symmetry

MDPI

Article

Blind Image Watermarking in Canonical and Cepstrum Domains Based on 4-Connected t-o'clock Scrambling

Farhana Shirin Chowdhury [1], Pranab Kumar Dhar [1], Kaushik Deb [1,*] and Takeshi Koshiba [2]

[1] Department of Computer Science and Engineering, Chittagong University of Engineering & Technology (CUET), Chattogram 4349, Bangladesh; fshirin2007@gmail.com (F.S.C.); pranabdhar81@cuet.ac.bd (P.K.D.)
[2] Faculty of Education and Integrated Arts and Sciences, Waseda University, 1-6-1 Nishiwaseda, Shinjuku-ku, Tokyo 169-8050, Japan; tkoshiba@waseda.jp
* Correspondence: debkaushik99@cuet.ac.bd

Received: 21 December 2019; Accepted: 31 January 2020; Published: 9 February 2020

Abstract: Copyright protection of multimedia content is confronted with great challenges such as easy access to the Internet. Digital watermarking is widely applicable technique for copyright protection of multimedia contents. In this paper, a blind symmetric watermarking method in canonical and cepstrum domains based on four-connected t-o'clock scrambling is proposed. Initially, the watermark image is scrambled using the four-connected t-o'clock method to enhance the security. Then, the rotation operation is applied to the host image to extract the region where the watermark bits are embedded. After that, discrete linear canonical transform (DLCT) is applied to the extracted region to obtain the DLCT region. Cepstrum transform (CT) is performed on DLCT region to attain CT region. The CT region is then divided into non-overlapping blocks. The watermark bits are inserted into each block using max-heap and min-heap tree property. Experimental results illustrate that the proposed method shows high robustness against numerous attacks. Moreover, it produces high quality watermarked images and provides high security. Furthermore, it has superior performance to recent methods in terms of imperceptibility, robustness, and security.

Keywords: robustness; discrete linear canonical transform; cepstrum transform; four-connected t-o'clock; security; scrambling

1. Introduction

In today's ever-changing world, Internet technology has become an imminent part of our day-to-day life. With the benefits of the Internet, we can easily copy, transmit, distribute, and store information [1]. However, unfortunately, at the same time, unauthorized usage of personal media content without the permission of owners is also increasing rapidly. Different types of online business sectors face a huge loss every year due to copyright violations. To solve this problem, in recent years, a new form of media content protection has evolved, known as watermarking. A watermark is a piece of information that is embedded into multimedia data such as text, audio, image, and video by the owner to ensure the authenticity of the data. An efficient and effective watermarking method should satisfy three common fundamental requirements: robustness, imperceptibility, and security. First, the watermark image should resist against various attacks. This feature is called robustness. Second, visually, there should be no difference between host and watermarked images. This property is known as imperceptibility [1]. Finally, the scrambling method is used to shuffle the watermark information before embedding, thereby ensuring the security of the watermark image against numerous attacks. Watermarking can be classified into spatial domain and transform domain

techniques. Spatial domain techniques embed a watermark of a given image by modifying its pixels directly. This technique is easy to implement and requires few computational resources [2,3]. On the other hand, the transform domain technique is applied to coefficients obtained as the result of a frequency transform of either a whole image or single block-shaped regions of a frame. Discrete cosine transform (DCT) [4–9], discrete Fourier transform (DFT) [10,11] and discrete wavelet transform (DWT) [12–16] are commonly used frequency-domain techniques. Recently, some watermarking methods have been proposed that use various decomposition and transform techniques jointly [17–24].

Yashar et al. [2] suggested a blind gray-scale image watermarking method based on the QR decomposition. In this method, the watermark bit is embedded into the R matrix; however, the imperceptibility is not quite satisfactory. Pizzolante et al. [3] introduced a novel method that can embed two watermarks into a con-focal 3-D microscopy image; however, the PSNR of this method is not high and the robustness has not been evaluated.

Shuai et al. [4] suggested a double encryption method, in which the watermark is converted into fractal encoding and then embedded into DCT transformed carrier image; however, their method was only tested against three attacks and the experimental result show that their method do not perform well against Gaussian noise attack. Jingyou et al. [5] proposed a bimodal structure and iterative selection method in the DCT domain. However, in this method, the PSNR is low. Shabir et al. [6] proposed a blind watermarking technique in the DCT domain, in which the watermark bit is embedded in the middle band frequency based on coefficient differences in the same position of succeeding blocks. The main disadvantage of this method is that its peak signal to noise ratio (PSNR) is quite low. In [7], an image watermarking technique using Redundant DWT (RDWT) and DCT domain is proposed, where the binary image is used as a watermark. However, the computational complexity of this method is quite high. Soumitra et al. [8] introduced a blind digital watermarking in the DCT domain, where multiple watermark images are embedded in the middle band frequency of the host image. However, the PSNR of this method is not high. In [9], a color image watermarking method based on DCT and DWT is introduced. In this method, the scrambled watermark image is transformed into the DCT domain. After that, transformed watermark information is embedded into four sub-bands region obtained from DWT. The main disadvantage of this method is that it does not provide a good trade-off between robustness and imperceptibility.

In [10], a DFT-DCT based hybrid image watermarking technique is presented. The watermark bit is scrambled using Arnold transform and is later embedded into middle sub-band frequency. Their method performed well for both imperceptibility and robustness. The main drawback of their method is that it has been tested against very few attacks, such as histogram equalization, JPEG compression, salt and pepper noise, Gaussian noise, etc. In [11], a DFT and two-dimensional (2D) histogram based hybrid image watermarking is proposed. However, the detector is unable to detect the true watermark image without knowing about the attacks.

In [12], the authors proposed an image watermarking scheme in the DWT domain. In this paper, the watermark bit is embedded into the coefficients obtained from the three high-frequency sub-bands of first level decomposition but the image quality is degraded with the higher quantization steps. In addition, the authors did not conduct the imperceptibility test. Jinyuan et al. [13] proposed a digital image watermarking algorithm in the DWT domain, in which the watermark image is scrambled using the logistic map. Then, the watermark bit is embedded into the multilevel DWT coefficients. However, the PSNR of this method is not high and the trade-off between robustness and imperceptibility is not satisfactory. In [14], the DWT based image watermarking is proposed, where the first-level DWT is applied to a watermark image before embedding. The main flaw of this method is that the host image is considered as the watermark image, which violates the basic requirement of image watermarking. Asma et al. [15] suggested a DWT-based method in which watermark bit is embedded into the low-frequency band using alpha blending. However, this method is least robust against Gaussian noise. Ravi et al. [16] introduced a watermarking scheme using DWT and DCT

along with Arnold transform, where watermark bit is embedded into a low frequency (LL) band. However, this method has low robustness against cropping attacks.

Qingtang et al. [17] proposed a color image watermarking method, where the watermark information is embedded into the largest eigenvalue of the Schur decomposed matrix. However, the PSNR of this method is not high and it does not perform well against the cropping attack. Radu et al. [18] introduced a watermarking method that uses chrip z-transform, DWT, and singular value decomposition (SVD). The disadvantage of this method is that it shows low robustness against the JPEG compression attack. An image watermarking method using DWT and shuffled SVD (SSVD) is suggested in [19]. In this method, the authors overcame the false positive problem but the PSNR value of this method is not good. In addition, this method is vulnerable against some attacks such as salt and pepper, gamma correction, image sharpening, image cropping, etc. Llukman et al. [20] presented a hybrid method using RDWT and SVD in which Arnold transform is used to scramble the watermark image to enhance the security. However, the PSNR of the watermarked images is not satisfactory. Yuqi et al. [21] presented an image watermarking method using DWT, DCT, and SVD. It embeds watermark bits into the singular values of DCT coefficients obtained from the DWT sub-band. However, the robustness result is not good against some attacks such as JPEG compression and low pass filtering. In [22], an image watermarking method is proposed that can be robust against different types of geometric attacks. However, this method has low robustness against scaling attack.

In [23], a different type of image watermarking method in the angular radial transform (ART) domain is proposed, in which the watermark bit is embedded in the geometric invariant domain. While these methods are efficient under the scaling or rotation attacks, the performance is still drastically degraded against other geometric attacks such as cropping. Qingtang et al. [24] suggested a color image watermarking method using LU decomposition, where the watermark information is embedded into the first elements of both second and third rows of the lower triangular matrix. It uses Arnold transform to enhance the watermark security and the pseudo-random MD5 hash function is applied to increase the watermark robustness. However, the PSNR of this method is quite low.

Image scrambling is one of the main features in image watermarking that removes the correlation between pixels of a given image. As a result, the image becomes an insignificant image and can resist malicious attacks to a certain extent [25]. Arnold transform [25,26] is a widely used image scrambling technique. Even though the modified version [27] is free from periodicity, it still needs a lot of information such as block sequences, block size of the image, etc. while unscrambling the image.

The main limitations of the existing methods are the difficulty in maintaining a good trade-off among robustness, imperceptibility, and security. Moreover, some methods have low robustness, whereas some are less imperceptible or less secured. To overcome these limitations, we propose a blind symmetric image watermarking scheme in canonical and cepstrum domains based on the four-connected t-o'clock scrambling method. To the best of our knowledge, this is the first image watermarking method that utilizes discrete linear canonical transform (DLCT), cepstrum transform (CT), and four-connected t-o'clock scrambling technique jointly. The main characteristics of the proposed method are: (i) it applies DLCT and CT jointly; (ii) it utilizes four-connected t-o'clock scrambling method to enhance the security of the watermark image; (iii) the watermark embedding location is selected after θ° rotation of the original image; (iv) the watermark bit is selected based on max-heap tree and min-heap tree property; (v) the watermark detection procedure is blind; and (vi) it achieves a good trade-off among security, imperceptibility, and robustness. Simulation results illustrate that the proposed method is highly robust and secured against different attacks. Moreover, the proposed method provides better results in terms of robustness and imperceptibility compared with the recent methods [17,24]. The PSNR, structural similarity index (SSIM), and normalized correlation (NC) of the proposed method vary within 51.02–53.39 dB, 0.9969–0.9988, and 0.9567–0.9986, respectively, in contrast to the recent methods whose PSNR, SSIM, and NC range 38.5471–41.5391 dB, 0.9804–0.9975, and 0.6482–0.9998, respectively.

Furthermore, in terms of security, the proposed scrambling method shows better performance than some well known scrambling methods.

The remainder of this paper is organized as follows. Section 2 briefly describes the background information including DLCT and CT. Section 3 introduces the proposed watermarking method consisting of four-connected t-o'clock scrambling technique, watermark embedding, and extraction processes. Section 4 provides the experimental results and compares the performance of the proposed method with recent methods in terms of imperceptibility, robustness, and security. Finally, in Section 5, the conclusion of this paper is presented.

2. Background Information

2.1. The Discrete Linear Canonical Transform (DLCT)

In DLCT, let $\delta_a = \delta_b = (N'/|\beta|)^{-1/2}$, $a = n'\delta_a$, $b = m'\delta_b$, and $m', n' = 0, 1, \ldots, N'$. Then, the N' point discrete LCT (DLCT) of *f(n')* can be defined as [28]

$$f(m') = \sum_{n'=0}^{N'-1} f(n')C'(m', n') \tag{1}$$

where

$$C'(m', n') = \frac{\sqrt{\beta}e^{-j\pi/4}}{\sqrt{N'|\beta|}} exp\left[j\pi \frac{1}{N'|\beta|}\left(\alpha m'^2 - 2\beta m'n' + \gamma n'^2\right)\right].$$

Because it is interval-independent and unitary, the DLCT method is available for image processing. Moreover, it also has the property of index additivity. The two-dimensional DLCT of an image *H(x, y)* can be rewritten as

$$H(x, y) = \sum_{y=0}^{Y-1} C'(y, n') \sum_{x=0}^{X-1} H(y, n')C'(x, m') \tag{2}$$

with $x \in \{0, 1, \ldots, X-1\}$, $y \in \{0, 1, \ldots, Y-1\}$. $C'(x, m')$ and $C'(y, n')$ are the parameters used in Equation (1).

2.2. The Cepstrum Transform (CT)

Cepstrum analysis, which is mainly used in speech analysis and recognition, is a non-linear signal processing technique. Cepstrul analysis consists of three steps: (1) apply Fourier transform (*FFT*) to a given signal; (2) take the logarithm (*ln*) of the *FFT* coefficients; and (3) apply inverse Fourier transform (*IFFT*) to this signal [29]. This procedure is described in the following

$$C(n) = IFFT\left(ln\left(FFT\left(x\left(n\right)\right)\right)\right) \tag{3}$$

where *x(n)* is an original signal. The *FFT* is applied to *x(n)* to obtain transform coefficients and then *ln* operation is applied to *FFT* coefficients. Finally, cepstrum sequence *c(n)* is obtained by applying the *IFFT* on *ln* coefficients.

3. Proposed Watermarking Framework

Let $H = \{h(x, y) \mid 1 \le x \le X, 1 \le y \le Y\}$ be a host image and $W = \{w(i, j) \mid 1 \le i \le I, 1 \le j \le J\}$ be a binary watermark image. Here, $w(i, j) \in (0, 1)$ is the pixel value at the point (i, j) that is embedded into the host image.

3.1. Watermark Preprocessing

To strength the security and to get an effective result in robustness, the watermark image should be preprocessed. In this paper, the four-connected t-o'clock scrambling method, as described below, is

used to encode the watermark image for improving the security of the proposed scheme. The binary watermark image is scrambled using Equation (4) .

Suppose an image W with the size $I \times J$ is scrambled using a four-connected t-o'clock method to form a new scrambled image W'. The randomly selected starting point S and the value t are considered as secret key K. This process is described below.

Let, S be a randomly selected pixel in W with the pixel position of (i, j). Now, we have to select for other neighbor pixels with the basis of t, where t=12, 3, 6, or 9. Therefore, the four neighbor pixels can be selected as

$$w'(i,j) = \begin{Bmatrix} w\left[(i-1,j),(i,j+1),(i+1,j),(i,j-1)\right] & if & t=12 \\ w\left[(i,j+1),(i+1,j),(i,j-1),(i-1,j)\right] & if & t=3 \\ w\left[(i+1,j),(i,j-1),(i-1,j),(i,j+1)\right] & if & t=6 \\ w\left[(i,j-1),(i-1,j),(i,j+1),(i+1,j)\right] & if & t=9 \end{Bmatrix} \quad (4)$$

where W' is the new scrambled image. Furthermore, $w'(1,1) = w(i,j)$, where *(i, j)* is the coefficient of randomly selected pixel S. The procedure is shown in Figure 1a with the different values of t and Figure 1b illustrates the detail procedure when the value of t is 12. The steps are described in the following.

Step 1. Initially, a pixel and the value of the t is chosen. Here, the selected pixel is $S(i,j) = S(2,2) = 6$ and the value of *t=12*, which is shown in Block 1 of Figure 1b.

Step 2. In this step, four neighbor pixels of $S(2,2) = 6$ are chosen using Equation (1). The four neighbor pixels are $w(i-1,j) = w(1,2) = 2$, $w(i,j+1) = w(2,3) = 7$, $w(i+1,j) = w(3,2) = 10$, and $w(i,j-1) = w(2,1) = 5$. Pictorial representation of this step is shown in Block 2 of Figure 1b.

Step 3. In this step, the four neighbor pixels of pixel 2 are selected. However, pixel 2 is a boundary value pixel and the pixels that exceed boundary value are not selected, where the boundary value $(i,j) \in (1,4)$. Therefore, in this step, only two neighbor pixels are chosen, which is shown in Block 3 of Figure 1b.

Step 4 In this stage, four neighbor pixels of pixel 7 is selected. In Block 4 of Figure 1b, we observe that only two neighbor pixels, namely 8 and 11 at position $(2,4)$ and $(3,3)$, respectively, are selected. Only these two pixels are chosen to avoid the redundancy.

Step 5. In this step, the neighbor pixels of pixel 10 is selected, which is a boundary value pixel; therefore, the pixels that exceed boundary value are excluded. In addition, pixels have already been chosen are excluded. Therefore, the only chosen neighbor pixel of 10 is 9. This step is illustrated in Block 5 of Figure 1b.

Step 6. In this step, only neighbor pixel 4 of pixel 3 is selected, which is shown in Block 6 of Figure 1b.

Step 7. The neighbor pixels of 8 are chosen in this step. Pixel 8 is a boundary value pixel and most of the neighbor pixels are already selected. Therefore only pixel 12 is chosen in this step, which is shown in Block 7 of Figure 1b.

Step 8. In Block 8, we find that all the pixels and their neighbor pixels are selected.

Step 9. In the final step, all pixels are shuffled according to the selection. Then, a scrambled image is generated.

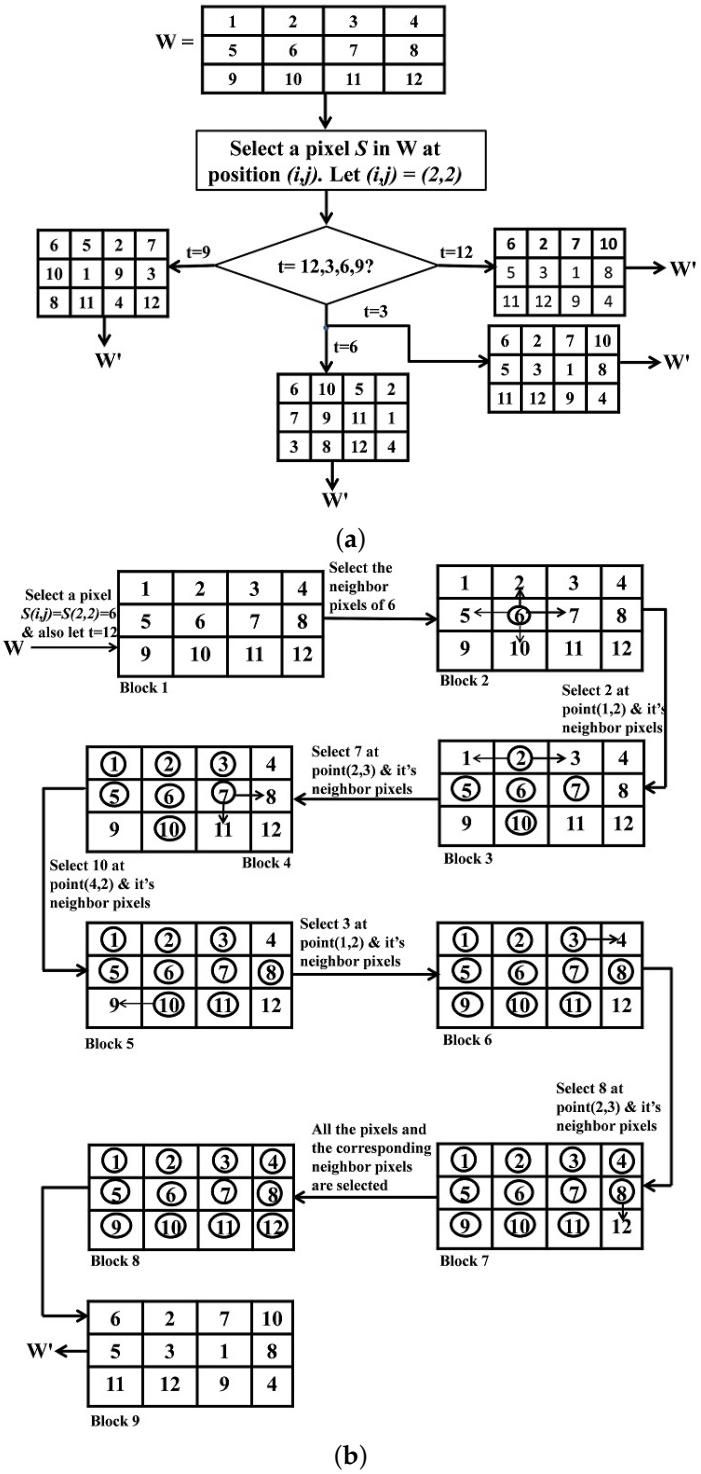


Figure 1. Example of four-connected t-o'clock scrambling method: (**a**) block diagram of the scrambled block; and (**b**) detailed steps when $t = 12$.

3.2. Embedding Location

Let (x, y) be the co-ordinate of the pixel and, after θ° rotation, it is (x', y'). This can be found according to the following equations:

$$x' = x\cos\theta - y\sin\theta \quad \text{and} \tag{5}$$

$$y' = x \sin\theta + y \cos\theta. \tag{6}$$

From the analysis, we found that up to 60° we lost the pixel value of an image. This loss continues for five consecutive columns and rows in the upper left-right and lower left-right corners of a 16×16 given image. For this reason, the embedding region $R = \{r(p,q) \mid i' \leq p \leq (X - i'), j' \leq q \leq (Y - j')\}$ is selected in such a way that the pixels are excluded in all directions, where $(i', j') \in \{5 \leq i' \leq (X - i'), 5 \leq j' \leq (Y - j')\}$.

3.3. Watermark Embedding Framework

The watermark region is selected from the host image according to the process described above. After that, DLCT is applied to that selected region. Here, DLCT parameters are $\alpha = 0.2$, $\beta = 0.6$, and $\gamma = 0.1$. Then, CT is applied to this region. The CT region is divided into $m \times m$ non-overlapping blocks. We select the pixels for embedding watermark based on max-heap tree and min-heap tree property [30]. The watermark embedding process is depicted in Figure 2. The steps of watermark embedding can be summarized as follows

1. The original host image H is separated into three channels R, G, and B.
2. Select the channel G for embedding watermark information.
3. A region $R = \{r(p,q) \mid i' \leq p \leq P, j' \leq q \leq Q\}$ using Equations (5) and (6) is extracted from channel G, where the watermark bit is embedded, with $P = X - i'$, $Q = Y - j'$, $i' \in \{5, 6, \ldots, X - i'\}$, and $j' \in \{5, 6, \ldots, Y - j'\}$.
4. Apply DLCT to R to obtain $T = \{t(a', b') \mid 1 \leq a' \leq P, 1 \leq b' \leq Q\}$ using Equation (2).
5. Then, apply CT to T with Equation (3) to obtain C denoted as $C = \{c(u,v) \mid 1 \leq u \leq P, 1 \leq v \leq Q\}$.
6. The cepstrum region C is segmented into L non-overlapping blocks B with size $m \times m$, where $B = \{B_1, B_2, B_3, \ldots, B_L\}$.
7. Watermark image W is encrypted using four-connected t-o'clock scrambling to get W' image.
8. Watermark bit is embedded into each blocks B using max-heap tree and min-heap tree property to obtain B', where $B' = \{B'_1, B'_2, B'_3, \ldots, B'_L\}$. A max-heap is a complex binary tree in which the value of each internal node is greater than or equal to the value of the children of that node. For the min-heap tree, the value of each internal node is less than its child node. If $w(i,j) = 1$, then apply max heap tree procedure to the block. If $w(i,j) = 0$, then apply min-heap tree property. This process is described in Figure 3.
9. After embedding all watermark bits, concatenate all sub-blocks B' to obtain watermarked CT region C'.
10. Then, apply inverse CT to C' to obtain watermarked region T'.
11. Apply inverse DLCT to T' to obtain watermarked region R'.
12. Reinsert the watermarked region R' to obtain G' channel and finally concatenate all three channels to get the watermarked image H'.

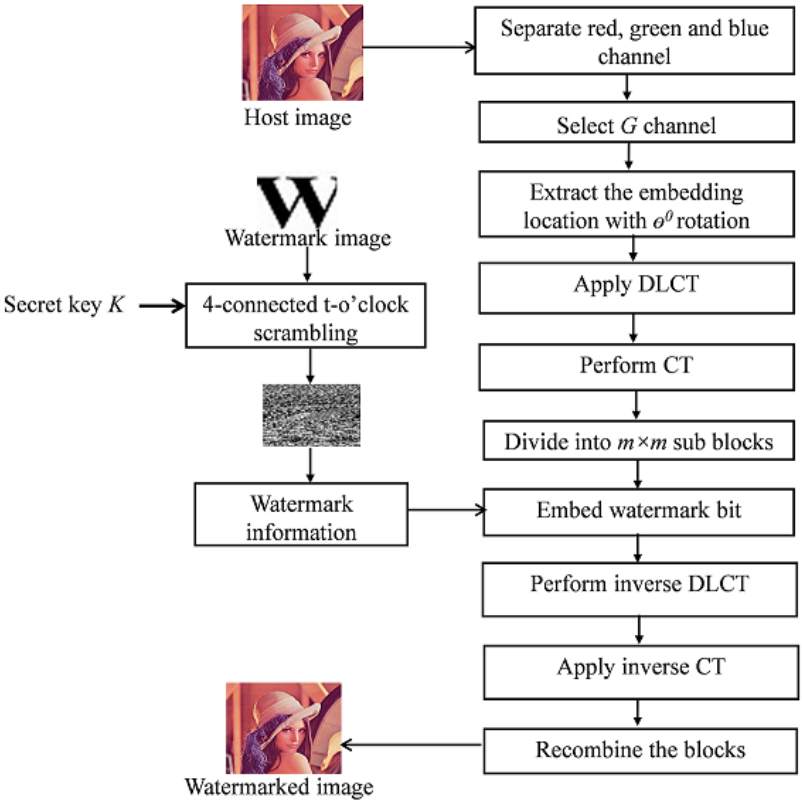


Figure 2. Watermark embedding framework.

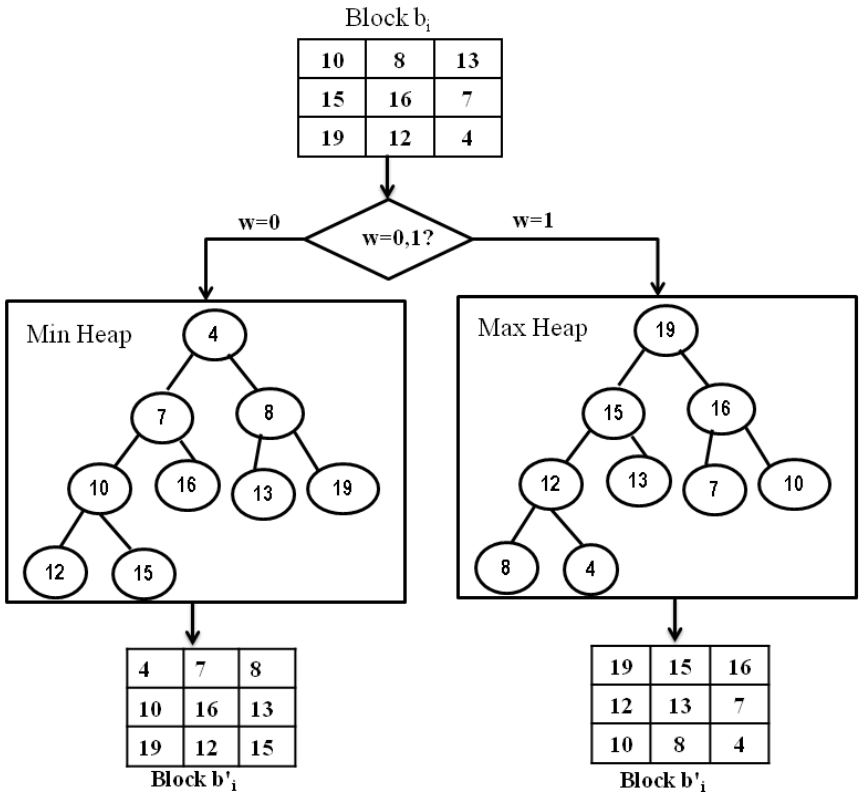


Figure 3. Block generation based on max-heap tree and min-heap tree property.

3.4. Watermark Extraction Framework

The proposed extraction procedure is shown in Figure 4. The watermark image is extracted without using the host image. The steps of the watermark extraction process are described as follows.

1. Apply DLCT to the extracted region R^* to get T^* and then apply CT to that region T^* to obtain region C^*.
2. Divide the C^* region into non-overlapping block B^* with size $m \times m$.
3. Extract the watermark bit from each block. If a selected block satisfies the max-heap tree property, then the watermark bit will be 1. If the selected block satisfies the min-heap tree property, then the watermark bit will be 0.
4. Finally, the inverse four-connected t-o'clock method is applied to reconstruct each component of watermark image W^*.

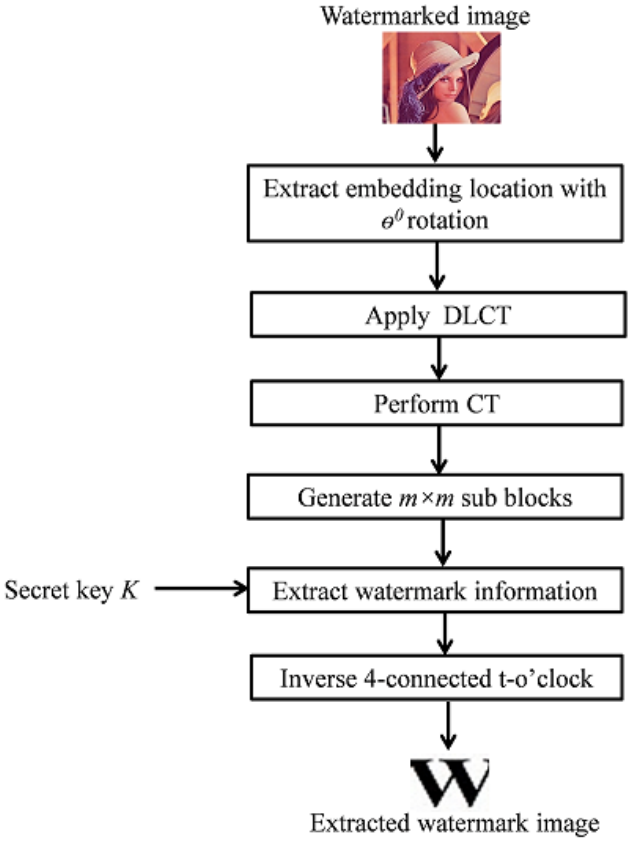


Figure 4. Watermark extraction framework.

4. Experimental Results

We evaluated the performance of the proposed algorithm and also compared it with some recent methods. The proposed algorithm was implemented in MATLAB R2016b environment with core i5 processor and 8 GB RAM. The tested images were collected from the USC-SIPI image dataset [31]. In this study, Mandrill, Lena, Fruits, and Peppers were used as the original host images (Figure 5). In the simulation, the original RGB host image with a size of 512×512 and a binary watermark image with a size of 32×32 was used. The selected region of the host image for embedding watermark was 192×192. The block size of the extracted host image used for embedding watermark was 3×3. In this study, the selected value for different parameters were $\alpha = 0.2$, $\beta = 0.6$, and $\gamma = 0.1$. In addition, the randomly chosen coordinate for scrambling process $S(i, j)$ was $S(2, 2)$ and the

value of t was 12. These parameters were chosen for achieving a good trade-off among robustness, imperceptibility, and security.

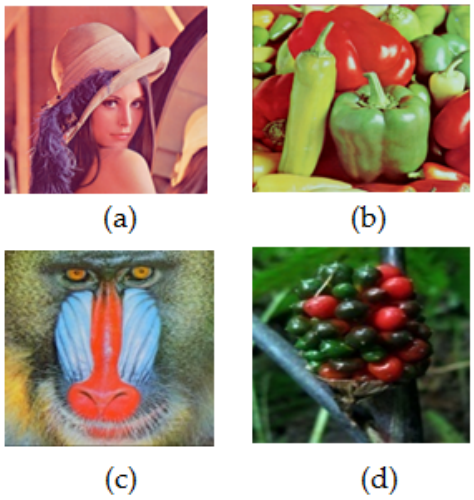


Figure 5. The host images: (**a**) Lena; (**b**) Pepper; (**c**) Mandrill; and (**d**) Fruits.

4.1. Imperceptibility Test

Imperceptibility means that the perceptual quality of the original image should remain the same as the watermarked image in the presence of watermark. The perceptual quality of the watermarked image can be calculated based on the *PSNR* and *SSIM* [16,21] given in Equations (8) and (9), respectively

$$MSE = \frac{1}{XY}\sum_{x=1}^{X-1}\sum_{y=1}^{Y-1}\left[h\left(x,y\right) - w'\left(x,y\right)\right]^2 \tag{7}$$

$$PSNR = 10\log_{10}\frac{255^2}{MSE} \tag{8}$$

$$SSIM(H,H') = \frac{\left(2\mu_H\mu_{H'}\right)\left(2\sigma_{HH'} + c_2\right)}{\left(\mu_H^2 + \mu_{H'}^2 + c_1\right)\left(\sigma_H^2 + \sigma_{H'}^2 + c_2\right)} \tag{9}$$

where MSE is the mean square error of the watermarked image, $h(x,y)$ represents the pixel values of the original image, $h'(x,y)$ represents the pixel values of the watermarked image, H is the original image, H' is the watermarked image, μ_H is the local means of H, $\mu_{H'}$ is the local means of H', and σ_H^2 and $\sigma_{H'}^2$ are the variances of H and H', respectively.

Table 1 demonstrates the PSNR and SSIM of the watermarked image using the proposed method. Generally, if the PSNR value is larger than 35 dB [11], then the watermark information is invisible to human eyes. After embedding watermark information to the host images, the PSNR of the four watermarked images range from 50.01 dB to 53.30 dB, which satisfies the standard PSNR requirements. Along with PSNR, the perceptual capability was also evaluated using SSIM. SSIM closes to 1 indicates the high perceptual quality of watermarked images. In Table 1, it is observed that the SSIM of the proposed method ranges from 0.9969 to 0.9985.

Table 1. PSNR and SSIM of various watermarked images.

Images	PSNR	SSIM
Lena	53.04	0.9980
Pepper	52.71	0.9985
Mandrill	51.02	0.9969
Fruits	53.39	0.9988

Further, Table 2 illustrates a comparison between the proposed and recent methods [17,24] in terms of PSNR and SSIM. The PSNR and SSIM of the proposed method vary from 51.02 dB to 53.39 dB and 0.9969 to 0.9988, respectively, in contrast to the recent method whose PSNR and SSIM vary from 38.5471 dB to 41.5391 dB and 0.9804 to 0.9975, respectively.

Table 2. Comparative analysis between the proposed and recent method in terms of PSNR and SSIM.

Watermarking Methods	Cover Image	PSNR	SSIM
[17]	Lena	41.5391	0.9975
	Mandrill	40.8315	0.9918
	Pepper	39.8431	0.9821
	Fruits	41.4162	0.9972
[24]	Lena	38.5471	0.9804
	Mandrill	41.2176	0.9870
	Pepper	41.3236	0.9908
	Fruits	40.59041	0.9911
Proposed Method	Lena	53.04	0.9980
	Mandrill	51.02	0.9969
	Pepper	52.71	0.9985
	Fruits	53.39	0.9988

4.2. Robustness Test

Robustness indicates that the watermark should not be removed by an unauthorized person. The normalized correlation (NC) was used to measure the similarity between the original watermark image and the extracted watermark image. The *NC* is calculated using Equation (10)

$$NC = \frac{\sum_{i=1}^{I} \sum_{j=1}^{J} w\left(i,j\right), w'\left(i,j\right)}{\sum_{i=1}^{I} \sum_{j=1}^{J} w\left(i,j\right)^2} \tag{10}$$

where *w(i,j)* and *w'(i,j)* are the original watermark and extracted watermark images, respectively. Different types of noise attacks were applied on watermarked Lena, pepper, Mandrill, and Fruits images. The robustness of the proposed watermarking scheme was evaluated by conducting various attacks on the watermarked image such as JPEG compression, cropping attack, filtering, noise attacks, rotational attacks, etc. which are given as follows:

1. JPEG compression: JPEG compression is a standard lossy compression technique in which an image is compressed to reduce its memory space and bandwidth requirements for transmission over the Internet. In our simulation, JPEG compression with $QF = 90$ was applied to the watermarked images.
2. Cropping: The watermarked images were cropped 50% from the top.
3. Rotation attack: The watermarked images were rotated by 3° and the rotated images were re-rotated in a counter-clockwise for extraction of watermark images.
4. Gaussian noise: Gaussian noise with variance 0.1 was applied to the watermarked images.
5. Speckle noise: Speckle noise with variance 0.01 was applied to the watermarked images.
6. Salt and pepper noise: Salt and pepper noise with variance 0.01 was applied to the watermarked images.
7. Poison noise: Poison noise was applied to the watermarked images.
8. Contrast adjustment: Contrast adjustment with minimum 0.2 and maximum 0.6 was applied to the watermarked images.
9. Sharpening: Sharpening with tolerance 0.1 was applied to the watermarked images.
10. Median filtering: 3×3 median filter was applied to the watermarked images.
11. Wiener filtering: 3×3 wiener filter was applied to the watermarked images.

Figures 6–8 show the NC of four images after applying various attacks. In Figure 6, JPEG compression attack, cropping, and rotation attack are analyzed. In this figure, we observe that the NC of the proposed method varies 0.8496–0.9986 for the JPEG compression attack, 0.9512–0.9719 for cropping attack, and 0.8765–0.9760 for rotation attack, respectively.

In Figure 7, the NC varies 0.9351–1.0 for Gaussian noise attack, 0.9068– 0.9915 for speckle noise attack, 0.9931–0.9956 for salt and pepper noise attack, and 0.9950–0.9992 for poison noise attack.

Finally, in Figure 8, the NC ranges 0.7644–0.9743 for contrast adjustment attack , 0.8594–0.9567 for sharpening attack, 0.9459–0.9902 for median filtering attack, and 0.7085–0.9934 for wiener filtering attack. From these results, we can say that the proposed method has high robustness against various attacks.

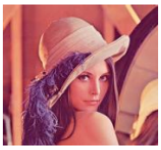
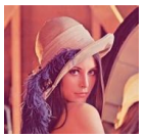
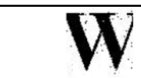
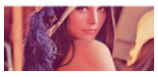
Attack type		Lena	Pepper	Mandrill	Fruits
No Attack	Watermarked Image				
	PSNR	53.04	52.71	51.02	53.39
	Extracted Watermark				
	NC	1.0	1.0	1.0	1.0
Compression (QF: 90%)	Watermarked Image				
	Extracted Watermark				
	NC	0.9986	0.9246	0.8571	0.8496
Cropped (50%)	Watermarked Image				
	Extracted Watermark				
	NC	0.9719	0.9719	0.9512	0.9710
Rotation (3^0)	Watermarked Image				
	Extracted Watermark				
	NCC	0.9760	0.8765	0.8920	0.9497

Figure 6. Analysis of the proposed method under no attack, compression (quality factor: 90%), cropping (50%), and rotation 3°.

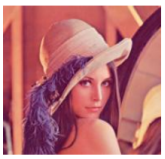
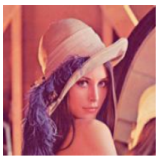
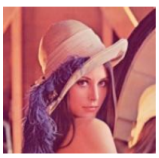
Attack type		Lena	Pepper	Mandrill	Fruits
Salt & Pepper noise(0.01)	Watermarked Image				
	Extracted Watermark				
	NC	0.9945	0.9931	0.9956	0.9944
Gaussian Noise (0.1)	Watermarked Image				
	Extracted Watermark				
	NC	0.9925	0.9823	1.0	0.9351
Speckle Noise (0.01)	Watermarked Image				
	Extracted Watermark				
	NC	0.9915	0.9292	0.9068	0.9349
Poison noise	Watermarked Image				
	Extracted Watermark				
	NC	0.9950	0.9963	0.9992	0.9956

Figure 7. Analysis of the proposed method under salt and pepper noise (0.01), Gaussian noise (0.01), speckle noise (0.01), and poison noise attack.

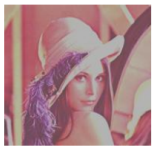
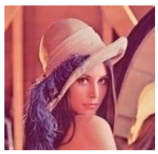
Attack type		Lena	Pepper	Mandrill	Fruits
Contrast adjustment	Watermarked Image				
	Extracted Watermark				
	NC	0.9743	0.7644	0.9014	0.8237
Sharpening (tol=0.1)	Watermarked Image				
	Extracted Watermark				
	NC	0.9567	0.9335	0.9241	0.8594
Weiner filtering	Watermarked Image				
	Extracted Watermark				
	NC	0.9934	0.7085	0.7384	0.7471
Median filtering	Watermarked Image				
	Extracted Watermark				
	NC	0.9902	0.9541	0.9896	0.9459

Figure 8. Analysis of the proposed method under Contrast adjustment, Sharpening (0.1), median filtering, and wiener filtering.

Table 3 shows a comparison between the proposed and recent methods [17,24] against various attacks. In this table, we observe that the NC of the proposed method against various attacks varies from 0.9567 to 0.9986, in contrast to recent methods whose NC vary from 0.6482 to 0.9998. In other words, the proposed method outperforms recent methods in terms of robustness.

Table 3. Comparison of the proposed algorithm with recent methods in terms of robustness.

Attack Type	[17]	[24]	Proposed Method
Gaussian(0.1)	0.9625	0.8823	0.9925
Speckle noise (0.01)	0.9663	0.9647	0.9915
Cropping (50%)	0.6482	0.8619	0.9719
Sharpening ($tol = 0.1$)	0.9935	0.9882	0.9567
Rotation (3°)	0.9361	0.9225	0.9760
Wiener filtering	0.9578	0.9765	0.9934
Salt and pepper noise (0.01)	0.9478	0.9733	0.9945
Median filtering	0.9419	0.8997	0.9902
JPEG Compression	0.9998	0.9791	0.9986

4.3. The Computational Time Comparison Analysis

Computation time, also known as running time or execution time, is the length of time required to perform a computational process. Table 4 shows a comparison between the proposed and recent methods [17,24] in terms of computational time. Our proposed method has shorter execution time than the method in [24] and higher computational time than the method in [17]. The main reason is that our method is performed on the frequency domain, which consists of complex computation, whereas the method in [17] performed in spatial domain, which only includes simple mathematical calculation.

Table 4. Comparison of the proposed algorithm with recent methods in terms of computational time.

Method	Embedding Time	Extraction Time	Total Time
[17]	0.274117	0.238315	0.512432
[24]	0.810820	0.269506	1.080326
Proposed Method	0.5016456	0.2394905	0.7411361

4.4. Security Analysis of Proposed Scrambling Method

The security of the proposed method was analyzed and compared with some recent methods. Figure 9a–d shows the original image and its scrambled image using the proposed four-connected t-o'clock scrambling method and the methods presented in [17,24]. To enhance the security, the randomly selected starting point S and the value t were considered as secret key K. In addition, the proposed method was compared with recent methods using some performance matrices such as correlation coefficient (CC), information entropy (IE), relative entropy (RE), number of pixel change rate (NPCR), and universal average change intensity (UACI) to evaluate the security.

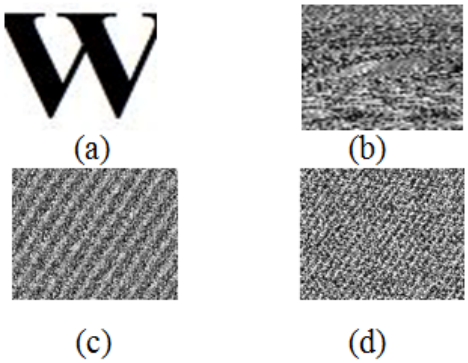


Figure 9. (**a**) Original image; (**b**) the image scrambled using proposed method; (**c**) the image scrambled using Schur decomposition method [17]; and (**d**) LU the image scrambled using decomposition method [24].

4.4.1. Correlation Coefficient (CC)

The correlation coefficient (CC) is one of the important metrics to evaluate the security of scrambling method. It defines the degree of resemblance between two variables [32]. The image scrambling method is considered efficient if it can distort the data of the host image completely with the lowest correlation value. CC between a host image and the encrypted image is equal to 1 and -1 for the inverse image. On the other hand, it is almost zero for a highly uncorrelated image [32]. *CC* between a host image and scrambled image can be calculated as

$$CC = \frac{cov\,(i,j)}{\sqrt{var\,(w) \times var\,(w')}},$$

$$var\,(w) = \frac{1}{J}\sum_{i=1}^{J}\left(w_i - E\,(w_i)\right)^2,$$

$$cov\,(w,w') = \frac{1}{J}\sum_{i=1}^{J}\left(w_i - E\,(w)\right)\left(w'_i - E\,(w')\right), \tag{11}$$

where w is the original image, w' is the scrambled image, $cov(w,w')$ is the co-variance between w and w', $var(w)$ is the variance at pixel value of w, $var(w')$ is the variance at pixel value of w', $E(w)$ is the mean of w_i, and $E(w')$ is the mean of w'_i. Table 5 shows a comparison among proposed method and two conventional methods [17,24]. In this table, it is shown that the proposed method effectively scrambled the original image compared with the other two methods.

Table 5. Comparison of the proposed algorithm with recent methods in terms of CC.

CC	[17]	[24]	Proposed Method
Horizontal	0.0074	0.0082	0.00040
Vertical	0.0065	0.0070	0.0024
Diagonal	0.0098	0.0058	0.0039

4.4.2. Information Entropy (IE)

Information entropy (IE) defines the quantities of uncertainties in an image. To improve the security of an image, the entropy of the scrambled image must be greater than the entropy of the host image [33]. The entropy of a source can be defined as

$$E(W) = -\sum_{i=1}^{J-1} w(W_i)\log_2\frac{1}{w(W_i)} \tag{12}$$

where W is the source image, $w(W_i)$ is the probability of W_i, and J is the number of bits to indicate W_i. Table 6 shows a comparison between the proposed method and two conventional methods [17,24] in terms of IE. Regardless of the host image, a greater entropy value indicates higher security. The IE of watermarked images using the proposed scheme are much higher than those of the other methods. Thus, we can say that the proposed t-o'clock scrambling method provides more security than the other methods.

Table 6. Comparison of the proposed algorithm with recent methods in terms of IE.

Scrambling Methods	Watermarked Image		
	Red	Green	Blue
[17]	2.2902	2.3444	2.4673
[24]	2.6372	2.4108	2.1091
Proposed Method	3.3553	3.4476	3.3274

4.4.3. Relative Entropy (RE)

Relative Entropy (RE) determines the diversity of two probability distributions over histograms of the two digital images. The relative entropy of two probability distributions is defined as the sum of all possible situations, which is represented by

$$R(I) = \sum_{i=1}^{I} w\left(I_i\right) \log_2 \frac{w\left(I_i\right)}{w'\left(I_i\right)} \tag{13}$$

where $w(I_i)$ and $w'(I_i)$ are the two probability distributions. If the probability distributions $w(I_i)$ and $w'(I_i)$ are identical to each other, then the relative entropy reaches zero. A comparative analysis between the proposed and several conventional methods [17,24] is illustrated in Table 7. In this table, we observe that the proposed method provides higher RE compared with several methods. This indicates that the proposed scrambling method is more secure than other methods.

Table 7. Comparison of the proposed algorithm with recent methods in terms of RE.

Scrambling Methods	Watermarked Image		
	Red	Green	Blue
[17]	0.0792	0.2091	0.2211
[24]	0.0122	0.3141	0.2135
Proposed method	0.0658	0.6850	0.5104

4.4.4. Differential Analysis

The security of a scrambling method can also be measured using the number of pixel change rate (*NPCR*) and the universal average change intensity (*UACI*) tests [32]. NPCR means the number of pixels that change in the encrypted image when one pixel change occurs in the original image. Suppose $w(i,j)$ and $w'(i,j)$ are the pixel values of original and scrambled images denoted by w and w' with ith row and jth column, respectively. This metric can be represented as

$$NPCR = \frac{\sum_{i,j} D\left(i,j\right)}{I \times J} \times 100\% \tag{14}$$

where

$$D(i,j) = \begin{Bmatrix} 0 & if w(i,j) = w'(i,j) \\ 1 & if w(i,j) \neq w'(i,j) \end{Bmatrix}. \tag{15}$$

A high value in NPCR means that the change in pixel value has a high impact on the appearance of an image. The theoretical critical value for this test is above 90% [32].

UACI measures the average intensity differences between the original image and scrambled image. It is defined as follows:

$$UACI = \frac{1}{I \times J} \sum_{i,j} \frac{w(i,j) - w'(i,j)}{255} \times 100\%. \tag{16}$$

The theoretical critical value for this test is above 33% [32] The NPCR of the proposed method is 34.6550%. Table 8 shows a comparison between the proposed method and two conventional methods [17,24] in terms of NPCR and UACI. The NPCR and UACI of the proposed four-connected t-o'clock scrambling method are 99.54% and 34.45, respectively. On the other hand, the NPCR and UACI of the methods in [17,24] are 98.48% and 98.76% and 25.40 and 24.38, respectively. From this comparison, we can say that the four-connected t-o'clock scrambling method is highly susceptible to small modification and is more secure against various attacks for privacy protection.

Table 8. Comparison of the proposed algorithm with recent methods in terms of NPCR and UACI.

Scrambling Methods	NPCR (%)	UACI(%)
[17]	98.48	25.40
[24]	98.76	24.38
Proposed Method	99.54	34.45

5. Conclusions

In this paper, a blind image watermarking method using DLCT, CT, and four-connected t-o'clock scrambling is proposed. DLCT is chosen to ensure the security, because it is not possible to extract the watermark image without knowing the DLCT parameters. CT is used to ensure low computational cost as well as high imperceptibility and robustness. On the other hand, the four-connected t-o'clock scrambling method is used to scramble the watermark, which secures the method against unauthorized detection. Experimental results indicate that the proposed method is highly robust against numerous attacks because the region for embedding watermark is selected in such a way that the proposed method can resist unauthorized attacks. Besides, it produces high quality watermarked images and provides high security. Furthermore, the proposed method shows superior performance to the recent state-of-the-art methods in respect of imperceptibility, robustness, and security. These results verify that the presented method can be used efficiently for copyright and ownership protection.

Author Contributions: All authors contributed equally to the conception of the idea, the design of experiments, the analysis and interpretation of results, and the writing and improvement of the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cox, I.J.; Miller, M.L. The first 50 years of electronic watermarking. *EURASIP J. Adv. Signal Process.* **2002**, *2002*, 820936. [CrossRef]
2. Naderahmadian, Y.; Hosseini-Khayat, S. Fast and robust watermarking in still images based on QR decomposition. *Multimed. Tools Appl.* **2014**, *72*, 2597–2618. [CrossRef]
3. Pizzolante, R.; Castiglione, A.; Carpentieri, B.; De Santis, A.; Castiglione, A. Protection of microscopy images through digital watermarking techniques. In Proceedings of the 2014 International Conference on Intelligent Networking and Collaborative Systems, Salerno, Italy, 10–12 September 2014; pp. 65–72.
4. Liu, S.; Pan, Z.; Song, H. Digital image watermarking method based on DCT and fractal encoding. *IET Image Process.* **2017**, *11*, 815–821. [CrossRef]
5. Li, J.; Zhang, C. Blind and robust watermarking scheme combining bimodal distribution structure with iterative selection method. *Multimed. Tools Appl.* **2019**, 79, 1–35. [CrossRef]
6. Parah, S.A.; Sheikh, J.A.; Loan, N.A.; Bhat, G.M. Robust and blind watermarking technique in DCT domain using inter-block coefficient differencing. *Digit. Signal Process.* **2016**, *53*, 11–24. [CrossRef]
7. Roy, S.; Pal, A.K. A robust blind hybrid image watermarking scheme in RDWT-DCT domain using Arnold scrambling. *Multimed. Tools Appl.* **2017**, *76*, 3577–3616. [CrossRef]
8. Roy, S.; Pal, A.K. A blind DCT based color watermarking algorithm for embedding multiple watermarks. *AEU Int. J. Electron. Commun.* **2017**, *72*, 149–161. [CrossRef]
9. Abdulrahman, A.K.; Ozturk, S. A novel hybrid DCT and DWT based robust watermarking algorithm for color images. *Multimed. Tools Appl.* **2019**, *78*, 17027–17049. [CrossRef]
10. Hamidi, M.; El Haziti, M.; Cherifi, H.; El Hassouni, M. Hybrid blind robust image watermarking technique based on DFT-DCT and Arnold transform. *Multimed. Tools Appl.* **2018**, *77*, 27181–27214. [CrossRef]
11. Cedillo-Hernández, M.; García-Ugalde, F.; Nakano-Miyatake, M.; Pérez-Meana, H.M. Robust hybrid color image watermarking method based on DFT domain and 2D histogram modification. *Signal Image Video Process.* **2014**, *8*, 49–63. [CrossRef]

12. Preda, R.O. Self-recovery of unauthentic images using a new digital watermarking approach in the wavelet domain. In Proceedings of the 2014 10th International Conference on Communications (COMM), Bucharest, Romania, 29–31 May 2014; pp. 1–4.
13. Hu, J.; Shao, Y.; Ma, W.; Zhang, T. A robust watermarking scheme based on the human visual system in the wavelet domain. In Proceedings of the 2015 8th International Congress on Image and Signal Processing (CISP), Shenyang, China, 14–16 October 2015; pp. 799–803.
14. Keshavarzian, R.; Aghagolzadeh, A. ROI based robust and secure image watermarking using DWT and Arnold map. *AEU Int. J. Electron. Commun.* **2016**, *70*, 278–288. [CrossRef]
15. Ahmad, A.; Sinha, G.; Kashyap, N. 3-level DWT Image watermarking against frequency and geometrical attacks. *Int. J. Comput. Netw. Inf. Secur.* **2014**, *6*, 58. [CrossRef]
16. Sheth, R.K.; Nath, V. Secured digital image watermarking with discrete cosine transform and discrete wavelet transform method. In Proceedings of the 2016 International Conference on Advances in Computing, Communication, & Automation (ICACCA) (Spring), Dehradun, India, 8–9 April 2016; pp. 1–5.
17. Su, Q.; Yuan, Z.; Liu, D. An Approximate Schur Decomposition-Based Spatial Domain Color Image Watermarking Method. *IEEE Access* **2018**, *7*, 4358–4370. [CrossRef]
18. Rasti, P.; Anbarjafari, G.; Demirel, H. Colour image watermarking based on wavelet and QR decomposition. In Proceedings of the 2017 25th Signal Processing and Communications Applications Conference (SIU), Antalya, Turkey, 15–18 May 2017; pp. 1–4.
19. Guo, J.M.; Prasetyo, H. False-positive-free SVD-based image watermarking. *J. Vis. Commun. Image Represent.* **2014**, *25*, 1149–1163. [CrossRef]
20. Çerkezi, L.; Çetinel, G. RDWT and SVD based secure digital image watermarking using ACM. In Proceedings of the 2016 24th Signal Processing and Communication Application Conference (SIU), Zonguldak, Turkey, 16–19 May 2016; pp. 149–152.
21. He, Y.; Hu, Y. A proposed digital image watermarking based on DWT-DCT-SVD. In Proceedings of the 2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Xi'an, China, 23–28 March 2018; pp. 1214–1218.
22. Fazli, S.; Moeini, M. A robust image watermarking method based on DWT, DCT, and SVD using a new technique for correction of main geometric attacks. *Optik Int. J. Light Electron Opt.* **2016**, *127*, 964–972. [CrossRef]
23. Singh, C.; Ranade, S.K. Geometrically invariant and high capacity image watermarking scheme using accurate radial transform. *Opt. Laser Technol.* **2013**, *54*, 176–184. [CrossRef]
24. Su, Q.; Wang, G.; Zhang, X.; Lv, G.; Chen, B. A new algorithm of blind color image watermarking based on LU decomposition. *Multidimens. Syst. Signal Process.* **2018**, *29*, 1055–1074. [CrossRef]
25. Pu, C. Image scrambling algorithm based on image block and zigzag transformation. *Comput. Model. New Technol.* **2015**, *18*, 489–493.
26. Ding, M.; Jing, F. Digital image encryption algorithm based on improved Arnold transform. In Proceedings of the 2010 International Forum on Information Technology and Applications, Kunming, China, 16–18 July 2010; Volume 1, pp. 174–176.
27. Tang, Z.; Zhang, X. Secure image encryption without size limitation using Arnold transform and random strategies. *J. Multimed.* **2011**, *6*, 202. [CrossRef]
28. Li, B.Z.; Shi, Y.P. Image watermarking in the linear canonical transform domain. *Math. Probl. Eng.* **2014**, *2014*, 1–9. . [CrossRef]
29. Zhang, M.-R.; Lu, C.-H.; Yi, K. Cepstrum digital image watermarking algorithm. In Proceedings of the 2004 IEEE International Symposium on Industrial Electronics, Ajaccio, France, 4–7 May 2004; Volume 1, pp. 283–286.
30. Lipschutz, S. *Data Structures with C (Schaum's Outline Series)*; Tata McGraw-Hill Education Pvt. Ltd.: New York, NY, USA, 2011; p. 7.55.
31. *The USC-SIPI Image Database*, 2009. Available online: http://sipi.usc.edu/database/ (accessed on 7 January 2019).

32. Ahmed, N.; Asif, H.M.S.; Saleem, G. A Benchmark for Performance Evaluation and Security Assessment of Image Encryption Schemes. *Int. J. Comput. Netw. Inf. Secur.* **2016**, *8*, 18–29. [CrossRef]
33. Ye, Z.; Yin, H.; Ye, Y. Quantitative Comparison of Discrete Wavelet Chaotic Watermarking and Dual Chaotic Watermarking. In Proceedings of the 2018 4th International Conference on Computing Communication and Automation (ICCCA), Greater Noida, India, 14–15 December 2018; pp. 1–6.