Secure and robust digital image watermarking scheme using logistic and RSA encryption

Yang Liu, Shanyu Tang*, Ran Liu, Liping Zhang, Zhao Ma

*Corresponding author: Professor Shanyu Tang
Chair Professor of Information Security
(E-mail: Shanyu.Tang@uwl.ac.uk;
Tel: +44 (0)20 8231 2948;
Fax: +44 (0)20 8231 2402)

Abstract In the era of big data and networking, it is necessary to develop a secure and robust digital watermarking scheme with high computational efficiency to protect copyrights of digital works. However, most of the existing methods focus on robustness and embedding capacity, losing sight of security or requiring significant computational resources in the encryption process. This paper proposed a new digital image watermarking model based on scrambling algorithm Logistic and RSA asymmetric encryption algorithm to guarantee the security of the hidden data at the foundation of large embedding capacity, good robustness and high computational efficiency. The experiments involved applying the encryption algorithms of Logistic and RSA to the watermark image and performing the hybrid decomposition of Discrete wavelet transform (DWT) and Singular Value Decomposition (SVD) on the host image, and the watermark was embedded into the low-frequency sub-band of the host. The values of PSNR and NCC were measured to estimate the imperceptibility and robustness of the proposed watermarking scheme, and the CPU running time was

recorded to measure the complexity of the proposed main algorithm in execution time. Experimental results showed the superiority of the proposed watermarking scheme.

Keywords: Image watermarking, DWT, SVD, Logistic, RSA.

1. Introduce

In the era of rapid development of digitalization and network technology, information sharing becomes much easier. A noteworthy fact is that, multimedia objects stored in the digital format are subject to cyber attacks in an unsecure public channel. First of all, digital media can be easily copied and re-disseminated by a cyber citizen when spreading in a public channel. The process costs low and the information is transmitted with no degradation, but it is unlikely to guarantee that the behavior is legitimate. In addition, digitalized media are easily manipulated by the use of computers. For example, one cracker could selectively crop and integrate part of a digital work into her or his own one, ignoring the copyright of the original work. It is obviously to see that encryption is an applicable way to make digital media secure. However, if the data is decrypted viciously into its original form, it will put in danger once again.

Taking the above analysis into consideration, some researchers have found that digital watermarking technology can solve the security problem to a certain degree. The basic idea of this kind of technology is to use copyright information, data block header information or time synchronization mark data as watermark information, and

embed them into a host signal, such as image, audio, or video and the likes. In this way, the watermark information is not transmitted in another digital channel, but transmitted as part of the host signal. Apart from protecting copyright of digital works, a qualified watermarking system requires that the process of embedding some extra data should bring the least degradation to the host signal, which means that the host data should not be changed visually after inserting the watermark information. Moreover, the watermarking system needs to be robust against possible cyber attacks. The extra data should not be removed or changed after the watermark information experiences a certain attack in a network environment. But if the host object is changed, the watermark data will be lost.

Digital watermarking is generally divided into spatial domain watermarking and frequency domain watermarking. Depending on the embedding domain chosen, the degree of robustness and invisibility of systems, the data embedding capacity has an effect on the robustness and imperceptibility of watermark (Verma & Jha, 2015). Usually, embedding data in frequency domain works better than that in spatial domain for the better robustness to resist multiple signal processing manipulations and attacks. In order to solve the ambiguous problem of watermark, YAVUZ and TELATAR (2007) applied three-dimensional discrete wavelet transform (DWT) to a host image, and four sub-bands of low-low (LL), high-low (HL), low-high (LH) and high-high (HH) were obtained. The watermark image was decomposed by singular value decomposition (SVD). The singular values (SV) of watermark was then embedded into that of the sub-bands LL and HL from host image, and left singular matrix (U) of

the watermark image was embedded into LH and HH of the host image. Mukherjee and Pal (2012) got discrete cosine transform (DCT) coefficients from the process of DCT transform of host image, and formed a new host image. The watermark was embedded into the new image by the course of SVD decomposition and recombination. Wang, Li and Kang (2015) divided host image into sub-blocks of $m \times n$, and performed DCT transform on each block. The watermark was then condensed and embedded into the intermediate frequency coefficients of the host image together with the decoded secret key. For the sake of security of digital watermark, watermark can be scrambled, and the Arnold's method is widely used. Sujatha and Sathik (2010) obtained the minimum values from each sub-block of the host image, scrambled those values three times with Arnold transform, and constructed a binary watermark. After DWT performed on the host image, watermark information is embedded into high frequency coefficients of the host. The sub-blocks of host image was processed by DCT and further quantized in (Han, Yang, Zhi, 2011). The watermark image was then embedded in the selected DC coefficients after scrambled by Arnold procedure. Prasad (2013) extracted parts of data from the spatial domain of the host image as a digital watermark, and employed one level DWT on the host image. Watermark was scrambled by Arnold and embedded in high frequency sub-band of the host image. Saikrishna and Resmipriya (2016) separated host image into two texture regions of white and black, decomposed them with two levels of DWT, and scrambled watermark with Arnold. The scrambled data was embedded in sub-bans of the white textured area. Niu, Cui, Li and Ding (2016) decomposed host

image with two level DWT (2-DWT) and applied SVD to the sub-bands of low frequency of the host and the scrambled watermark, and got the watermarked image by use of adding their singular values. Sikder, Dhar and Shimamura (2017) proposed a novel watermarking technique in which a host image was performed by slant transform and lower upper decomposition successively. The extra data were encrypted by Arnold function and then inserted into the obtained upper triangular matrix to resist some common image attacks. Those works above can reach robustness to a certain degree and maintain the visual quality of watermarked image. But the transformation cycle of Arnold is not long, if attackers carried out a limited times of scrambling process continuously, they could restore the original image. That is, Arnold has a low degree of security for its small size of secret key space. To achieve a higher level security, the works in (Kishore, Venkatram, Sarvya, & Reddy, 2014; Saha, Pradhan, Kabi, & Bisoi, 2014; Ray, Padhihary, Patra, & Mohanty, 2015; Patel, P. & Patel, Y., 2015) all encrypted the watermark image with RSA data encryption algorithm, which raises a new issue that image encryption with RSA is time consuming.

Some researchers have made use of technique of joint fingerprinting and decryption (JFD) to save the computational time. Kundur and Karthik (2004) used JFD for media encryption and fingerprinting in the area of digital rights management. The fingerprinting process was done on the receiver side, thus the media was encrypted once before being sent to users to achieve the purpose of saving the computational time. Similarly, in the method proposed by Czaplewski and Rykaczewski (2014), the host image was encrypted by a matrix multiplication based

block cipher algorithm, and the encrypted image was then transmitted to different uses. After decrypting the received image, the process of fingerprint embedding was conducted on the coefficients of discrete cosine transform domain. Czaplewski (2016) used the method of quaternion algebra to rotate and translate the components of the host image in a three-dimensional color space, and encrypted the image at the source. Receivers designed different decryption keys according to their own fingerprints, and inserted the fingerprints into the image decrypted. The advantage of the JFD algorithm is that the watermark embedding process is placed at the receiving end, without considering the robustness of the watermark subject to various attacks when transmitted on the network, resulting in less time consumption on the receiving side. However, the data embedding capacity is limited in those algorithms, and their security needs to be improved. They utilise difference information entropy between encryption keys and decryption keys to form digital fingerprints, leading to a situation that a large amount of difference information will cause serious distortion to the host image. Thus, the embedded fingerprint is relatively small. In addition, the technique of symmetric encryption is applied in their algorithms although the keys used in encryption and decryption are different, and the distributor must know each user's decryption key to complete the process of copyright authentication. So the security risk arises when processing secret key management and distribution.

To address the above mentioned issues, we attempted to lower the consuming time as well as to improve the security of watermark apart from ensuring high robustness. Subsequently we proposed a new image watermarking scheme based on

DWT, SVD, Logistic and RSA. The rest of this paper is organized as follows. In section 2, we discuss the principle of DWT, SVD, Logistic and RSA. In section 3, we propose our new watermarking scheme including embedding and extraction process. In section 4, experimental results and discussion are presented. In section 5, we briefly conclude what we contribute.

2. Preliminaries

2.1 Discrete Wavelet Transform

Discrete wavelet transform (DWT) is a process of multi-scale and spatial-frequency decomposition to an image. In the DWT-based watermarking scheme, DWT is used to decompose an aimed image into four types of sub-bands which are LL, HL, LH and HH. LL is low frequency component, and has a low resolution, representing approximate information of an image. The other three are horizontal high-frequency part, vertical high-frequency part, and high frequency part, respectively, and their resolutions are high, representing detailed image information. One or more sub-bands can be used to embed watermark information. DWT is an efficient frequency model for HVS, which is widely applied in the field of image compression and enhancement. Wavelet transform has the characteristics of multi-resolution, so hierarchical display is a feasible application of continuous image transmission. In watermark applications, it has less computation complexity when watermark is embedded hierarchically or nested by using DWT technology.

DWT-based techniques reflect better robustness against various attacks compared with watermarking in spatial domain (Verma & Jha, 2015). Another feature of DWT is the ability to select different filter banks for the required broadband. The commonly used filters are Haar, Daubechies, Coiflets and Biorthogonal, and adjustments can be easily done when necessary. With regard to a multidimensional signal, the ideal of DWT is to split the signal into high and low frequencies, and the low frequency part is further split up into high and low frequencies until the original signal is completely decomposed, as shown in Figure 1. After the process of inverse wavelet transform (IDWT), the image can be reconstructed and restore from the DWT coefficients.

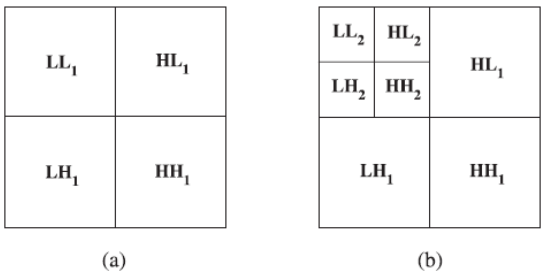


Figure 1 (a) 1-level DWT, (b) 2-level DWT

2.2 Singular Value Decomposition

Singular Value Decomposition (SVD) is commonly used in linear algebra (Verma & Jha, 2015). This mathematical tool can be used in many applications, such as signal or image processing, including digital watermarking. In a SVD-based watermarking technique, SVD usually acts on the host image, or the host image is first divided into many small blocks and then those blocks are decomposed with SVD

to get singular values, which are used to embed watermark information. There are some advantages to use this kind of decomposition in digital watermarking: the size of the SVD coefficients is fixed, the singular values can represent the basic algebraic features of an image, and the singular values are unlikely to change noticeably when the image is slightly disturbed (Thakkar & Srivastava, 2017). The formula of the decomposition is as follows:

$$I = U \cdot S \cdot V^T,$$

where I is the matrix of $m \times n$ corresponding to a image, U is the matrix of $m \times m$, V is the matrix of $n \times n$, S is the diagonal matrix with the sane size of I, and T is the matrix transformation coefficient.

2.3 Logistic

Logistic map is a kind of one-dimensional chaotic mapping (Pareek, Patidar, & Sud, 2006), which is widely used in digital communication security, multimedia data security and other fields. The formula of the map is initially developed from the demographic, featured with a simple form but a very significant regularity. Its definition is:

$$X(k+1) = u * X(k) * [1 - X(k)], \quad k = 0,1,...,n, \quad X(k) \in (-1,1), \quad u \in (0,4) \qquad (1).$$

In this formula, $X(k)$ is the mapping variable and u is the system parameter. When the following two conditions are met, the function of logistic works in a mixed state, that is, in a disorder and unpredictable way.

$$0 < X(0) < 1,$$

$3.5699456 < u < 4$.

The basic idea of the formula is that iterating a given initial value n times, n values are produced, $X(1), X(2),..., X(n)$, which is a one-dimensional chaotic sequence. When an image of $m \times n$ is encrypted, it is necessary to iterate $m \times n$ times to obtain a one-dimensional sequence. The sequence is then normalized and a new sequence in range of (0, 255) is generated. Finally, the new one is converted into a two-dimensional matrix, which is the encrypted image matrix. The secret key used is $[X(0), u]$.

It could achieve a high level of security when encrypting an image with Logistic map for the sequences generated by this encryption function with features of aperiodic, non-convergent and irrelevant. Moreover, the function is very sensitive to initial values, that is, even if initial conditions are quite close, the iteration results are not the same, and the number of the uncorrelated chaotic sequences is very large. So it is difficult for attackers to deduce the exact initial condition of the chaotic system from a finite length sequence.

2.4 RSA

Public key encryption system makes use of an asymmetric encryption mechanism (Rojat, 2012), protecting encrypted data by using a pair of keys. RSA is a typical algorithm used in this domain. It also belongs to block cipher domain. The details of the process of its key generation are shown below：

Step 1. Select two prime numbers randomly, p , q .

Step 2. Calculate one secret key member N and Euler's totient function $\varphi(N)$:

$$N = p * q, \quad \varphi(N) = (p-1) * (q-1) \tag{2}.$$

Step 3. Choose an encryption key at random, two conditions below need to be met:

$$1 < e < \varphi(N), \quad \gcd(e, \varphi(N)) = 1 \tag{3},$$

where the logic expression presents the great common divisor between e and $\varphi(N)$, equals one and further illustrates that e and $\varphi(N)$ are co-primes.

Step 4. Calculate the decryption key d with the formula below:

$$e * d = 1 \bmod \varphi(N), \quad 0 \le d \le N \tag{4}.$$

In this public key system, there are two kinds of secret key, public key and private key, which are given in the form of even, (e, N) and (d, N) respectively. The public key is known to all and the private key is secret. In the process of data transmission, the sender knows the public key of the recipient, and then encrypts the message with the public key. After obtaining a cipher text, the sender transmits the cipher text to the receiver. Then the receiver processes the cipher text with its own private key and gets the plain text message. Security of the RSA encryption system is guaranteed. Although attackers may know the public key for e and N being open to the public, e is a random number and N is a great number. Thus it is unlikely for attackers to get the values of p and q by large integer factorization which is a NP-hard problem.

3. The Proposed Algorithm

In this section, we present a new watermarking scheme in Figure 2, which contains two main processes of watermark embedding and watermark extraction. Watermark preprocessing and embedding procedure are performed on the sending side, and the watermarked image is transmitted to the receiver over the internet. At the receiving end, watermark extraction and recovery is carried out. The blocks in the figure represent the corresponding algorithm operations. Arrow symbols pointing to them means that some words nearby are the inputs of the operation. The main processes of our proposed scheme are that watermark was first scrambled by Logistic, and the scrambling parameters were encrypted by RSA and then transmitted through network. In the process of watermarking embedding, one level discrete wavelet transform was applied to the host image and a low-frequency sub-band was then obtained. The sub-band was further processed by Singular value decomposition. Plus singular value of the sub-band and the scrambled watermark, and new singular value was acquired. In the addition operation, a scaling factor was chosen to control the embedding strength of watermark, and an appropriate strength could be traded off between imperceptibility and robustness. This new value was decomposed once again to get a new singular value which was used to reconstruct a low-frequency sub-band. Finally, the watermarked image was formed by making using of the new sub-band after the process of inverse discrete wavelet transform. The details of watermark embedding and extraction are depicted in Sections 3.1 and 3.2.

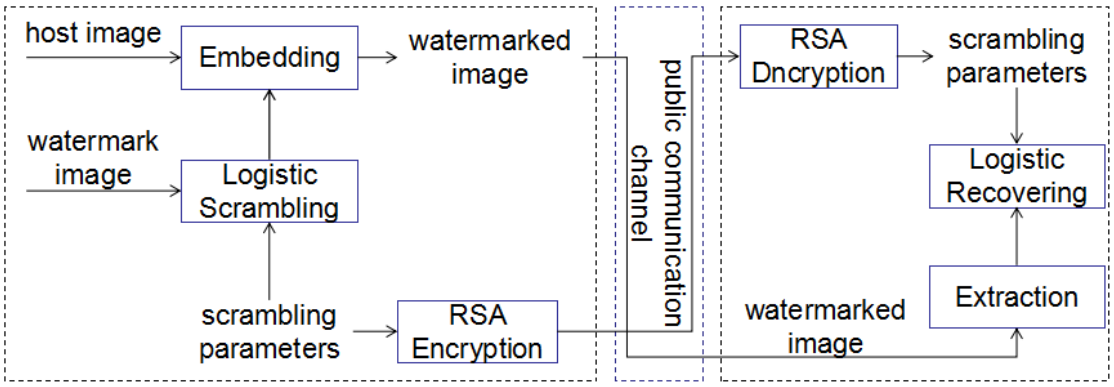


Figure 2. Proposed secure watermarking scheme

3.1 Watermark Embedding

In the procedure of watermark embedding, a watermark image was first scrambled by using a Logistic algorithm before being embedded into the transformed domain of the host image. Assuming that the pixel information of watermark can be denoted as m-by-m matrix A, A is reshaped to a 1-by-m*m matrix T, whose elements are taken column-wise from A. $x(i)$ is a one-dimension array with a length of m^2, and given a initial $x(0)$ and a system parameter u, a chaotic sequence is generated in the process of iteratively calculating $x(0)$ and u by the formula given in Section 2.3, and elements of the sequence is saved in $x(i)$ successively. The array is then normalized to become a new sequence in range of (0, 255) through the following formula:

$$x(i)=\begin{cases}\mathrm{mod}(round(\alpha*(\beta*x(i)-round(\beta*x(i)))),256) & ,round(\beta*x(i))<\beta*x(i)\\ \mathrm{mod}(round(\alpha*(1-\beta*x(i)-round(\beta*x(i)))),256), & round(\beta*x(i))>\beta*x(i)\end{cases}$$

(5),

where α and β are two amplification factors, and $i\in\{1,2,3,...,m^2\}$. The function of round(X) rounds the element X to its nearest integer. Function mod(a,b) returns the remainder after a is divided by b. $B(i)=bitxor(x(i),T(i))$ (6), where the bitxor()

function performs a bitwise exclusive OR on $x(i)$ and $T(i)$ and yields $B(i)$ and the scrambled version of $T(i)$. The 1-by-m*m matrix B is further reshaped to m-by-m matrix C, which is the encrypted watermark.

To improve the security of watermark, the initial parameters, $x(0)$ and u, are further encrypted by RSA, a typical asymmetric encryption algorithm, which is described in Section 2.4. Two prime numbers, p and q, combined with $x(0)$ and u which are viewed as plaintext, are inputs of the RSA function, the outputs of which are the ciphertext that has nothing to do with the inputted parameters. The private key (e, N) and the public key (d, N) are calculated by using Formula (2) and (4). After the encrypted watermark is inserted into the host image, the watermarked image is transmitted to the receiving end along with the ciphertext over the Internet. The private key is then used in the process of watermark extraction to recover a clear version of watermark. The specific steps of watermark insertion are described below.

Step 1. A number-pair of $[X(0), u]$ was chosen as scrambling parameters, and then encrypted by employing RSA algorithm to get the cipher text R, where R = $[X(0), u]^e \bmod N$, and (e, N) was the public key.

Step 2. Gray-scale watermark image W was scrambled by the Logistic algorithm with the scrambling parameters, and scrambled watermark W_d was obtained.

Step 3. The gray-scale host image C was decomposed into four sub-bands C^i by DWT, where $i = LL$, HL, LH, HH.

Step 4. SVD was performed on LL, $U_C \cdot S_C \cdot V_C = SVD(LL)$.

Step 5. Computed a new singular value S_{new} by adding S_C and the scrambled

watermark together with a scaling factor α, $S_{new} = S_C + \alpha \cdot W_d$.

Step 6. Applied SVD to the new singular value, $U_W \cdot S_W \cdot V_W = SVD(S_{new})$.

Step 7. Reconstructed a new low-frequency approximate coefficient LL_{new}, and $LL_{new} = U_C \cdot S_W \cdot V_C^{'}$.

Step 8. Obtained the watermarked image C_W by performing inverse DWT with the modified approximate coefficient.

3.2 Watermark Extraction

To construct the watermark, the operation of DWT was applied to the watermarked image, and the low frequency approximate coefficient of which was further decomposed. The scrambled watermark was obtained under the utilization of the original host image and the newly formed singular value S_{new}. The detailed steps are as follows:

Step 1. The original host image C and the watermarked image C_W were respectively decomposed into four sub-bands C^i and $C_W{}^j$ by DWT, where $i = LL, HL, LH, HH$ and $j = LL_W, HL_W, LH_W, HH_W$.

Step 2. SVD was performed on LL_W, $U_{CW} \cdot S_{CW} \cdot V_{CW} = SVD(LL_W)$.

Step 3. Reconstructed a new low-frequency approximate coefficient LL_{new1}, and $LL_{new1} = U_W \cdot S_{CW} \cdot V_W^{'}$.

Step 4. Obtained the scrambled watermark image by the formula of $W_{dnew} = (LL_{new1} - S_C)/\alpha$.

Step 5. Decrypted the cipher text R to get the plain text of scrambling parameters

by using the private key (d, N), $[X(0), u] = R^d \bmod N$.

Step 6. The Logistic algorithm was utilized to recover the watermark image W with the scrambling parameters.

4. Experimental results and discussion

A series of experiments were conducted to validate the effectiveness of the proposed watermarking scheme, which was further compared with the related methods. Table 1 shows the CUP running time in the process of watermark encryption using Saha, Pradhan, Kabi and Bisoi (2014)'s scheme, Kishore, Venkatram, Sarvya and Reddy (2014)'s scheme, and our proposed scheme. Saha et al.'s scheme represents a class of algorithms that use the asymmetric encryption algorithm RSA for the watermark image. They yielded higher security compared with the methods based on symmetric key encryption. However, encrypting the image with RSA consumes more time, which is confirmed later in our experiments. Figures 3-5 and Table 2 are the results of the proposed scheme with a scaling factor of 0.005, and Table 3 is the result with the factor of 0.05. Table 3 compares NCC values between the proposed scheme with the method proposed by Kishore, Venkatram, Sarvya and Reddy (2014) and the method proposed by Saha, Pradhan, Kabi and Bisoi (2014) in case of various types of attacks. Kishore et al.'s method is a typical frequency domain watermarking algorithm that embeds watermark into low-frequency sub-band after applying DWT to the host image, but the use of hybrid decomposition can make up for some of its

flaws.

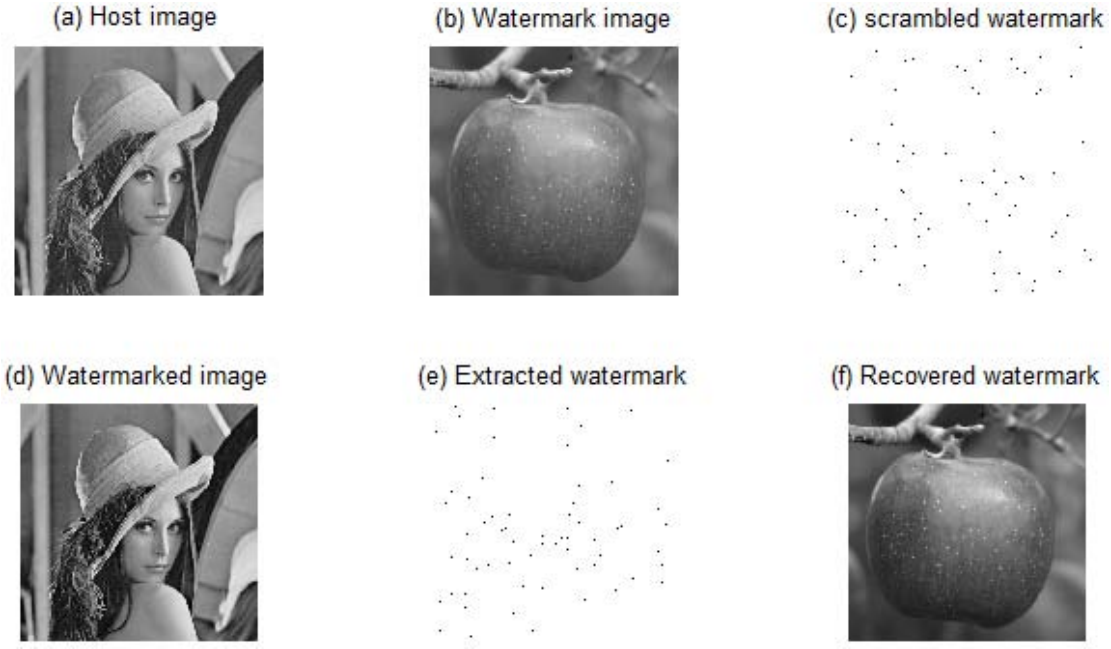


Figure 3. (a) Host image, (b) Watermark image, (c) Scrambled watermark, (d) Watermarked image, (e) Extracted watermark, and (f) recovered watermark

As shown in Figure 3, the gray-scale image lena of 512×512 was taken as the host image, and the watermark was an apple image of 256×256. From Figure 3(c) to Figure 3(f), the details of this proposed watermarking scheme are depicted. Figure 3(c) shows the scrambled watermark with the scrambling parameters of 0.2 and 3.6. Figure 3(d) shows the watermarked image based on the combination of one level discrete wavelet transform and singular value decomposition and the scaling factor was 0.005. Figure 3(e) shows the extracted watermark from the watermarked image, and the extracted image is in a chaotic state. Figure 3(f) shows the watermark recovered from the extracted image with the same scrambling parameters.

Figures 4(a), (b), (c), (d), (e), and (f) are the histograms of the original host image, watermark image, scrambled watermark, watermarked image, extracted watermark image, and recovered watermark, respectively. These histograms reflect the pixel distribution of corresponding images, where the horizontal axis represents different level grayscales of an image, and the vertical axis the number of pixels of a certain grayscale. Normally, if an image is changed, its pixel distribution will change, too. Taking Figure 4(a) as an example, the composition of histogram of the host covered a wide range of gray levels as a result of the high contrast of the host image. The results of Figs. 4(a) and (d) indicated a small impact on the host image from the extra data, indicating the imperceptibility of the proposed scheme. Figs. 4 (c) and (e) show that the watermark image was scrambled with a high degree, and Figs. 4 (b) and (f) show a high similarity between the original watermark and the recovered watermark image.

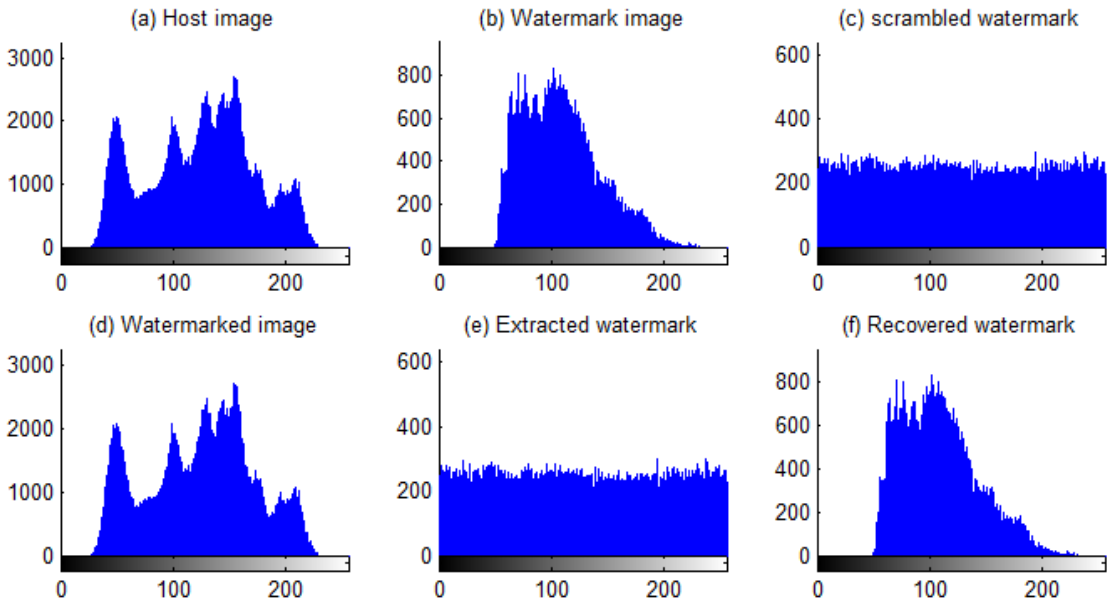


Figure 4. Hist of (a) host image, (b) watermark image, (c) scrambled watermark, (d) watermarked image, (e) extracted watermark, and (f) recovered watermark

Table 1 compares the elapsed time in RSA encryption and decryption of the proposed scheme and scheme of Saha, Pradhan, Kabi and Bisoi (2014) and Kishore, Venkatram, Sarvya and Reddy (2014). The prime numbers of p and q were randomly chosen for calculating n, one of secret keys. Three sets of numbers of this kind were taken to test the CUP running time. As shown in Table 1, the encryption and decryption processes using Saha et al and Kishore et al.'s methods need more time than the proposed scheme. As the volume of encrypted object increases, the time required for RSA encryption increases, too. A watermark image is a data matrix, and it takes time to encrypt it. The proposed scheme first scrambles the watermark and then encrypts the scrambling parameters with RSA. Scrambling parameters can be regarded as a string of several characters. So the proposed scheme is less time-consuming.

Table 1. CUP running time (Second)

Encryption Scheme	p	q	n	d	Encryption Time	Decryption Time
Proposed scheme	23	28	667	109	0.0156	0.0624
	107	113	12091	6927	0.0468	0.0468
	233	281	65473	21041	0.0321	0.0321
Saha et al. (2014)	23	28	667	493	4.9688	19.7031
	107	113	12091	4749	4.2813	170.9375
	233	281	65473	36089	4.6094	1.2878e+03

	23	28	667	287	4.8572	17.9268
Kishore (2014)	107	113	12091	4675	4.0791	165.7437
	233	281	65473	3267	4.4218	1.1783e+03

Two commonly used methods, PSNR and NCC, were used to evaluate the performance of the proposed watermarking scheme.

The PSNR is short for Peak Signal to Noise Ratio, which is used to measure the peak error between the cover image and the image into which extra information has been embedded, and its formula is as follows:

$$PSNR = 10\log 10\frac{255^2}{MSE},\quad MSE = \frac{1}{m \cdot m}\sum_{i=1}^{m}\sum_{j}^{m}(X_{ij} - \tilde{X}_{ij})^2 . \tag{1a}$$

In this mathematical expression, MSE refers to the mean square error and X_{ij} is the matrix of original host image, and $\tilde{X}_{ij}$ is the matrix of the watermarked image. The unit of PSNR is dB. And the greater the PSNR value, the less distortion on the visibility of an image.

The NCC is short for Normalized Cross-Correlation, with which, the realization of quality evaluation of the extracted data become easier. NCC can be utilized to measure the similarity degree between the original watermark and the watermark extracted from the watermarked image, and its formula is given below:

$$NCC = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}W(x,y)\cdot\tilde{W}(x,y)}{\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}W^2(x,y)}\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}\tilde{W}^2(x,y)}} . \tag{2a}$$

In this formula, $W(x,y)$ is the matrix of the original watermark and $\tilde{W}(x,y)$

is the matrix of the extracted watermark. The values of NCC range from 0 to 1, and bigger the value, the better performance of the watermarking scheme.

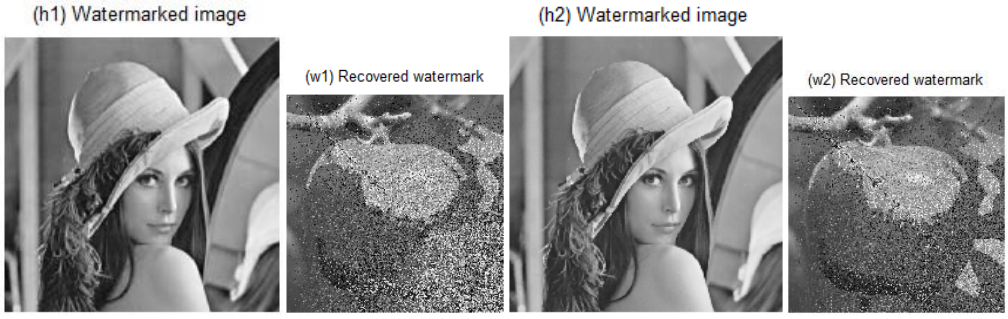
Table 2. PSNR and NCC values

Attacks	PSNR (db)	NCC
No attacks	50	1
Mean Filtering (3×3)	41.04	0.7791
Mean Filtering (5×5)	38.73	0.6738
Mean Filtering (7×7)	37.57	0.6146
Median Filtering (3×3)	43.10	0.8470
Median Filtering (5×5)	40.53	0.7672
Median Filtering (7×7)	39.45	0.6890
Rotation (15)	31.69	0.8419
Rotation (30)	31.32	0.8256
Rotation (45)	31.23	0.8304
Rotation (90)	31.71	1
Rotation (135)	30.79	0.8304
Rotation (180)	31.54	1
Gaussian Noise (0.001)	32.80	0.7923
Salt & Pepper Noise (0.02)	48.10	0.7981
Salt & Pepper Noise (0.1)	41.07	0.7564
Crop (100,100)	----	0.9405
Crop (150,150)	----	0.9336

Crop (200,200)	----	0.9220
Crop (250,250)	----	0.8471
Crop (300,300)	----	0.7538

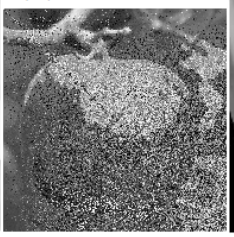
The PSNR and NCC values measured after various attacks having been imposed on the watermarked image were listed in Table 2. In the third row in Table 2, after the watermarked image was attacked by Mean Filtering with a window area of 3x3, the watermark was extracted from the image that had suffered the attack, and then, values of 41.04 and 0.7791 were obtained by applying formulas (1a) and (2a).

Figure 5 shows the watermarked image and recovered watermark image after various attacks, such as (h1) and (w1) are the results of Mean filtering of 3×3; (h2), (w2) and (h3), (w3) Media filtering of 3×3 and 5×5, respectively; (h4~h9) and (w4~w9) Rotation angles of 15, 30, 45, 90, 135, and 180; (h10), (w10) Gaussian noise densities of 0.001; (h11), (w11) and (h12), (w12) Salt & Pepper of 0.02 and 0.1, respectively; (h13~h16) and (w13~16) Crop attacks with an area of 100×100, 150×150, 200×200, and 250×250.



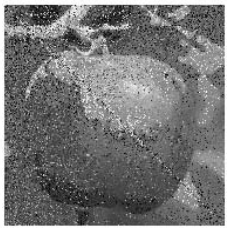
(h3) Watermarked image

(w3) Recovered watermark



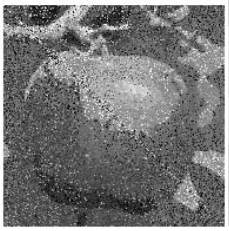
(h4) Watermarked image

(w4) Recovered watermark



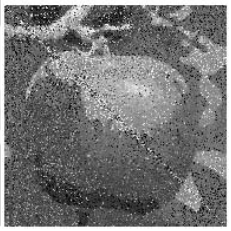
(h5) Watermarked image

(w5) Recovered watermark



(h6) Watermarked image

(w6) Recovered watermark

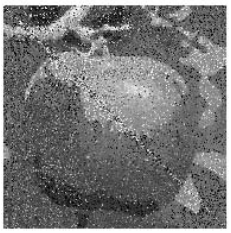


(h7) Watermarked image

(w7) Recovered watermark

(h8) Watermarked image

(w8) Recovered watermark

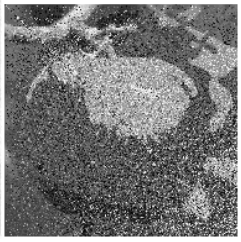


(h9) Watermarked image

(w9) Recovered watermark

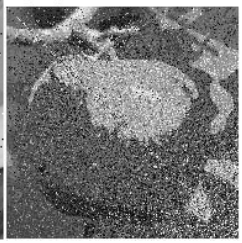
(h10) Watermarked image

(w10) Recovered watermark



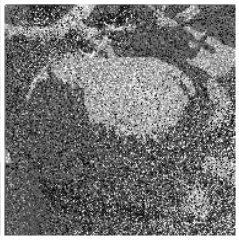
(h11) Watermarked image

(w11) Recovered watermark



(h12) Watermarked image

(w12) Recovered watermark



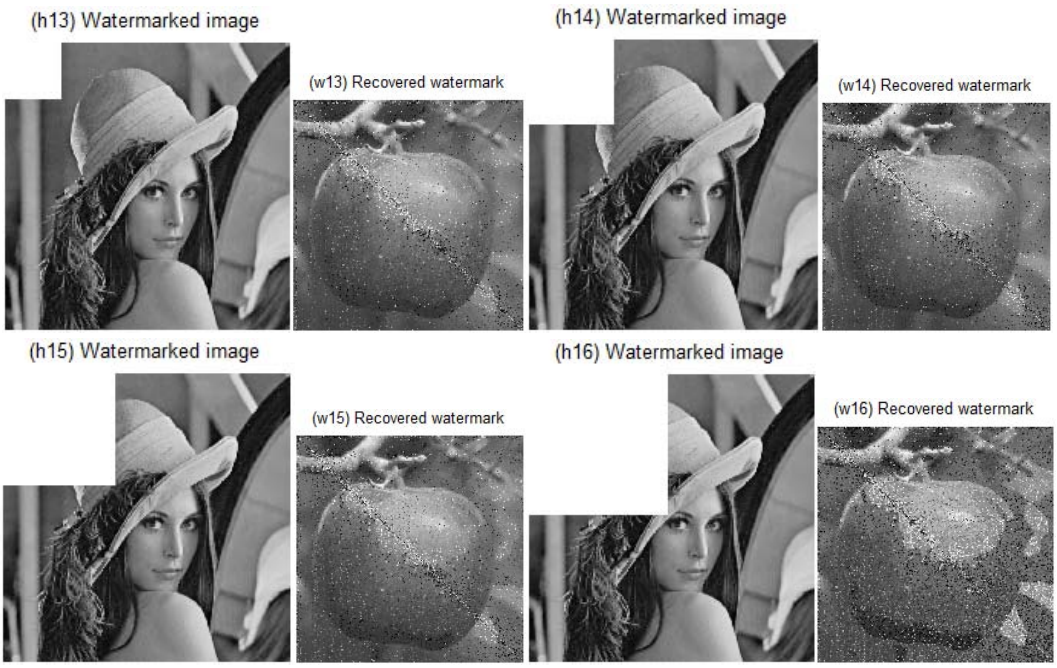


Figure 5. (h1~h16) Watermarked images and (w1~w16) Recovered watermark after various attacks

Table 3 shows the robustness comparisons of the proposed watermarking scheme with the scheme of Kishore, Venkatram, Sarvya and Reddy (2014) and the scheme of Saha, Pradhan, Kabi and Bisoi (2014) at the same embedding intensity of 0.05. By considering all the attacks, our proposed scheme is more robust to various attacks than other two schemes. In the algorithm of Kishore et al, watermark was embedded into the low-frequency sub-band of host image after being processed by DWT. And Saha et al hid watermark in the middle frequency band of the host. In the proposed algorithm, the sub-band was further decomposed by SVD and watermark was embedded into the singular value of the sub-band. As singular value has the property of geometric invariance, the low frequency sub-bands of DWT is not sensitive to various noises, making the proposed algorithm performs better when it is against

various types of manipulation.

Table 3. Comparisons in NCC values

Attacks	Proposed scheme	Kishore et al. (2014)	Saha et al. (2014)
Mean Filtering (3×3)	0.8156	**0.9765**	0.6783
Median Filtering (3×3)	**0.8620**	0.7892	0.7817
Rotation (45)	0.8067	**0.9642**	**0.9507**
Rotation (90)	**1**	0.9519	0.9493
Rotation (135)	0.8067	**0.9396**	**0.9412**
Rotation (180)	**1**	0.9273	0.9138
Gaussian Noise (0.001)	**0.8063**	0.6073	**0.8948**
Gaussian Noise (0.005)	**0.8049**	0.6050	**0.8369**
Gaussian Noise (0.01)	**0.8032**	0.6027	0.7859
Salt & Pepper Noise (0.1)	**0.7615**	0.6004	0.6874
Crop (100,100)	**0.9560**	0.7322	0.8146

Image compression is a common image processing method for saving the cost of network transmission, because it can reduce the size of the original image. Figure 6

shows the results of compression attacks on different watermarking schemes proposed by Mukherjee and Pal (2012), Wang, Li and Kang (2015), YAVUZ and TELATAR (2007) and us. We tested four kinds of JPEG compression ratios, 60%, 70%, 80%, and 90%, respectively. To be more precise, the ratio of 90% means the volume of the tested image was compressed to ninety percent of its original state, and the compression degree of 90% is lower than that of 80%. As the comparisons show, when the compression intensity increases, the NCC values get smaller; that is, the similarity between the original watermark and the extracted one gets smaller. Another fact should be noted is that values in the red curve in this figure are bigger than that in the other three curves, indicating that the proposed algorithm is more robust to compression attacks.

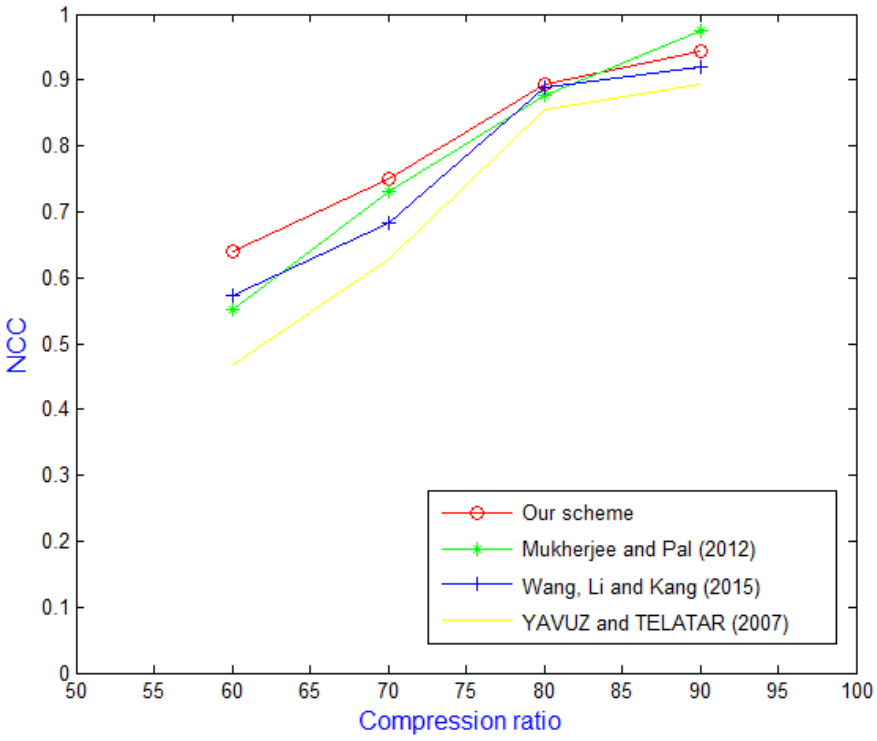


Figure 6. Compression attacks

To further evaluate the performance of the proposed algorithm, computer simulations of collusion attacks were conducted, and testing results is shown in Figure 7. In order to facilitate copyright infringement tracking, in general, the distributor of digital works selects different Logo information according to different authorised users. That is to say, the same host images of different copied versions are embedded with different watermarks. We set three, five, seven and nine attackers, respectively, to test four algorithms. Take the first three attackers case for instance, three different watermark images with the same size of 256×256 were embedded into the host image lena, and three watermarked lenas were obtained. Those lenas were then superposed and averaged to form an averaged lena. A new watermark was extracted to compare with the original three watermarks, and the corresponding three NCC values were calculated. By using these three values, an average value was obtained and shown in Figure 7. As the figure shows, when the number of attackers increased, the NCC value became smaller. All the four schemes have the capability to resist average collusion attacks to a certain degree. Because the watermark image is encrypted by scrambling algorithm beforehand, the scrambled watermark is not affected much by the collusion attack. Especially, Figure 7 shows the superiority of the proposed scheme in terms of resisting collusion attacks.

We also test the proposed scheme with a number of colour images. The host image is first partitioned into three channels, red (R), green (G), and blue (B). Then, the embedding process depicted in Section 3.1 is applied to the channel B. The newly obtained B is combined with the prior elements, R and G, to form a new image. To

test the performance of the proposed scheme, six commonly used color images of 512×512 are specified as host objects, and one gray-scale image is the watermark, as shown in Figure 8.

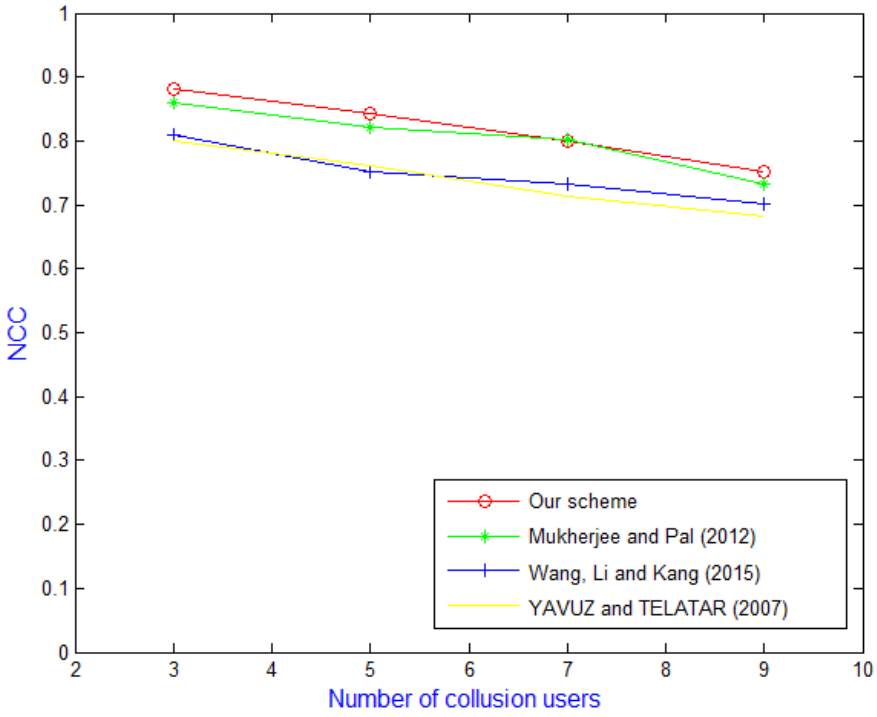


Figure 7. Collusion attacks

Generally, the embedding capacity affects the imperceptibility and robustness of the watermarked image. Just as shown in Table 4, the PSNR values of the color images (CPSNR) were determined under the condition that the embedding strength λ was assigned to 0.05, 0.1 and 0.5, respectively, providing references for users with different requirements. In this table, the first column lists the name of the watermarked images, and the last row presents the corresponding average CPSNR of the images in each state of λ. As can be seen, the higher the strength of the watermark is, the smaller the CPSNR becomes, which means less imperceptibility.

This is probably because the embedded information can be treated as an extra signal noise. So users can choose suitable strength to trade off between watermark capacity and imperceptibility. The evaluations of the experimental results show that λ with a value of 0.5 is a good choice.

Figure 8. Color images. (a) lena, (b) airplane, (c) saiboat, (d) peppers, (e) baboon, (f) house.

Table 4. PSNR of these watermarked images with different scaling factors.

Scaling factors / Images	0.05	0.1	0.5
lena	48.03	46.18	42.31
airplane	47.93	47.21	43.73

sailboat	44.69	43.14	40.59
peppers	46.20	45.51	41.37
baboon	44.75	42.81	39.82
house	44.26	43.39	38.68
average	45.98	44.70	41.08

Apart from ensuring good imperceptibility, it is necessary to test the robustness of the proposed scheme. The watermarked image was separated into three channels, following by the extraction process described in Section 3.2, and the watermark was extracted from the channel B. Table 5 shows the NCC values between the original watermark and the extracted one after performing various attacks on the watermarked images, and the embedding scaling factor λ equals 0.1. Just as what is observed, it sees a good robustness to various attacks overall. It is noted that, for the same attack, the testing values of different images are diverse. For the reason that though the pixel sizes of two images are the same, their color distributions vary significantly; embedding the same watermark into different host images with the same size yields different impacts on their visual quality, and the quality of the watermarks extracted from different host images is varying after being exerted the same image processing.

Table 5. NCC values after various attacks.

Images / Attacks	lena	airplane	sailboat	peppers	baboon	house
Mean Filtering (3×3)	0.7574	0.7218	0.6837	0.7103	0.6916	0.6689

Median Filtering (3×3)	0.9517	0.9410	0.8738	0.9105	0.8871	0.8693
Gaussian Noise (0.01)	0.8546	0.8419	0.7867	0.8133	0.8041	0.7710
Salt & Pepper Noise (0.1)	0.8521	0.8458	0.7692	0.8316	0.7885	0.6912
Rotation	0.7529	0.7325	0.6753	0.7142	0.7030	0.6594
Crop (10%)	0.8673	0.8357	0.7914	0.8201	0.8086	0.7546
JPEG compression (10)	0.9217	0.8993	0.8431	0.8736	0.8707	0.8104

More comparative experiments were conducted in the same condition that the sizes of cover image and watermark image were 512×512 and 256×256 respectively, and the value of λ was 0.1. Table 6 shows the testing results for the cover image lena. We added four types of signal noise to test the robustness of the proposed scheme. Both Mean filtering and Median filtering with the sliding windows of 3×3 and 5×5, Gaussian noise with the jamming strengths of 0.001, 0.005, and 0.01, and Salt and peppers noise of 0.005, 0.1, 0.3 and 0.5 were added to watermarked lena, respectively. Apart from the conventional image processing, geometric transformations were also applied. In the process of attack simulation, the watermarked image was rotated 15, 30, and 45 degrees successively. Another attack was to crop the image from its upper-left corner with the sizes of 50×50, 100×100, 150×150, and 200×200. And JPEG compression with the compressive strengths of 10%, 20%, 30%, and 40% was also considered. As is observed in the table, the

proposed scheme outperforms the methods proposed by Saikrishna et al. (2016) and Han et al. (2011) in terms of the robustness of watermark. Taking cropping attacks as an example, we averagely disseminated the watermark information into the cover image, so the cropping part of image did not degrade the quality of the embedded data noticeably.

Table 6. Comparisons in NCC values

Manipulations	Our method	Saikrishna and Resmipriya (2016)	Han, Yang and Zhi (2011)
Mean Filtering (3×3)	**0.7574**	0.7348	0.6947
Mean Filtering (5×5)	0.6053	**0.6513**	**0.6372**
Median Filtering (3×3)	0.9517	**0.9682**	0.9274
Median Filtering (5×5)	**0.8628**	0.8607	0.7849
Rotation (15)	**0.8649**	0.8573	0.8618
Rotation (30)	**0.7529**	0.6947	0.7158
Rotation (45)	**0.7629**	0.7121	0.6895
Gaussian Noise (0.001)	**0.9437**	0.9429	0.8978
Gaussian Noise (0.005)	0.9029	**0.9173**	0.8436
Gaussian Noise (0.01)	**0.8546**	0.8176	0.7648
Salt & Pepper Noise (0.05)	0.9237	**0.9461**	**0.9341**
Salt & Pepper Noise (0.1)	**0.8521**	0.8273	0.8312
Salt & Pepper Noise (0.3)	**0.7875**	0.7648	0.7719
Salt & Pepper Noise (0.5)	**0.6792**	0.6486	0.6651

Crop (50,50)	**0.9673**	0.9347	0.9543
Crop (100,100)	**0.9567**	0.9138	0.8952
Crop (150,150)	**0.9276**	0.8750	0.8217
Crop (200,200)	**0.8657**	0.8139	0.6976
JPEG (40%)	0.6292	**0.6471**	0.6127
JPEG (30%)	0.8015	0.7938	**0.8146**
JPEG (20%)	**0.8738**	0.8673	0.8706
JPEG (10%)	0.9217	**0.9341**	**0.9357**

To quantify the watermarking property and to prove the advantage of less time consumption in the proposed scheme, an experiment was implemented on the platform of MATLAB 7.0 running on a PC with a CPU of Inter Core2 2.66 GHz and a memory chip of 4 GB. From the visual effects of Figure 3 and statistic data of Figure 4, it is noted that the proposed watermarking algorithm had a good imperceptibility. Table 2 and Figure 5 show the results for watermarked images and recovered watermark after various attack tests with a low embedding intensity and the mean NCC value was 0.8325, indicating a good visibility in the extracted watermark. When the intensity was increased to 0.05, as shown in Table 3, most NCC values of the proposed scheme were bigger than those of Kishore, Venkatram, Sarvya and Reddy (2014)’s scheme, and the mean NCC value was improved to 0.8566, illustrating that the proposed watermarking algorithm had good robustness. Apart from considering the properties of robustness and imperceptibility, the proposed scheme also focus on security and computational complexity. While preserving the security of RSA, we

attempted to reduce the time consumed by the scheme as much as possible, and have made several improvements. The comparisons as shown in Table 1 indicated that the proposed encryption process cost less time than Saha, Pradhan, Kabi, and Bisoi (2014)'s scheme, for the reason that the watermark image was directly encrypted by RSA encryption algorithm in the latter scheme but this process required a longer time to proceed, while the former scheme (proposed) was first to encrypt watermark with a Logistic algorithm that needs less time and then to encrypt the encryption parameters of Logistic with RSA, which not only guaranteed security of watermark but also reduced the time elapsed.

5. Conclusion

In this paper, an improved secure and robust digital watermarking scheme has been proposed. The embedding process includes embedding a gray-scale image into the singular value of low frequency sub-band of the host image. The combination of a image-scrambling algorithm and a secure data-encryption algorithm was used to improve the security of the proposed watermarking scheme. The scrambling parameters could be stolen when information is transformed on a public network, but the asymmetric encryption system can protect those parameters form being attacked by hackers. The simulation experiments demonstrated that the proposed scheme had taken the main performance of watermarking technique into consideration and outperformed other similar approaches, having better robustness, less encryption time

and large data embedding capacity.

In the era of big data, a huge amount of digital images need to be processed and then transmitted through the network full of various threats. The issues of computational refinement and data security attract more and more people's attention and the proposed scheme addresses these two points to some extent. In the future, guaranteeing the data security of image watermarking is an important direction and the use of asymmetric encryption method should be a good choice.

6. References

Czaplewski, B., Rykaczewski, R. (2014). Matrix-based robust joint fingerprinting and decryption method for multicast distribution of multimedia. Signal Processing, 111 (C), 150-164.

Czaplewski, B. (2016). Joint fingerprinting and decryption method for color images based on quaternion rotation with cipher chaining. Journal of Visual Communication and Image Representation, 40, 1-13.

Han, W., Yang, Y. & Zhi, H. (2011). Digital Watermark Encryption Algorithm Based on Arnold and DCT Transform. Electrical, Information Engineering and Mechatronics (pp. 613-621).

Kundur, D., Karthik, K. (2004). Video Fingerprinting and Encryption Principles for Digital Rights Management. Proceedings of the IEEE, 92(6), 918-932.

Kishore, P. V. V., VenKatram, N., Sarvya, Ch., & Reddy, L. S. S. (2014). Medical Image Watermarking using RSA Encryption in Wavelet Domain. International Conference on Networks & Soft Computing (pp. 258-262).

Mukherjee, S., & Pal, A. K. (2012). A DCT-SVD based Robust Watermarking Scheme for Grayscale Image. International Conference on Advances in Computing, Communications and Informatics (pp. 573-578).

Niu, Y., Cui, X., Li, Q., & Ding, J. (2016). A SVD-Based Color Image Watermark Algorithm in DWT Domain. Advanced Graphic Communications, Packaging Technology and Materials (pp. 303-309).

Pareek, N. K., Patidar, V., & Sud, K. K. (2006). Image encryption using chaotic logistic map. Image and Vision Computing, 24(9), 926–934.

Prasad, D. V. R. (2013). An Improved Invisible Watermarking Technique for Image Authentication. International Journal of Advanced Research in Computer Science and Software Engineering, 3(9), 284-291.

Patel, P., & Patel, Y. (2015). Secure and authentic DCT image steganography through DWT–SVD based Digital watermarking with RSA encryption. International Conference on Communication Systems and Network Technologies (pp. 736-739).

Rojat, A. (2012). Review of cryptanalysis of RSA and its variants by Jason Hinek. ACM SIGACT News, 43(1), 16-18.

Ray, A. K., Padhihary, S., Patra, P. K., & Mohanty, M. N. (2015). Development of a New Algorithm Based on SVD for Image Watermarking. Advances in Intelligent Systems and Computing (pp. 79-87).

Sujatha, S.S., & Sathik, M. Mohamed. (2010). Feature Based Watermarking Algorithm by Adopting Arnold Transform. International Conference on Advances in Information and Communication Technologies (pp. 78-82).

Saha, B. J., Pradhan, C., Kabi, K.K., & Bisoi, A. K. (2014). Robust Watermarking Technique using Arnold's Transformation and RSA in Discrete Wavelets. International Conference on Information Systems and Computer Networks (pp. 83-87).

Saikrishna N, & Resmipriya M G. (2016). An Invisible Logo Watermarking using Arnold Transform. Procedia Computer Science, 93, 808 – 815.

Sikder, I., Dhar, P. K., & Shimamura, T. (2017). A Semi-Fragile Watermarking Method Using Slant Transform and LU Decomposition for Image Authentication. International Conference on Electrical, Computer and Communication Engineering (ECCE) (PP, 881-885).

Thakkar, F. N., & Srivastava, V. K. (2017). A blind medical image watermarking: DWT-SVD based robust and secure approach for telemedicine applications. Multimed Tools Appl, 76(3), 3669–3697.

Verma, V. S., & Jha, R. K. (2015). An Overview of Robust Digital Image Watermarking. IETE Technical Review, 32(6), 479-496.

Wang, Q., Li, J., & Kang, B. (2015). Digital Watermarking Algorithm for QR Code Images Based on DCT and Blocked Images. Journal of Information & Computational Science, 12(11), 4153-4159.

YAVUZ, E., & TELATAR, Z. (2007). Improved SVD-DWT Based Digital Image Watermarking Against Watermark Ambiguity. ACM Symposium on Applied Computing (pp. 1051-1055).

Mr. Yang Liu (First author) is a PhD research student majoring in Data Communications Security, supervised by Professor Shanyu Tang who is Chair Professor of Information Security in the School of Computing and Engineering at the University of West London, St Mary's Road, Ealing, London W5 5RF, UK (E-mail: Shanyu.Tang@uwl.ac.uk; Tel: +44 (0)20 8231 2948; Fax: +44 (0)20 8231 2402).