m^{k-1}\}$ using (7). Also, using the k private keys, the recipient generates the image patterns $\{\mathbf{I}_{G,0}, \mathbf{I}_{G,1}, \ldots, \mathbf{I}_{G,k-1}\}$, and DFT is applied respectively to obtain $\mathbf{C}_G$=$\{\mathbf{DFT}_{G,0}, \ldots, \mathbf{DFT}_{G,k-1}\}$. In each ring except the innermost, the recipient spares the coefficients inside the ring of and set the other coefficients as zero. Then, he conducts the invert DFT on $\mathbf{DFT}_D$ to get the spatial domain images $\mathbf{I}'_D$ and $\mathbf{C}'_G$=$\{\mathbf{I}'_{G,1}, \ldots, \mathbf{I}'_{G,k-1}\}$. Then he calculates the correlation between $\mathbf{I}'_D$ and every image pattern in $\mathbf{C}'_G$. The secret bits are extracted by decoding into the relevant k-bit code word the index of the pseudo-random image which has the maximized correlation. Finally, the recipient gets the watermark by concatenating all secret bits.

3. Experimental results

To verify the proposed method, we have conducted many experiments on thousands of colored images in some popular databases and typical testing images such as “Lenna”. We use binary random sequences as digital watermark, i.e., the possibilities for 0 and 1 are close. We compare the proposed

watermarking scheme with some state-of-the-art methods proposed in [9] and [12–14] that also simultaneously have the robustness against common attacks mentioned above. Note that the method in [9] is compared by extending it in the application of colored image by conducting the watermarking scheme on the U component while leaving Y and V components unchanged. The term "QF" in Table 1 refers to the quality factor in JPEG compression. We test the watermark robustness to several attacks, including JPEG compression, Gaussian filtering, contrast enhancement (using the histogram equalization function in MATLAB), cropping, scaling, and rotating.

(a) Lenna (b) Baboon

Figure 5. Embedding test for typical images.

Table 1. Number of error bits under different attacks.

Type of attack		Error bits (Figure 5(a))	Error bits (Figure 5(b))
No Attack		0	0
JPEG Compression	QF: 60	0	0
	QF: 20	5	4
Adding Gaussian Noise (σ = 20)		1	0
Median Filtering		3	2
Image Contrast Enhancement		5	3
Cropping	70% Area Left	2	2
	90% Area Left	0	0
Scaling	70%	2	2
	150%	2	2
Rotating	30 degrees	3	2
	10 degrees	0	0

3.1. Settings and evaluations

The main parameters in the proposed framework are the number of rings in the ring partition and the number of seeds used in generating pseudo-random images. The experiments are mainly conducted by partitioning the cover image into 25 rings, and we choose totally 17 different pseudo-random seeds, 16 of which is used for watermark embedding in the Y component. And thus, the embedding rate in our experiments is fixed as $25 \times 4 = 100$ bits. In practice, other settings that provide different bit rate can also ensure a satisfying embedding performance. The data hider can

also hide 64 bits into the cover image in [9], 128 bits in [12–14].

Averagely, the watermark embedding and extraction can be done within several seconds by a personal laptop with 2.60 GHz CPU and 8.00 GB RAM. The computational complexity is mainly on the receiver's side, since the receiver needs to conduct the image rectification ahead of watermark extraction. Figure 5 offers the embedding test of the proposed watermarking scheme, and Table 1 shows the error bits after extracting the watermark from the received images that have gone through different kinds of attacks. In the two sets of images in Figure 5, the left represents the original cover images and the right represents the watermarked images. No obvious visual distortion between the two sets of images can be detected. In the table, we can easily find that no more than 5 out of 100 error bits occurs during the watermark extraction of the two images. It proves that the proposed method has high robustness to common attacks.

Table 2. Comparisons of SSIM and PSNR for different methods.

Method	Proposed	[9]	[12]	[13]	[14]
SSIM	0.9623	0.9654	0.9458	0.9501	0.9566
PSNR	40.2317	42.2316	37.6767	38.4975	40.0746

For an objective image quality assessment of the proposed framework, we employ the Structural Similarity (SSIM) [17] and the peak signal-to-noise ratio (PSNR) to measure the consistent and subjective image visual quality of the marked images. While PSNR is a well-known index that measures the distortion between two images, SSIM is proposed generally based on the degradation of structural information. The value of SSIM ranges from 0 to 1, and a higher structure similarity between the cover image and the marked image is indicated by a high SSIM closer to 1. Table 2 shows the average SSIM and PSNR value of the watermarked images produced by different algorithms. The proposed framework generally ranks high in the two indices, which indicate a high visual quality and fidelity to the original images.

3.2. Embedding performances

In Table 3, we have added some kinds of non-geographical attacks onto the marked images and conduct similar comparisons. From the table it is obvious that the average amount of error bits of the proposed scheme is lower than those of the rest methods. The reason behind a better performance is that we can successfully rectify the image at the early stage using DTCWT. Such advantage is not reachable by the rest comparing methods.

Table 3. Extracted watermark images under other non-geographical attacks.

Type of attack	Amount of error bits				
	Proposed	[9]	[12]	[13]	[14]
AWGN (σ = 20)	0.277/100	1.21/64	0.9980/128	0.449/128	0.513/128
S&P (p = 0.05)	0.139/100	1.45/64	0.5516/128	0.768/128	0.640/128
Median filter 4 × 4	0.123/100	2.46/64	1.1427/128	7.16/128	0.569/128

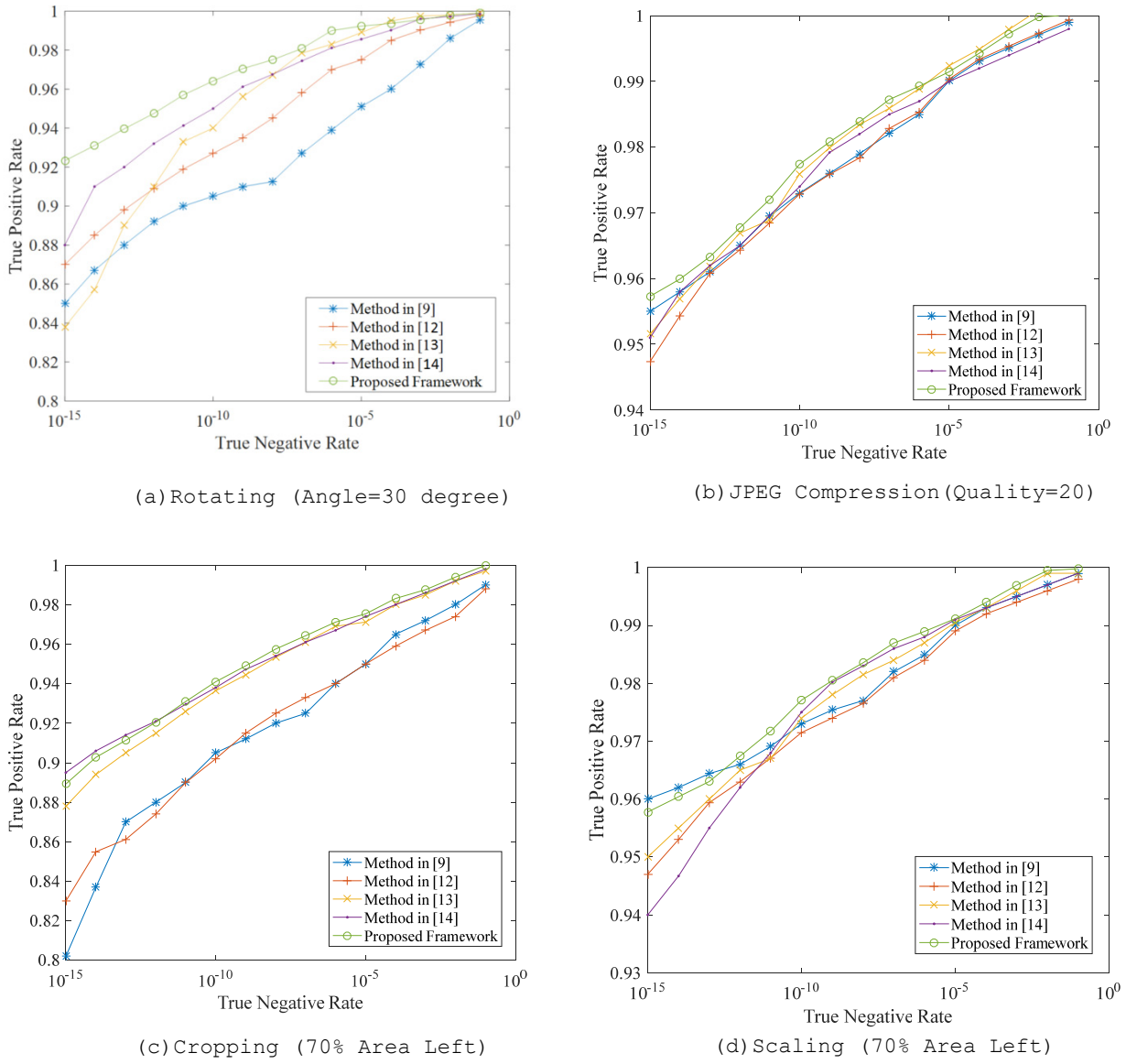

Figure 6. Comparison of ROC curves under different attacks.

In Figure 6, we compare with [9] and [12–14] the receiver operating characteristic (ROC) curves prior to rotation, JPEG compression, cropping and scaling. The curves are plotted by doing the watermark embedding and extraction test on randomly-selected 500 colored images from the typical image databases UCID [18] and OSU [19]. From the curves, we find that the true positive rate of the proposed method is generally the closest to 1, even if the attacks are strong. The high robustness against rotation attack is that rotating operations generally take the geographical center of the image as origin. Thus, the pixels of the original image and those of rotated image are usually counted in a same ring, letting the amplitude of the DFT coefficients mostly unchanged. The robustness against JPEG compression and scaling lies in a similar reason. The proposed method in [12–14] generally gives a close result to our method. However, the proposed framework can reach ROC = 1

much earlier judging from the four figures. It can also be observed that the proposed method excels a lot in anti-JPEG compressing and anti-rotating watermarking. While the performance of anti-cropping watermarking is not as good as that, it still succeeds in having a close or slightly higher value of ROC. Thus, the proposed method succeeds in out-performing these state-of-the-art methods.

For a brief evaluation of the security of our data hiding framework, we also include the popular SPAM feature set proposed by Penvy et al. [20]. The proposed steganalysis algorithm uses first-order and second-order Markov chains to model the value differences between neighboring pixels. Same as other typical steganalysis algorithms, SPAM first extracts the local features of the cover images and the marked images. Then the features are submitted to the ensemble classifier proposed in [21] for a classified detection. For space limit, we refer readers to acquire the detailed implementation of SPAM feature in [20].

We use the second-order SPAM features that contains 686 features. The error rate of SPAM detection is calculated to evaluate the security of the proposed framework. The error rate P_E is defined in (11).

$$P_E = \frac{1}{2}(P_{FP} + P_{FN}) \tag{11}$$

Here, P_{FP} and P_{FN} respectively mean the false alarm rate and false denying rate. A higher P_E of a data hiding method usually indicates a better capability in data secrecy. P_E is always positive and cannot exceed 0.5. Generally, the security will be weaker against the SPAM detection if the watermark embedding strength is stronger.

Table 4 shows the classification error of the steganalyzers of the proposed framework. As can be easily observed, the proposed method generally provides high information security. The error detecting rate is close to 50% for the proposed framework. It indicates the proposed framework can evade the steganalysis detection easily, ensuring the data secrecy. For comparison, the abilities of anti-SPAM for the proposed framework, [9] and [12] are much close. It is worth notice that, although the algorithm proposed in [13] and [14] offers a close embedding performance compared to ours, the anti-SPAM capabilities of the two scheme are much lower. Thus, the proposed framework also succeeds in outperforming these methods in data secrecy.

To give a concise conclusion, our method succeeds in providing strong robustness against common attacks, even if the attacks might be strong. Also, the framework is much comparable in terms of security analysis.

Table 4. Comparison of the Error Rate Obtained By SPAM.

Method	Proposed	[9]	[12]	[13]	[14]	Database
Classification Error ($\boldsymbol{P_E}$)	48.97%	49.63%	49.01%	45.52%	44.87%	UCID
	49.12%	49.57%	48.79%	46.63%	41.58%	OSU

4. Conclusion

In this paper, we propose a new framework of robust digital watermarking for colored images. The framework is resilient to common attacks such as scaling, JPEG compression, cropping, etc. Y

and U channels of the cover image are utilized for watermark embedding and image rectification. Using private keys as seeds, the data hider generates image patterns, and conducts the ring partition and DFT on the patterns and the original image. The watermark sequence is represented as the selection of the private keys, and for each ring, the DFT coefficients the selected pattern are added of onto those of the cover image. To make the watermark resilient to geographic distortion, another image pattern is generated and embedded into U channel of the cover image using DTCWT. On the recipient's side, the received image is firstly rectified according to the side information in U channel, and the watermark can be extracted thereafter. We conduct the embedding test under 100-bit payload capacity, and the experimental results prove that the proposed method is better in preserving image quality as well as providing strong resilience to common attacks.

Acknowledgments

This work is supported by the National Nature Science Foundation of China (Grant 61572308, and Grant U1736213), the Shanghai Excellent Academic Leader Plan (16XD1401200), and the National Key Research Development Program of China (2016QY01W0200). We thank the anonymous reviewers for their constructive suggestions. We also thank Dr. Shuozhong Wang for his kind help in this work.

Conflict of interest

The authors declare no conflict of interest.

References

1. H. Larijani and G. Rad, A new spatial domain algorithm for gray scale images watermarking, *International Conference on Computer and Communication Engineering*, (2008), 157–161.
2. H. Nemade and V. Kelkar, Reversible watermarking for colored medical images using histogram shifting method, *3rd International Conference on Computing for Sustainable Global Development*, (2016), 2664–2668.
3. A. Al-Nu'aimi and R. Qahwaji, Adaptive watermarking for digital colored images based on the energey of edges, *IEEE International Conference on Signal Processing and Communications*, (2007), 1371–1374.
4. J. O'Ruanaidh and T. Pun, Rotation, scale, and translation invariant digital image watermarking, *Sign. Process.*, **66** (1998), 303–317.
5. F. N. Thakkar and V. K. Srivastava, A blind medical image watermarking: DWT-SVD based robust and secure approach for telemedicine applications. *Multimed. Tools Appl.,* **76** (2017), 3669–3697.
6. H. Hu and H. Lyu, Collective blind image watermarking in DWT-DCT domain with adaptive embedding strength governed by quality metrics, *Multimed. Tools Appl.,* **76** (2017), 6575–6594.
7. S. Horng, A blind image copyright protection scheme for e-government. *J. Vis. Communic. Image*, **24** (2013), 1099–1105.
8. S. A. Parah, J. A. Sheikh, N. A. Loan, et al., Robust and blind watermarking technique in DCT domain using inter-block coefficient differencing, *Dig. Signal Process.*, **53** (2016), 11–24.

9. X. Kang, J. Huang and W. Zeng, Efficient general print-scanning resilient data hiding based on uniform log-polar mapping, *IEEE T. Inf. Foren. Sec.*, **5** (2010), 1–12.
10. D. Zheng, J. Zhao and A. El. Saddik, RST invariant digital image watermarking based on log-polar mapping and phase correlation, *IEEE T. Circ. Syst. Vid.*, **13** (2003), 753–765.
11. P. Niu, X. Wang, H. Jin, et al., A feature-based robust digital image watermarking scheme using Bandelet transform, *Optics Laser Tech.*, **43** (2011), 437–450.
12. M. Amini, M. O. Ahmad and M. N. S. Swamy, A robust multibit multiplicative watermark decoder using vector-based hidden Markov model in wavelet domain, *IEEE T. Circ. Syst. Vid.*, **28** (2018), 402–413.
13. M. Amini, M. O. Ahmad and M. N. S. Swamy, Digital watermark extraction in wavelet domain using hidden Markov model, *Multimed. Tools Appl.*, **76** (2017), 3731–3749.
14. E. Nezhadarya, Z. J. Wang and R. K. Ward, Robust image watermarking based on multiscale gradient direction quantization, *IEEE T. Inf. Foren. Sec.*, **6** (2011), 1200–1213.
15. V Ananthaneni, U. R.Nelakuditi, Hybrid digital image watermarking using contourlet transform (CT), DCT and SVD. *Int. J. Image Process.*, **11** (2017), 85–93.
16. A. K. Singh, M. Dave and A. Mohan, Hybrid technique for robust and imperceptible image watermarking in DWT–DCT–SVD Domain. *Natl. Acad. Sci. Lett.,* **37** (2014) 351–358.
17. Z. Wang, A. C. Bovik, H. R. Sheik, et al., Image quality assessment: from error visibility to structural similarity, *IEEE T. Image Process.*, **13** (2004), 600–612.
18. UCID—an Uncompressed Colour Image Database–a benchmark database for image retrieval with predefined ground truth, Accessed: 2019. (available: http://imagedatabase.cs.washington.edu/groundtruth/)
19. T. Lin, M. Maire, S. Belongie, et al., Microsoft COCO: Common objects in context, *European Conference on Computer Vision*, (2014), 740–755, Accessed: 2019. (available: http://cocodataset.org/#home)
20. T. Pevny, P. Bas, and J. Fridrich, Steganalysis by subtractive pixel adjacency matrix, *IEEE T. Inf. Foren. Sec.*, **5** (2010), 215–224.
21. J. Kodovsky, J. Fridrich and V. Holub, Ensemble classifiers for steganalysis of digital media, *IEEE T. Inf. Foren. Sec.*, **7** (2012), 432–444.

AIMS Press