

# Reversible Watermarking with Adaptive Embedding Threshold Matrix

Guangyong Gao<sup>1,2</sup>, Yun-Qing Shi<sup>2</sup>, Xingming Sun<sup>3,\*</sup>, Caixue Zhou<sup>1</sup>, Zongmin Cui<sup>1</sup> and Liya Xu<sup>1</sup>

<sup>1</sup> School of Information Science & Technology, Jiujiang University, Jiujiang, 332005, China

<sup>2</sup> Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, 07102, USA

<sup>3</sup> School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China

[e-mail: sunnudt@163.com]

\*Corresponding author: Xingming Sun

*Received April 23, 2016; revised July 5, 2016; accepted July 22, 2016;  
published September 30, 2016*

---

## Abstract

In this paper, a new reversible watermarking algorithm with adaptive embedding threshold matrix is proposed. Firstly, to avoid the overflow and underflow, two flexible thresholds, TL and TR, are applied to preprocess the image histogram with least histogram shift cost. Secondly, for achieving an optimal or near optimal tradeoff between the embedding capacity and imperceptibility, the embedding threshold matrix, composed of the embedding thresholds of all blocks, is determined adaptively by the combination between the composite chaos and the average energy of Integer Wavelet Transform (IWT) block. As a non-linear system with good randomness, the composite chaos is suitable to search the optimal embedding thresholds. Meanwhile, the average energy of IWT block is calculated to adjust the block embedding capacity, and more data are embedded into those IWT blocks with larger average energy. The experimental results demonstrate that compared with the state-of-the-art reversible watermarking schemes, the proposed scheme has better performance for the tradeoff between the embedding capacity and imperceptibility.

---

**Keywords:** Reversible watermarking, embedding threshold matrix, composite chaos, average block energy

---

This work is supported in part by the National Natural Science Foundation of China (Grant Nos. 61362032, 61662039, 61364025, 61462048), the Natural Science Foundation of Jiangxi Province, China (Grant Nos. 20151BAB207003, 20151BAB217015, 20161BAB202036), the State Scholarship for Overseas Studies (Grant No. 201408360019), PAPD and CICAET.

## 1. Introduction

As we all know, the digital watermarking technology has been widely applied to the copyright protection of digital resource [1-3]. Different from the irreversibility of traditional watermarking schemes [4-7], the reversible watermarking can restore the original host after extracting the embedded watermarking information [8,9], thus it is a good choice to apply the reversible watermarking to the authentication of digital works, especially to military and medical digital resources.

In recent years, the reversible watermarking technology acquires a fast development, and the existing proposed schemes include the spatial domain-based [10-16] and frequency domain-based algorithms [17-22]. The reversible watermarking algorithms based on spatial domain can be divided into three categories, namely, histogram shifting-based method [10-12], difference expansion-based method [13,14], and prediction error-based method [15,16]. The reversible watermarking algorithms based on frequency domain mainly focus on the Discrete Cosine Transform (DCT) domain and wavelet transform domain.

The histogram shifting-based reversible watermarking method was firstly proposed by Ni et al. [10]. In this method, the watermarking bits were embedded by the shifting of zero-peak pairs of the image histogram. The proposed method needs the encoder to transfer the extra side-information to the decoder using other channel, hence it is non-blind. In [13], Tian firstly presented the difference expansion-based method, in which the difference and average values of neighbor pixels are applied to hide watermarking information. The prediction error-based method was developed by Thodi et al. [15], in which a pixel's three-neighbor context is utilized to predict the pixel value, and the expansion of prediction-error between the original pixel value and the estimated one is used to embed message. The prediction error-based algorithm achieved a maximal embedding rate of 1 bit per pixel (bpp) during one-round embedding.

Besides the above-mentioned algorithms performed in the spatial domain, some reversible watermarking algorithms based on frequency domain are also presented. Lin proposed a reversible watermarking scheme based on the varieties of DCT coefficients of an image. The cover images are decomposed into several different frequencies, and the high-frequency parts are embedded with secret data [17]. In the Haar wavelet transform domain, Chang et al. proposed a reversible watermarking scheme, which encodes the wavelet coefficients by an adaptive arithmetic coding method and embeds the secret message in it [18]. This scheme provides high embedding capacity and good image quality. In [19], a histogram modification based reversible data hiding scheme in the integer Discrete Wavelet Transform (integer DWT) domain is presented on the compressed images based on Block Truncation Coding (BTC). Additional data is embedded into the middle and high frequency sub-bands of the constructed image after integer DWT. In [20], Xuan et al. adopted the companding technology on Integer Wavelet Transform (IWT) coefficients to develop a reversible watermarking algorithm. The companding technology is conducted by applying the compression and expansion function to lower the distortion introduced by watermarking embedding. On the base of the method proposed by Xuan et al., Arsalan et al. [21] presented an intelligent reversible watermarking approach denoted by *GA-RevWM* on IWT domain for medical images using the block-based embedding strategy. In this method, the Genetic Algorithm (GA) is utilized to select intelligently the embedding threshold for enhancing the imperceptibility of the watermarked image and improving the performance of the algorithm. In [22], a Distortion-Oriented,

Minimized (DOM) embedding algorithm is proposed. By using a cascading trellis code, the overall modifications to the host coefficients are minimal, and some specified host coefficients are kept intact. To increase the payload, both the scaling coefficients and the wavelet coefficients are involved in computation.

The existing reversible watermarking schemes provide the effective tradeoff between the embedding capacity and imperceptibility. Due to that the tradeoff depends largely on the reasonable selection of the embedding threshold, thus the improvement room for the algorithm performance still exists. In this paper, a new intelligent reversible watermarking algorithm denoted by *CE-RevWM* is proposed. The combination between the composite chaos and the average energy of IWT block is used to adaptively determine the embedding thresholds, by which a better tradeoff between the embedding capacity and imperceptibility can be achieved. The experimental results demonstrate that compared with the existing reversible watermarking schemes, the proposed scheme has better performance.

The outline of the paper is organized as follows. The preliminary knowledge is presented in Section 2. Section 3 describes in detail the proposed intelligent reversible watermarking algorithm. Experimental results and analyses are shown in Section 4. Finally, the conclusion is drawn in Section 5.

## 2. Preliminary knowledge

### 2.1 Composite chaos

As a non-linear system, chaos system is applied widely in many fields. For example, as the learning tool, the performance of Neural Network (NN) [23, 24] and Support Vector Machine (SVM) [25-27] is not satisfactory, but we can use chaos system to further optimize and improve their learning ability. Moreover, compared with common chaos, composite chaos has stronger randomness and larger complexity [3], so that it is a good choice to use composite chaos to optimize the search for the adaptive embedding thresholds, which can improve the performance of reversible watermarking algorithm. Next, the definition and performance analysis of a composite chaos is presented below [3].

#### 2.1.1 Definition of composite chaos

**Definition 1** Assume  $x_{n+1} = G(x_n) = g_0^{(U_n)}(x_n)$ ,  $U_n = S(\lfloor rg_1(y_n) \rfloor \bmod t) \oplus S(\lfloor pg_1(y_n) \rfloor \bmod q)$ ,  $g_0$  and  $g_1$  are discrete chaos systems, and  $x_n, y_n$  are the  $n$ th elements of  $g_0, g_1$ , respectively.  $U_n$  denotes the interval factors used to sample the chaos sequence of  $g_0$ .  $S(\cdot)$  is the function taking binary bits.  $\oplus$  expresses XOR operation,  $r, t, p$  and  $q$  are the integer constants, we call  $x_{n+1} = G(x_n) = g_0^{(U_n)}(x_n)$  composite system defined on  $g_0$  and  $g_1$ .

Logistic map (Eq. (1)) is used to achieve the original chaotic sequence, and a Hybrid Optical Bistable Equation (HOBE, Eq. (2)) is adopted to obtain the random interval factors.

$$x_{n+1} = g_0(x_n) = 4x_n(1 - x_n) \quad (1)$$

$$y_{n+1} = g_1(y_n) = 4 \sin^2(y_n - 2.5) \quad (2)$$

Then by (3), a composite system is generated with the initial values  $(x_0, y_0)$  as (0.5217, 0.3148).

$$\begin{cases} x_{n+1} = G(x_n) = g_0^{(U_n)}(x_n) \\ U_n = S(\lfloor 157y_n \rfloor \bmod 6) \oplus S(\lfloor 213y_n \rfloor \bmod 8) \end{cases} \quad (3)$$

For 1-dimension mapping, the Lyapunov exponent [28] of  $G(x)$  is defined by

$$\xi = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln \left| \frac{dG(x)}{dx} \right|_{x=x_i} \quad (4)$$

where  $\xi$  is the calculated Lyapunov exponent value. After the calculation using Eq. (4), we get  $\xi = 1.5909$  for the Lyapunov exponent of the proposed composite system (Eq. (3)), and a positive  $\xi$  denotes that the composite system is a discrete chaotic system.

### 2.1.2 Performance analysis

The Lyapunov exponent represents the average apart index introduced by each iteration of chaotic system, and reflects the sensitivity to slight change of initial values and randomness of movement tracks, hence, it is suitable to embody the randomness of a chaotic sequence. The approximate entropy is presented by Pincus [29] to solve the problem of getting entropy in chaos phenomenon, and is a nonnegative quantitative description about nonlinear time series complexity. The bigger the value of approximate entropy, the more complex the nonlinear physical process. The randomness and complexity of a chaotic sequence are two important indexes evaluating the algorithm security.

**Table 1** shows the comparative results for the Lyapunov exponent and approximate entropy among four single-chaotic maps and the proposed composite chaos, which demonstrates that the composite chaos has stronger randomness and bigger complexity than four single-chaotic maps. In addition, the discretized binary form  $\{\beta_{n,K}\}_{n=1}^{\infty}$  of the composite chaotic sequence  $\{x_n\}_{n=1}^{\infty}$  is generated by

$$B_K(x) = \begin{cases} 0, & x \in \bigcup_{j=0}^{2^{K-1}-1} T_{2^j}^K \\ 1, & x \in \bigcup_{j=0}^{2^{K-1}-1} T_{2^{j+1}}^K \end{cases} \quad (5)$$

where  $K$  is a positive integer [30]. Let  $x_n \in [I, J]$ ,  $T_0^K, T_1^K, \dots, T_{2^{K-1}-1}^K$  mean  $2^K$  continuous equant subintervals of  $[I, J]$ , then  $\{\beta_{n,K}\}_{n=1}^{\infty} = \{B_K(x_n)\}_{n=1}^{\infty}$ . From [3], it is known that the  $\{\beta_{n,K}\}_{n=1}^{\infty}$  with  $K=1$  passes 16 different statistical tests, i.e., SP 800-22 standard published by NIST [31], which further indicates the composite chaos has a good randomness.

**Table 1.** Comparison of randomness and complexity

| Chaos           | Lyapunov exponent | Approximate entropy |
|-----------------|-------------------|---------------------|
| HOBE            | 0.6474            | 0.6309              |
| Logistic        | 0.6914            | 0.6430              |
| Lorenz          | 0.0318            | 0.2140              |
| Tent            | 0.8434            | 0.6127              |
| Composite chaos | 1.5909            | 1.6909              |

## 2.2 Integer wavelet transform

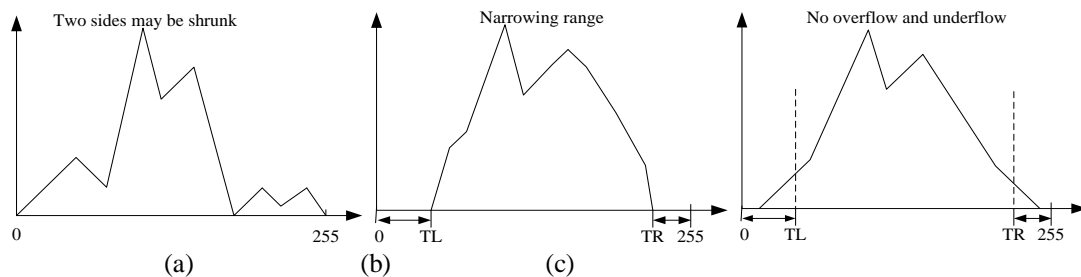
The wavelet transform has been widely used in digital watermarking because the wavelet coefficients have the features of decorrelation and are consistent with Human Visual System (HVS). The IWT mapping integer to integer [32] can reconstruct the original signal without any distortion, thus it is suitable to apply IWT to reversible watermarking. In the domain of IWT, the approximation sub-band is denoted by  $LL$ , and the horizontal sub-band, diagonal sub-band and vertical sub-band, consisting of the detail sub-bands, are denoted by  $LH$ ,  $HH$  and  $HL$ , respectively. The CDF(2,2) is a wavelet family. It is indicated in [20] that CDF(2,2) is better than other wavelet families in terms of high embedding capacity and visual quality of marked images. In addition, CDF(2,2) has also been used by JPEG2000 standard. So CDF(2,2) is adopted for IWT in proposed scheme due to its merits.

## 3. Proposed scheme

In this Section, an intelligent reversible watermarking algorithm denoted by *CE-RevWM* is presented. Firstly, the preprocessing procedure is applied to avoid the overflow and underflow issues. Then the data embedding on the detailed sub-bands of IWT is developed, and the embedding thresholds are ascertained in terms of the combination between the composite chaos and the average energy of IWT block.

### 3.1 Preprocessing against overflow and underflow

For avoiding the overflow and underflow issues brought by the watermark embedding process, a preprocessing procedure using histogram modification needs to be taken. The histogram modification mainly changes the pixel value close to upper boundary and lower boundary (e.g. 255 and 0 for an 8-bit gray-level image). Currently, the most of existing algorithms adopt a constant threshold to change the histograms located at an image's two sides, which may lead to excessive modification increasing the image distortion for a part of images, and also may lead to insufficient modification causing the overflow and underflow for another part of images. It is due to that every image has a different histogram distribution. In this paper, two flexible thresholds,  $TL$  and  $TR$ , are applied to preprocess the image histogram with least histogram shift cost, i.e. the pixel values from 0 to  $TL-1$  are increased by  $TL$ , and those from  $256-TR$  to 255 are subtracted by  $TR$ , which can refer to Fig. 1 [33]. The position information of modified pixels in the preprocessing operation is saved in a location map, which is compressed by Arithmetic Encoding (AE) to decrease its size. The initial values of  $TL$ ,  $TR$  are all set as 0, then  $TL$ ,  $TR$  are adjusted by adding 1 according to the histogram of the watermarked image, i.e., if  $x < 0$ , then  $TL = TL + 1$ , and if  $x > 255$ , then  $TR = TR + 1$ , where  $x$  is a pixel of the watermarked image. The adjustment for  $TL$ ,  $TR$  stops until the suitable  $TL$  and  $TR$  are achieved, with which  $x$  satisfies  $0 \leq x \leq 255$ .



**Fig. 1.** Histogram modification with  $TL$ ,  $TR$  for preprocessing. (a) Original histogram, (b) adjusted histogram, (c) histogram after data embedding

### 3.2 Data embedding on detailed sub-bands of IWT

Firstly, the cover image with the size of  $M \times N$  is divided into some non-overlapped blocks with the size of  $P \times P$ , then IWT is applied on every block. The watermarking data are embedded into the detailed sub-bands of every block with the companding technique consisting of a successive process of compression and expansion [20].

For a bit  $w$ , we embed it by  $z_w = 2z + w$ , where  $z$  denotes the coefficient of IWT detail sub-bands and  $z_w$  is the extended coefficient after data embedding. Since the change of  $z$  with a big value is large after  $w$  is embedded into  $z$ , which may degrade severely the image imperceptibility and cause easily the situation of overflow and underflow, thus a compression operation on  $z$  should be made to decrease the change. The compression function  $f_c$  is shown by

$$z_c = f_c(z) = \begin{cases} z, & |z| < T(p, q) \\ \lambda(z) \cdot \left( \left\lfloor \frac{|z| - T(p, q)}{2} \right\rfloor + T(p, q) \right), & |z| \geq T(p, q) \end{cases} \quad (6)$$

where  $z$ ,  $z_c$  are the original and compressed coefficients of IWT detail sub-bands, respectively.  $T(p, q)$  is the embedding threshold for a specific block  $(p, q)$  in which the detail sub-bands coefficient  $z$  resides,  $p = 1, 2, \dots, \lfloor M/P \rfloor$ ,  $q = 1, 2, \dots, \lfloor N/P \rfloor$ , where  $\lfloor k \rfloor$  means taking the bottom integral of  $k$ .  $T(p, q)$  is a positive integer and  $T(p, q) \in [1, 15]$ . In terms of Eq. (6) and the embedding equation of a bit  $w$ ,  $z_w = 2z + w$ , if the value of  $T$  is too big, then a number of IWT coefficients  $z$  will not be compressed, which will be adverse to the imperceptibility of the embedded image. Thus, the upper limitation of  $T$  can not be too big. By the repeated experiments, we adopt an appropriate value 15 as the upper limitation of  $T$ . The mapping matrix for  $T(p, q)$  is shown in **Table 2**. For the different block  $(p, q)$ ,  $T(p, q)$  may be also different, which value is determined by the method proposed in Section 3.4. In Eq. (6), if  $z \geq T(p, q)$ ,  $\lambda(z) = 1$ , otherwise,  $\lambda(z) = -1$ . The embedded version of  $z_c$  is denoted by  $z_{cw} = 2z_c + w$ .

**Table 2.** Mapping matrix for the block embedding threshold  $T(p, q)$

|                             |                             |                                 |   |
|-----------------------------|-----------------------------|---------------------------------|---|
| $T(1, 1)$                   | $T(1, 2)$                   | $T(1, \dots)$                   | $T(1, \lfloor N/P \rfloor)$                   |
| $T(2, 1)$                   | $T(2, 2)$                   | $T(2, \dots)$                   | $T(2, \lfloor N/P \rfloor)$                   |
| $T(\dots, 1)$               | $T(\dots, 2)$               | $T(\dots, \dots)$               | $T(\dots, \lfloor N/P \rfloor)$               |
| $T(\lfloor M/P \rfloor, 1)$ | $T(\lfloor M/P \rfloor, 2)$ | $T(\lfloor M/P \rfloor, \dots)$ | $T(\lfloor M/P \rfloor, \lfloor N/P \rfloor)$ |

The expanding function  $f_e$  corresponding to  $f_c$  is given by

$$z' = f_e(z_c) = \begin{cases} z_c, & |z_c| < T(p, q) \\ \lambda(z_c) \cdot (2|z_c| - T(p, q)), & |z_c| \geq T(p, q) \end{cases} \quad (7)$$

where  $z'$  denotes the expanded version of  $z_c$ . For the coefficient  $z$  satisfying  $|z| \geq T(p, q)$ , the companding error  $e$  between  $z$  and  $z'$  is given by

$$e = z - f_e(f_c(z)) \quad (8)$$

It is observed that  $e \neq 0$  for some coefficients  $z$ . Moreover, for  $z \geq T(p, q)$ , we can achieve  $e \in \{0, 1\}$  and for  $z \leq -T(p, q)$ ,  $e \in \{-1, 0\}$ .

To restore the original coefficient  $z$  at the receiving side, the values of  $TL$ ,  $TR$ , the compressed location map, the companding error  $e$ , and the threshold matrix  $T$ , called overhead data, should be embedded into the cover image along with the pure payload (namely the watermarking data). It is noted for the embedding of  $e$  that the bit '1' is embedded when  $e = -1$ . The original coefficient  $x$  can be restored by Eq. (9).

$$z = \begin{cases} z' + 1 & \text{if } z' > 0 \text{ and } e = 1 \\ z' & \text{if } e = 0 \\ z' - 1 & \text{if } z' < 0 \text{ and } e = 1 \end{cases} \quad (9)$$

### 3.3 Selection of embedding thresholds

The higher the complexity of block texture is, the less the influence of visual perception on the block brought by embedding data is. Thus we can embed more data in these blocks of higher texture complexity. In other words, the assigned embedding threshold for the block with larger texture complexity is larger than that for the block with less one. Also, the larger the detailed sub-band coefficients of an IWT block are, the higher the texture complexity of this block is. Here, the average energy of detail sub-band coefficients of an IWT block is used as the measurement of texture complexity, which can be computed as shown below.

$$E(p, q) = \frac{\sum_{i=1}^N (c_{LHi} + c_{HHi} + c_{HLi})}{3 \times N} \quad (10)$$

where  $E(p, q)$  denotes the average energy of a block,  $c_{LHi}$ ,  $c_{HHi}$ ,  $c_{HLi}$  are the coefficients of  $LH$ ,  $HH$ ,  $HL$  sub-bands of a block, respectively, and  $N$  is the amount of every detailed sub-band coefficients.

In the embedding procedure, the block embedding threshold  $T(p, q)$  is randomly selected by a chaos sequence according to the value of  $E(p, q)$ . The three thresholds for the average energy of image block are denoted by  $E_1$ ,  $E_2$  and  $E_3$ , respectively. The corresponding embedding thresholds are  $T_1$ ,  $T_2$  and  $T_3$ , respectively. Note the discretized binary sequence of the composite chaos presented in Section 2 as  $S$ . All subsequences of  $S$  are composed of sequential four elements of  $S$ , which constitute a new sequence marked as  $R$ . Then  $R$  is divided into four parts  $R_1$ ,  $R_2$ ,  $R_3$  and  $R_4$ , where  $D(R_{1i}) \in (1, T_1)$ ,  $D(R_{2i}) \in (T_1 + 1, T_2)$ ,  $D(R_{3i}) \in (T_2 + 1, T_3)$  and  $D(R_{4i}) \in (T_3 + 1, 15)$ .  $D(x)$  denotes taking the decimal digit value of  $x$ .  $R_{1i}$ ,  $R_{2i}$ ,  $R_{3i}$  and  $R_{4i}$



mean the  $i$ th element of  $R_1$ ,  $R_2$ ,  $R_3$  and  $R_4$ , respectively. For example, if  $S=\{01101100011\}$ ,  $T_1=3$ ,  $T_2=7$  and  $T_3=11$ , then  $R=\{‘0110’, ‘1101’, ‘1011’, ‘0110’, ‘1100’, ‘1000’, ‘0001’, ‘0011’\}$ ,  $R_1=\{‘0001’, ‘0011’\}$ ,  $R_2=\{‘0110’, ‘0110’\}$ ,  $R_3=\{‘1011’, ‘1000’\}$  and  $R_4=\{‘1101’, ‘1100’\}$ .

The block embedding threshold  $T(p, q)$  is achieved by

$$T(p, q) = \begin{cases} D(R_{1i}) & E(p, q) \leq E_1 \\ D(R_{2i}) & E_1 < E(p, q) \leq E_2 \\ D(R_{3i}) & E_2 < E(p, q) \leq E_3 \\ D(R_{4i}) & E_3 < E(p, q) \end{cases} \quad (11)$$

Below, the process to select  $T(p, q)$  is illustrated by a simple example. Let  $E_1$ ,  $E_2$  and  $E_3$  be 6, 11, 16, respectively. If the average energy of current block  $E(p, q) = 9$ , then  $T(p, q)$  is selected as  $D(R_{2i})$  from  $R_2$ , where  $R_{2i}$  denotes the current element of  $R_2$ , then  $i=i+1$ .

The energy thresholds  $E_1$ ,  $E_2$  and  $E_3$  can be used to control the embedding capacity and the Peak-Signal-to-Noise-Ratio (PSNR) of the marked image. The larger  $E_1$ ,  $E_2$  and  $E_3$  are, the smaller  $T(p, q)$  is, leading to the decrease of the embedding capacity as well as the increase of PSNR of the marked image. Conversely, the smaller  $E_1$ ,  $E_2$  and  $E_3$  are, the larger  $T(p, q)$  is, thus leading to the increase of data embedding capacity as well as the decrease of the marked image's PSNR. In addition, a concrete example for  $T(p, q)$  may refer to the subsection 4.4.

### 3.4 Procedure of proposed algorithm

#### 3.4.1 Watermarking embedding

The procedure of watermarking embedding is illustrated in Fig. 2. The concrete steps are summarized as follows:

Step1. Generate the composite chaos sequence  $S$  according to the method mentioned in Section 2, then  $S$  is divided into four parts  $R_1$ ,  $R_2$ ,  $R_3$  and  $R_4$  in terms of the embedding thresholds  $T_1$ ,  $T_2$  and  $T_3$ .

Step 2. Divide the image into some non-overlapped blocks with the size of  $P \times P$ , then compute the average energy  $E(p, q)$  of every block by Eq. (10).

Step 3. Using  $E(p, q)$ ,  $E_1$ ,  $E_2$  and  $E_3$ , the embedding threshold  $T(p, q)$  is allocated by Eq. (11). We search the adaptive  $T(p, q)$  for every block by the  $N$  times looping execution from Step 3 to Step 6, where  $N$  is a predefined threshold. That is the marked image can achieve the largest PSNR with the adaptive  $T(p, q)$ . Here an example is given for allocating  $T(p, q)$  in  $N$  times loops for an appointed block  $(p, q)$ . If  $E(p, q) \leq E_1$ , then  $T(p, q) = D(R_{11})$  in the first loop. In next loop,  $T(p, q) = D(R_{12})$ . Similarly, in  $N$ th loop,  $T(p, q) = D(R_{1N})$ .

Step 4. The preprocessing against overflow and underflow is made in terms of the method proposed in Section 3.1. Firstly, the initial values of  $TL$  and  $TR$  are all set as 0, then the pure payload and overhead data are embedded into the detailed sub-bands of every block using the companding technique on IWT domain according to the strategy presented in Section 3.3. If the marked image has an underflow, then  $TL=TL+1$ . Also, if an overflow happens, then  $TR=TR+1$ . Repeat this procedure until the marked image avoids overflow and underflow.

Step 5. Compute the PSNR of the marked image and save the best PSNR denoted by  $P^*$  and the corresponding optimal threshold matrix  $T^*$  consisting of  $T(p, q)$  of every block.

Step 6. If  $L < N$ , where  $L$  denotes the number of loops, then go to Step 3. Else, go to Step 7.

Step 7. If  $EP=0$  and  $E_1 > E_{\min}$ , then the energy thresholds  $E_1$ ,  $E_2$  and  $E_3$  are decreased by



subtracting 1 to enhance the embedding capacity of pure payload and go to Step 3, where  $EP=0$  denotes the given pure payload cannot be embedded, and  $E_{\min}$  is the lower limit. Else, if  $EP=1$  and  $E_3 < E_{\max}$ , then the energy thresholds  $E_1$ ,  $E_2$  and  $E_3$  are increased by adding 1 to enhance the PSNR and go to Step 3, where  $EP=1$  denotes the given pure payload can be embedded, and  $E_{\max}$  is the upper limit. Else, the optimal block embedding threshold matrix  $T^*$  is utilized to embed the pure payload and overhead data, then go to Step 8.

Step 8. The inverse integer wavelet transform is made on every image block to achieve the final marked image  $H'$ .

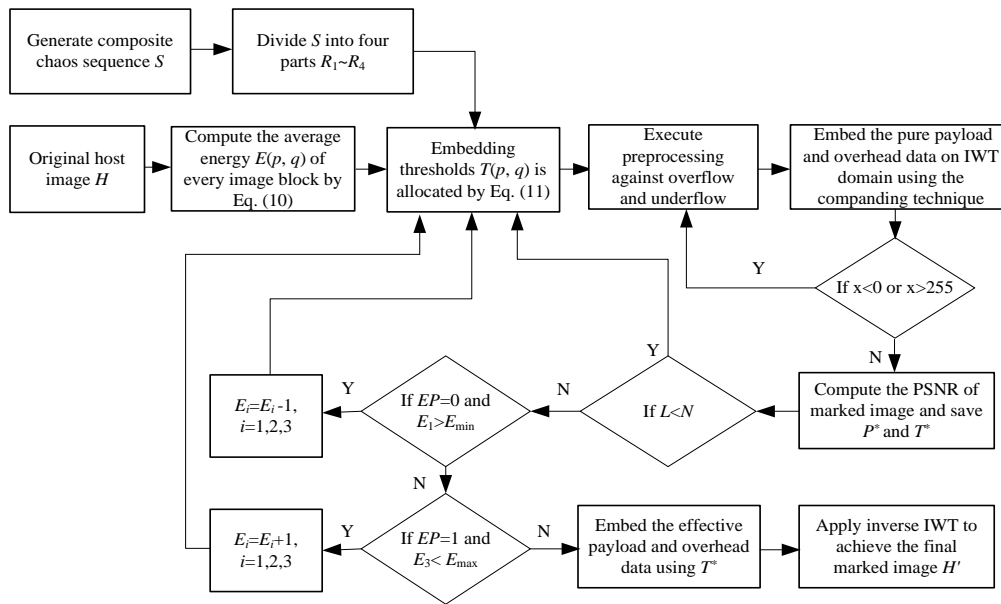


Fig. 2. The diagram for the procedure of watermarking embedding

### 3.4.2 Watermarking extracting

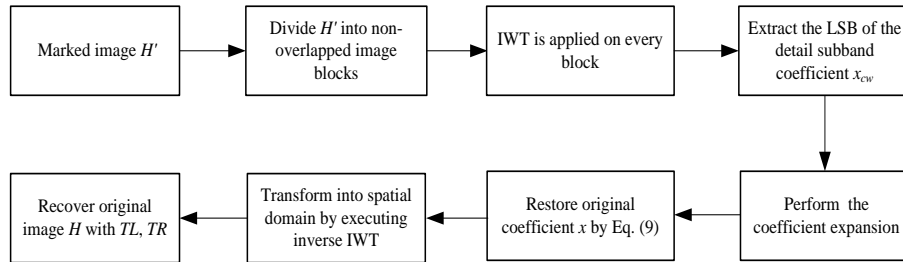
On the receiving side, the watermarking extracting procedure is simpler compared with the watermarking embedding procedure. The diagram for the extracting watermarking is presented in Fig. 3. The concrete steps are summarized as follows.

Step 1. The received marked image  $H'$  is divided into some non-overlapped blocks with the size of  $P \times P$ .

Step 2. The IWT is applied to every image block, then the *LSB* of the detail sub-band coefficient  $x_{cw}$  of every block is extracted, which means  $w = LSB(x_{cw})$  and  $x_c = (x_{cw} - w)/2$ . From the extracted *LSBs*, the pure payload and the overhead data involving the preprocessing thresholds  $TL$ ,  $TR$ , the compressed location map, the companding error  $e$  and the threshold matrix  $T$  can be separated.

Step 3. The expansion is performed using the threshold matrix  $T$  and  $x_c$  by Eq. (7) to obtain  $x'$ . After the expansion, the companding error  $e$  is used to restore the original coefficient  $x$  by applying Eq. (9).

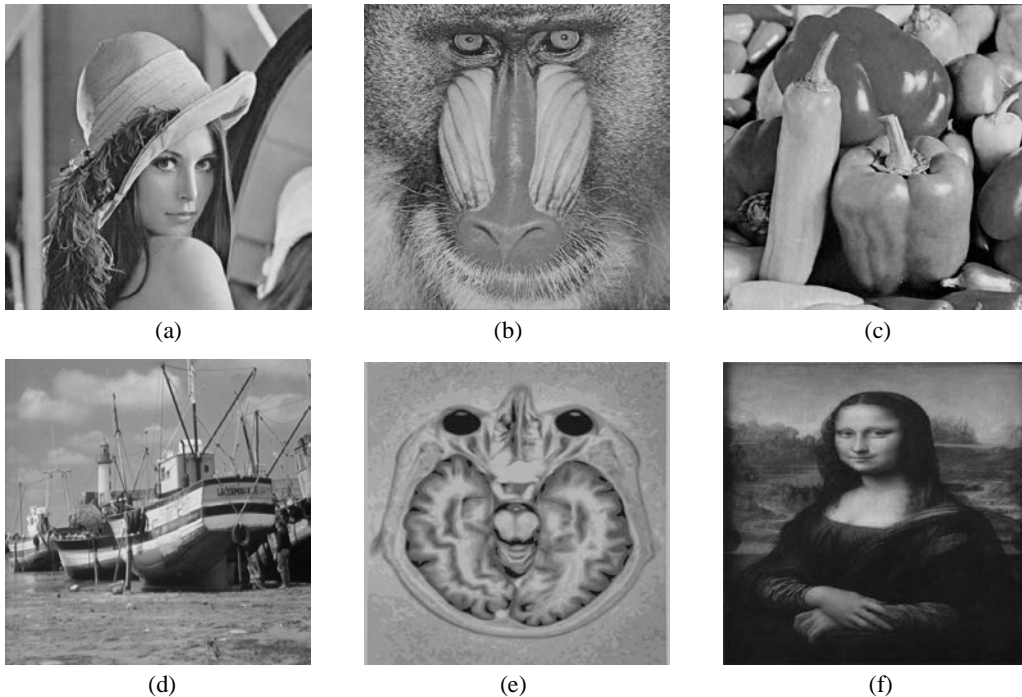
Step 4. The image is transformed into the spatial domain by executing the inverse IWT. Then with decompressed location map, those pixels changed in preprocessing phase are labeled. If the identified pixel value is less than 128, it is subtracted by  $TL$ , or is increased by  $TR$  otherwise. To comply with this rule, the maximum value of  $TL$ ,  $TR$  is set as 64 for avoiding ambiguity. Thus, the original image  $H$  is recovered.



**Fig. 3.** The diagram for the procedure of extracting watermarking

#### 4. Experimental results and analysis

In this section, the experimental results are presented with the experimental environment of MATLAB R2012b under windows 8. For measuring the effect of preprocessing scheme, the different kinds of grayscale images are selected as test images with the size of  $512 \times 512$ . These images shown in **Fig. 4** include four standard images (Lena, Baboon, Pepper and Boat), one medical image, one artistic image, one antique image and one calligraphy image, which are with different histogram distributions. The pure payload are the amount of embedded watermarking bit per pixel (bpp), excluding the overhead data.





**Fig. 4.** The Different kinds of grayscale images used for the experiments.

(a) Lena image, (b) Baboon image, (c) Pepper image, (d) Boat image, (e) Medical image,  
(f) Artistic image, (g) Antique image, (h) Calligraphy image.

#### 4.1 Pure embedding payload

Since the data are embedded into three detailed sub-bands of IWT, therefore, the maximum embedding payload that can be obtained is 0.75 bpp. Moreover, the overhead data need to be embedded so as to restore original image in reversible way at receiving side. Thus, the maximum pure payload embedded is generally less than 0.7 bpp. Through multiple runs of watermarking embedding process, we can embed more pure payload than 0.7 bpp. **Table 3** shows the pure payloads, the sizes of overhead data and the corresponding PSNRs with two-round runs of embedding process for Lena image. For the convenience of presentation and comparison with other algorithms, the experiments in later subsections only provide the results for the pure payloads from 0.1 to 0.6 bpp with one-round run of embedding process.

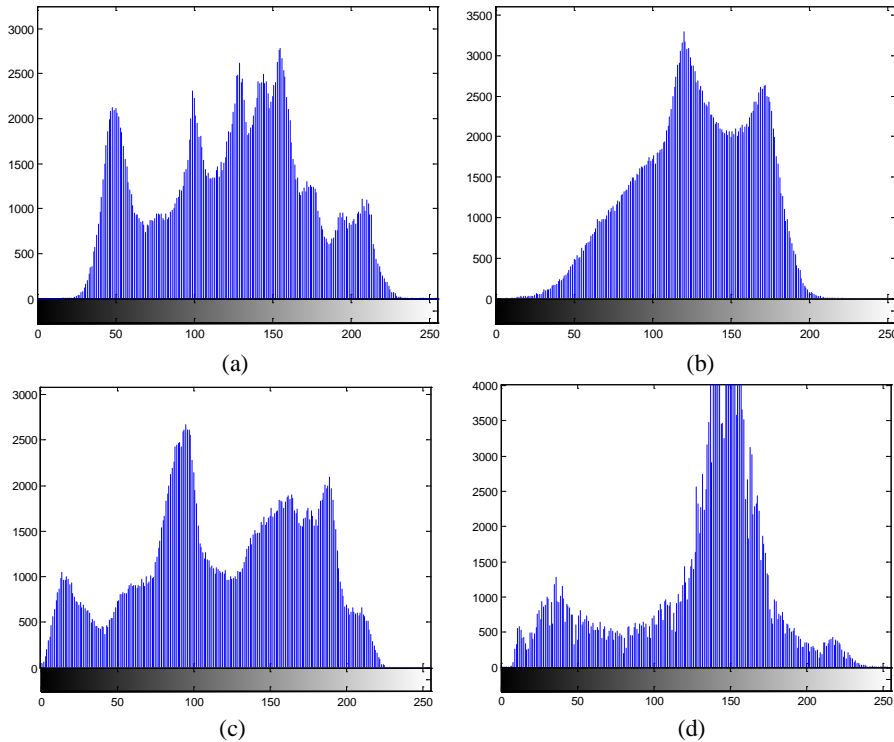
From **Table 3**, it is observed that the size of overhead data along with the embedding of the pure payload of 1.1 bpp is less than that of overhead data along with the embedding of the pure payload of 1.0 bpp due to the limitation of total embedding capacity. In fact, for an image, the embedding capacity, including the pure payload and the overhead data, is definite. When the total size of pure payload and overhead data is beyond the embedding capacity provided by an image, we can decrease the energy thresholds  $E_1$ ,  $E_2$  and  $E_3$  to enhance the embedding capacity of pure payload, which can refer to the step 7 of watermarking embedding procedure. Meanwhile, the embedding capacity of overhead data is lowered.

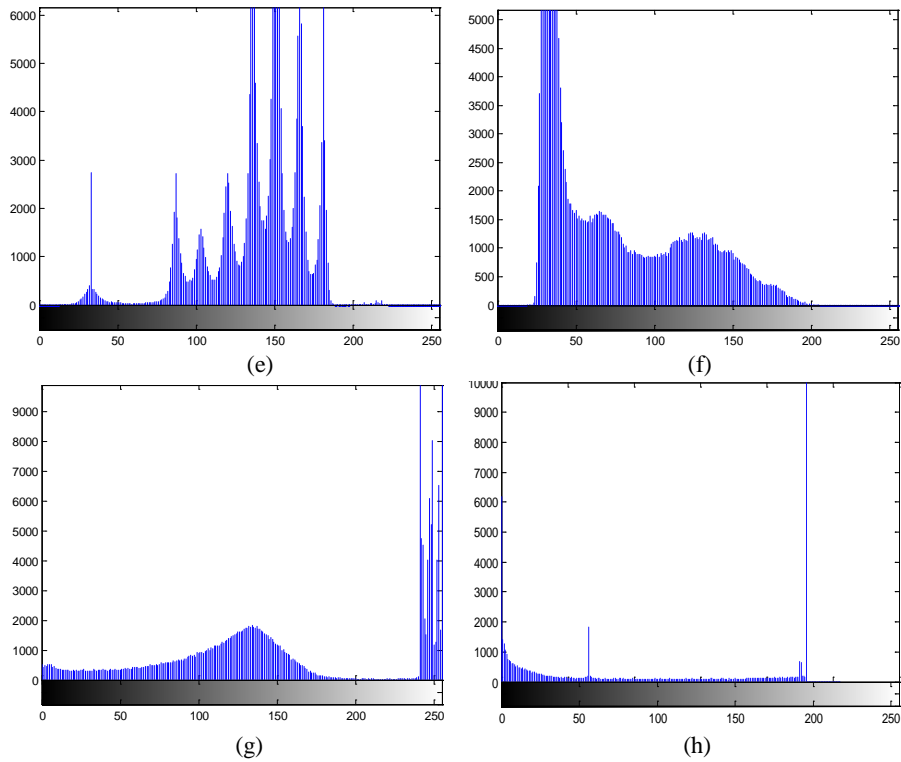
**Table 3.** The experimental results with two-round runs of embedding process for Lena image.

| Pure payload (bpp) | Size of overhead data (bits) | PSNR (dB) |
|--------------------|------------------------------|-----------|
| 0.1                | 23175                        | 49.9      |
| 0.2                | 43728                        | 47.0      |
| 0.3                | 69054                        | 45.5      |
| 0.4                | 82914                        | 45.0      |
| 0.5                | 64110                        | 41.9      |
| 0.6                | 34937                        | 39.9      |
| 0.7                | 88385                        | 38.0      |
| 0.8                | 141366                       | 36.9      |
| 0.9                | 152602                       | 36.2      |
| 1.0                | 111077                       | 33.7      |
| 1.1                | 86306                        | 32.8      |

## 4.2 Preprocessing thresholds with different histogram distributions

The histogram distributions of the different kinds of test images are shown in Fig. 5. It is observed that the histograms of Lena image, Medical image and Artistic image are centered without distributions at two sides, the histograms of Baboon image, Pepper image, and Calligraphy image are focused on the left side and central area without distributions at the right side, and the Boat image and Antique image have the histogram distributions at two sides. The different histogram distributions lead to the different two thresholds used in the preprocessing phase, i.e.  $TL$  and  $TR$ . Table 4 gives the  $TL$ s and  $TR$ s of different kinds of test images under different pure payload (from 0.1 to 0.6 bpp). It is observed from Table 4 that  $TL$ s and  $TR$ s are all 0 for Lena image, Medical image and Artistic image,  $TL$ s are positives and  $TR$ s are 0 for Baboon image, Pepper image and Calligraphy image, and  $TL$ s and  $TR$ s are all positives for Boat image and Antique image. Thus, we can conclude that it is not suitable to apply the constant thresholds to the preprocessing against the overflow and underflow, and the flexible thresholds should be adopted to preprocess the images with different histogram distributions. In Table 4, the notation ‘–’ indicates the embedding is unavailable with corresponding embedding capacity.





**Fig. 5.** Histogram distribution of different kinds of test images.

(a) Lena image, (b) Baboon image, (c) Pepper image, (d) Boat image, (e) Medical image, (f) Artistic image, (g) Antique image, (h) Calligraphy image.

**Table 4.** TLs and TRs of different kinds of test images under different bpp.

(a) Lena image, (b) Baboon image, (c) Pepper image, (d) Boat image, (e) Medical image, (f) Artistic image, (g) Antique image, (h) Calligraphy image.

| bpp | TL | TR |
|-----|----|----|
| 0.1 | 0  | 0  |
| 0.2 | 0  | 0  |
| 0.3 | 0  | 0  |
| 0.4 | 0  | 0  |
| 0.5 | 0  | 0  |
| 0.6 | 0  | 0  |

(a)

| bpp | TL | TR |
|-----|----|----|
| 0.1 | 0  | 0  |
| 0.2 | 0  | 0  |
| 0.3 | 25 | 0  |
| 0.4 | 32 | 0  |
| 0.5 | —  | —  |
| 0.6 | —  | —  |

(b)

| bpp | TL | TR |
|-----|----|----|
| 0.1 | 4  | 0  |
| 0.2 | 3  | 0  |
| 0.3 | 5  | 0  |
| 0.4 | 6  | 0  |
| 0.5 | 12 | 0  |
| 0.6 | —  | —  |

(c)

| bpp | TL | TR |
|-----|----|----|
| 0.1 | 3  | 5  |
| 0.2 | 8  | 13 |
| 0.3 | 9  | 15 |
| 0.4 | 9  | 14 |
| 0.5 | 14 | 26 |
| 0.6 | —  | —  |

(d)

| bpp | TL | TR |
|-----|----|----|
| 0.1 | 0  | 0  |
| 0.2 | 0  | 0  |
| 0.3 | 0  | 0  |
| 0.4 | 0  | 0  |
| 0.5 | 0  | 0  |
| 0.6 | 0  | 0  |

(e)

| bpp | TL | TR |
|-----|----|----|
| 0.1 | 0  | 0  |
| 0.2 | 0  | 0  |
| 0.3 | 0  | 0  |
| 0.4 | 0  | 0  |
| 0.5 | 0  | 0  |
| 0.6 | 0  | 0  |

(f)

| bpp | TL | TR |
|-----|----|----|
| 0.1 | 5  | 6  |
| 0.2 | 7  | 7  |
| 0.3 | 22 | 17 |
| 0.4 | 33 | 30 |
| 0.5 | —  | —  |
| 0.6 | —  | —  |

(g)

| bpp | TL | TR |
|-----|----|----|
| 0.1 | 6  | 0  |
| 0.2 | 6  | 0  |
| 0.3 | 6  | 0  |
| 0.4 | 6  | 0  |
| 0.5 | —  | —  |
| 0.6 | —  | —  |

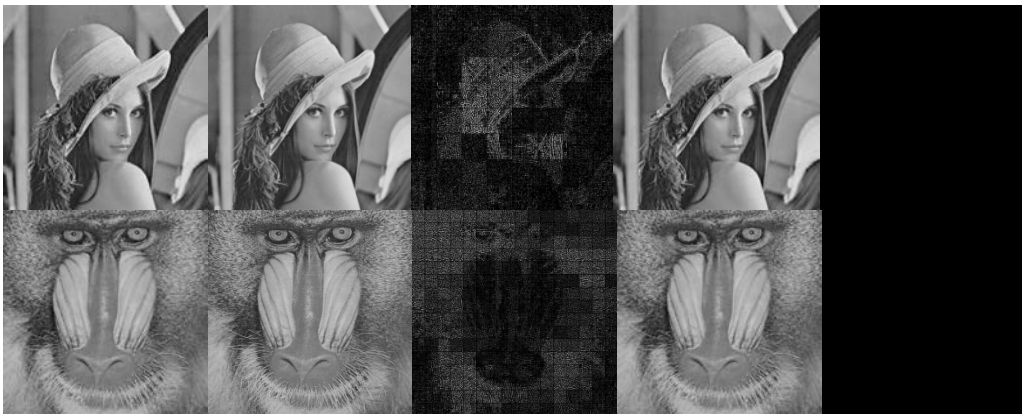
(h)

### 4.3 Experimental results for reversibility

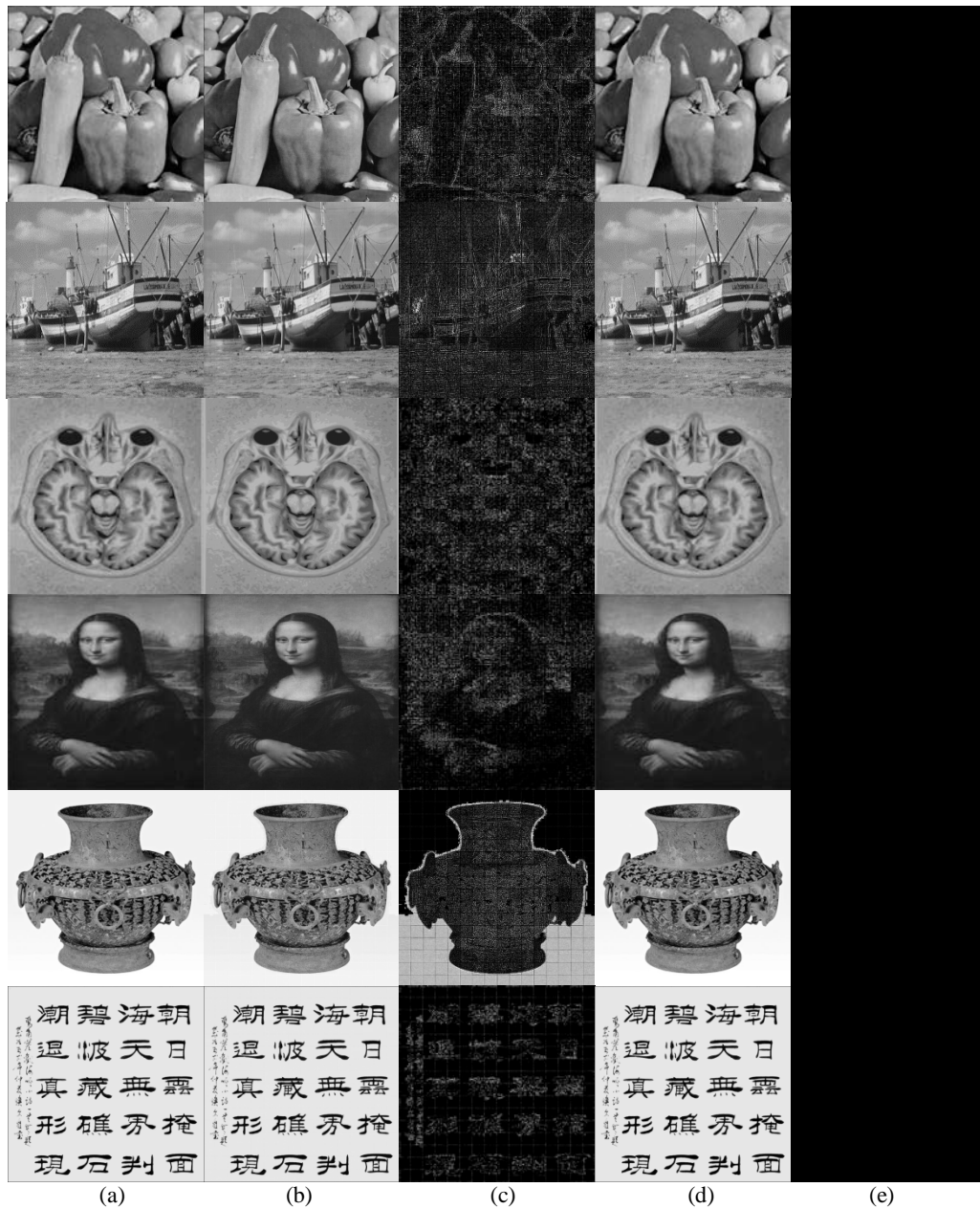
**Fig. 6** shows the reversibility test of proposed scheme for the different kinds of images. The original and marked images are presented in columns (a) and (b) of **Fig. 6**, respectively. The differences between the original and marked images are given in columns (c) of **Fig. 6**. Note that since the differences are not easily perceptible for human eyes, thus an enhancement measure is adopted through Matlab function *adapthisteq* to make the difference visible. The recovered images are shown in columns (d) of **Fig. 6** after the embedded data have been extracted. The differences between the original and recovered images are given in columns (e) of **Fig. 6**. Then we compute the Mean Square Error (MSE) between original and recovered images, and get  $MSE = 0$ , manifesting that proposed scheme is fully reversible. The calculating equation of MSE is shown as follows.

$$MSE = \frac{\sum_{1 \leq i \leq M} \sum_{1 \leq j \leq N} (f_{i,j} - f'_{i,j})^2}{M \times N} \quad (12)$$

where  $f_{i,j}$ ,  $f'_{i,j}$  are the pixel values of original and marked images, respectively.  $M, N$  are the length and width of original and marked images, respectively.







**Fig. 6.** Reversibility test of proposed scheme.

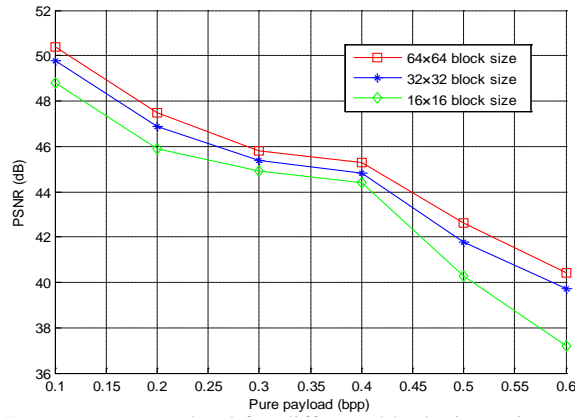
Column (a) lists original images, Column (b) lists watermarked images, Column (c) lists the difference between the original and watermarked images (difference of (a) and (b)), Column (d) lists the restored images, and Column (e) lists the difference between the original and restored images (MSE=0, difference of (a) and (d)).

**4.4 Experimental results with block size**

**Fig. 7** shows the effect of changing block size from 16×16 to 64×64. It is observed that the performance is improved with the increase of block size. Due to that the different embedding thresholds are determined in terms of the average energy of image block, compared with the



block with smaller size, the block with larger size after embedding watermarking generates the block artifact more easily for HVS. With the tradeoff consideration, the block size is set as 32×32, leading to an embedding threshold matrix with the size of 16×16 for a 512×512 test image. Next, an example for the embedding threshold matrix is listed in **Table 5** with the pure payload of 0.6 bpp and 32×32 block size using Lena image.



**Fig. 7.** PSNR vs. pure payload for different block size using Lena image

**Table 5.** Embedding threshold matrix with the pure payload of 0.6 bpp and 32×32 block size using Lena image

|   |   |   |   |   |   |   |    |    |    |    |   |    |   |    |   |
|---|---|---|---|---|---|---|----|----|----|----|---|----|---|----|---|
| 6 | 6 | 7 | 6 | 5 | 7 | 4 | 7  | 7  | 7  | 6  | 7 | 4  | 7 | 6  |   |
| 6 | 6 | 6 | 7 | 5 | 7 | 7 | 5  | 5  | 6  | 7  | 6 | 6  | 6 | 7  | 5 |
| 5 | 5 | 7 | 2 | 1 | 2 | 1 | 3  | 5  | 6  | 6  | 6 | 7  | 6 | 6  | 6 |
| 6 | 4 | 5 | 6 | 4 | 5 | 5 | 4  | 11 | 11 | 7  | 5 | 6  | 7 | 7  | 6 |
| 6 | 5 | 6 | 5 | 7 | 6 | 7 | 4  | 10 | 5  | 9  | 8 | 10 | 4 | 11 | 8 |
| 4 | 6 | 4 | 6 | 6 | 6 | 6 | 4  | 4  | 4  | 11 | 5 | 4  | 8 | 4  | 6 |
| 5 | 6 | 4 | 4 | 5 | 5 | 6 | 5  | 4  | 8  | 5  | 6 | 1  | 4 | 7  | 4 |
| 4 | 4 | 6 | 5 | 5 | 5 | 4 | 11 | 5  | 4  | 4  | 7 | 7  | 6 | 7  | 5 |
| 6 | 6 | 5 | 6 | 4 | 4 | 8 | 7  | 5  | 7  | 6  | 5 | 5  | 7 | 6  | 4 |
| 5 | 5 | 4 | 7 | 4 | 4 | 4 | 5  | 4  | 6  | 7  | 4 | 7  | 6 | 7  | 5 |
| 7 | 6 | 6 | 4 | 4 | 4 | 4 | 6  | 6  | 4  | 4  | 5 | 5  | 4 | 6  | 6 |
| 4 | 7 | 5 | 5 | 4 | 5 | 4 | 6  | 4  | 4  | 4  | 6 | 6  | 6 | 5  | 6 |
| 6 | 7 | 5 | 4 | 6 | 7 | 3 | 5  | 5  | 6  | 7  | 4 | 6  | 4 | 2  | 7 |
| 5 | 7 | 4 | 5 | 7 | 7 | 7 | 4  | 5  | 7  | 5  | 5 | 4  | 4 | 7  | 7 |
| 7 | 4 | 1 | 4 | 4 | 5 | 4 | 4  | 6  | 5  | 6  | 5 | 7  | 7 | 3  | 4 |
| 1 | 2 | 6 | 4 | 4 | 4 | 7 | 5  | 5  | 7  | 5  | 7 | 4  | 4 | 3  | 1 |

#### 4.5 Performance comparison with state-of-the-art approaches

**Fig. 8** shows the experimental comparisons between the proposed method and some state-of-the-art reversible hiding methods performed in the frequency domain. In Lin’s method [17], the secret bits are embedded into the chosen high-frequency coefficients of DCT of each image block. Chang et al.’s [18], Zhang et al.’s [19], Mao et al.’s [22], and Chan et al.’s [34] methods all use only one-level DWT coefficients for data embedding. It is observed from **Fig. 8** that the proposed method and Mao et al.’s method achieve better results than other four methods as a whole. For Mao et al.’s method, by using a cascading trellis coding algorithm, the minimal overall modifications to the host coefficients are achieved to provide high quality to the stego image. But Mao et al. do not conduct the preprocessing against overflow and underflow since they think that only the LSBs of the host detailed sub-band coefficients are modified to embed secret bits so as not to lead to the overflow and underflow problems. It is right for most natural images without histogram distributions at two sides. However, for this type of images with histogram distributions at two sides, for example, the Boat image and

Antique image shown in Fig. 5 (d) and (g), the overflow and underflow may occur. For the proposed method, the improvement is due to the optimization derived from the good searching ability of composite chaos and the decrease of visual distortion caused by the application of the average energy thresholds of image block. By the improvement measures, the adaptive embedding threshold matrix is generated to achieve an effective tradeoff between the embedding capacity and imperceptibility.

The *GA-RevWM* scheme presented by Arsalan et al. [21] selects intelligently the embedding thresholds by applying the genetic algorithm in the IWT domain. By comparison, the proposed *CE-RevWM* scheme pursues the adaptive embedding thresholds by using the composite chaos in combination with the average energy of IWT block. The values of PSNRs and Structure Similarities (SSIMs) against different pure payload values for the two schemes are provided in Table 6 and Table 7, respectively. It is observed that on the whole, the *CE-RevWM* scheme achieves better PSNRs, SSIMs than the *GA-RevWM* scheme for the different kinds of grayscale images. That is mainly because that the searching ability of composite chaos is superior to that of genetic algorithm. In addition, for Antique and Calligraphy images with histogram distributions at two sides or one side, the *GA-RevWM* scheme causes the overflow and underflow, while the proposed *CE-RevWM* scheme avoids it successfully due to the application of two flexible thresholds. In Table 6 and Table 7, ‘OF’ and ‘UF’ indicate the overflow and underflow, respectively.

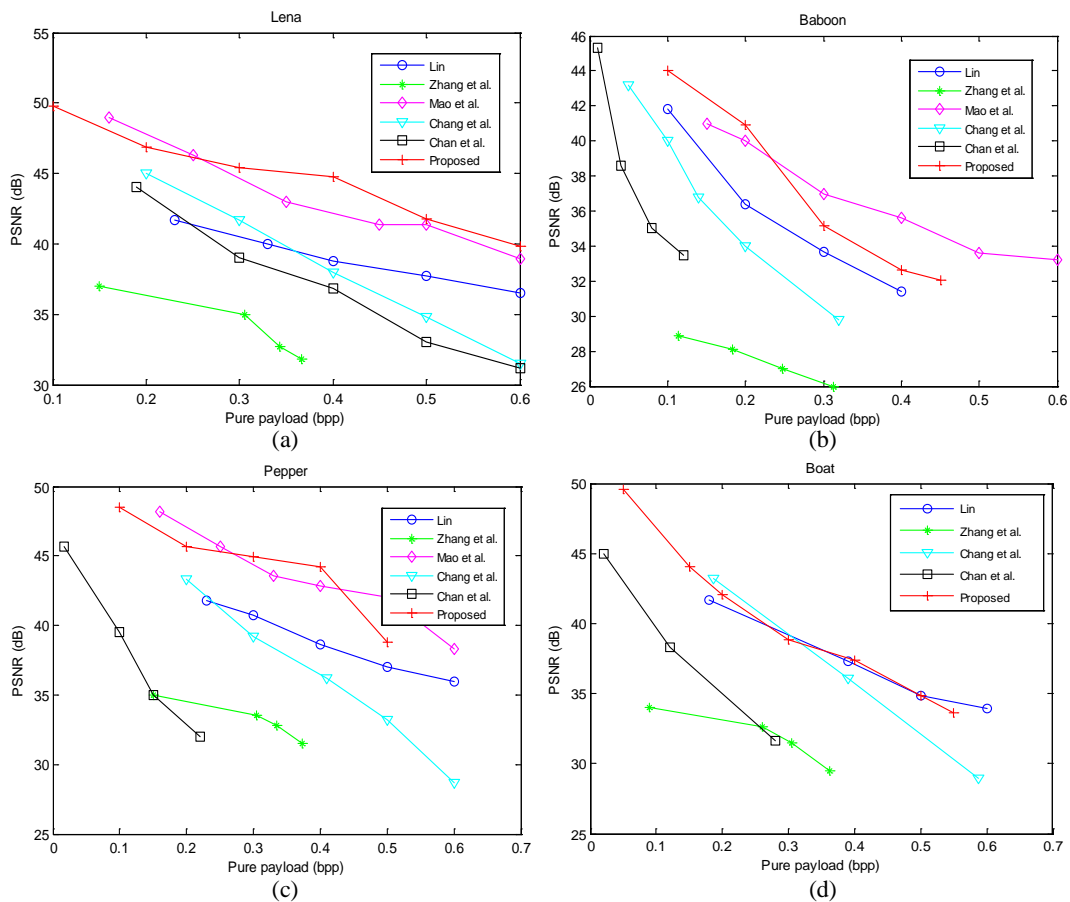


Fig. 8. Performance comparisons with some state-of-the-art approaches. (a) Lena. (b) Baboon. (c) Pepper. (d) Boat.

**Table 6.** Comparison of PSNRs of proposed scheme with that of the GA-based optimization scheme proposed by Arsalan et al. [21] against different pure payload values

| Images                  | Pure payload |       |       |       |       |       |
|-------------------------|--------------|-------|-------|-------|-------|-------|
|                         | 0.1          | 0.2   | 0.3   | 0.4   | 0.5   | 0.6   |
| Lena image              |              |       |       |       |       |       |
| PSNR([21], GA-RevWM)    | 47.8         | 44.5  | 42.8  | 40.7  | 40.2  | 39.8  |
| PSNR(Proposed CE-RevWM) | 49.9         | 47.0  | 45.5  | 45.0  | 41.9  | 39.9  |
| Baboon image            |              |       |       |       |       |       |
| PSNR([21], GA-RevWM)    | 38.7         | 36.5  | 35.4  | 34.7  | —     | —     |
| PSNR(Proposed CE-RevWM) | 44.1         | 41.0  | 35.4  | 32.9  | —     | —     |
| Pepper image            |              |       |       |       |       |       |
| PSNR([21], GA-RevWM)    | 46.5         | 43.5  | 41.4  | 40.3  | 40.0  | —     |
| PSNR(Proposed CE-RevWM) | 48.5         | 45.8  | 45.0  | 44.5  | 38.9  | —     |
| Boat image              |              |       |       |       |       |       |
| PSNR([21], GA-RevWM)    | 45.0         | 41.4  | 38.2  | 37.2  | 36.5  | —     |
| PSNR(Proposed CE-RevWM) | 45.4         | 42.0  | 38.8  | 37.6  | 34.6  | —     |
| Medical image           |              |       |       |       |       |       |
| PSNR([21], GA-RevWM)    | 48.3         | 44.9  | 43.4  | 42.1  | 41.7  | 41.4  |
| PSNR(Proposed CE-RevWM) | 49.9         | 46.9  | 45.6  | 45.2  | 41.1  | 40.8  |
| Artistic image          |              |       |       |       |       |       |
| PSNR([21], GA-RevWM)    | 46.4         | 44.3  | 42.9  | 42.2  | 41.9  | 41.6  |
| PSNR(Proposed CE-RevWM) | 49.4         | 47.1  | 46.0  | 45.5  | 45.0  | 41.7  |
| Antique image           |              |       |       |       |       |       |
| PSNR([21], GA-RevWM)    | UF+OF        | UF+OF | UF+OF | UF+OF | UF+OF | UF+OF |
| PSNR(Proposed CE-RevWM) | 40.3         | 38.1  | 26.1  | 21.1  | —     | —     |
| Calligraphy image       |              |       |       |       |       |       |
| PSNR([21], GA-RevWM)    | UF           | UF    | UF    | UF    | UF    | UF    |
| PSNR(Proposed CE-RevWM) | 44.5         | 43.9  | 43.6  | 43.3  | —     | —     |

‘—’ denotes the corresponding pure payload cannot be embedded.

**Table 7.** Comparison of SSIMs of proposed scheme with that of the GA-based optimization scheme proposed by Arsalan et al. [21] against different pure payload values

| Images                  | Pure payload |        |        |        |        |        |
|-------------------------|--------------|--------|--------|--------|--------|--------|
|                         | 0.1          | 0.2    | 0.3    | 0.4    | 0.5    | 0.6    |
| Lena image              |              |        |        |        |        |        |
| SSIM([21], GA-RevWM)    | 0.9981       | 0.9964 | 0.9946 | 0.9925 | 0.9917 | 0.9913 |
| SSIM(Proposed CE-RevWM) | 0.9986       | 0.9976 | 0.9969 | 0.9966 | 0.9941 | 0.9904 |
| Baboon image            |              |        |        |        |        |        |
| SSIM([21], GA-RevWM)    | 0.9919       | 0.9877 | 0.9850 | 0.9829 | —      | —      |
| SSIM(Proposed CE-RevWM) | 0.9973       | 0.9955 | 0.9879 | 0.9785 | —      | —      |
| Pepper image            |              |        |        |        |        |        |
| SSIM([21], GA-RevWM)    | 0.9981       | 0.9952 | 0.9937 | 0.9916 | 0.9909 | —      |
| SSIM(Proposed CE-RevWM) | 0.9981       | 0.9969 | 0.9963 | 0.9960 | 0.9870 | —      |
| Boat image              |              |        |        |        |        |        |
| SSIM([21], GA-RevWM)    | 0.9970       | 0.9936 | 0.9877 | 0.9862 | 0.9853 | —      |
| SSIM(Proposed CE-RevWM) | 0.9969       | 0.9953 | 0.9896 | 0.9874 | 0.9767 | —      |
| Medical image           |              |        |        |        |        |        |
| SSIM([21], GA-RevWM)    | 0.9980       | 0.9959 | 0.9945 | 0.9934 | 0.9927 | 0.9925 |
| SSIM(Proposed CE-RevWM) | 0.9988       | 0.9976 | 0.9970 | 0.9968 | 0.9917 | 0.9913 |
| Artistic image          |              |        |        |        |        |        |
| SSIM([21], GA-RevWM)    | 0.9961       | 0.9942 | 0.9921 | 0.9909 | 0.9904 | 0.9898 |
| SSIM(Proposed CE-RevWM) | 0.9978       | 0.9966 | 0.9958 | 0.9955 | 0.9950 | 0.9899 |
| Antique image           |              |        |        |        |        |        |

|                         |        |        |        |        |       |       |
|-------------------------|--------|--------|--------|--------|-------|-------|
| SSIM([21], GA-RevWM)    | UF+OF  | UF+OF  | UF+OF  | UF+OF  | UF+OF | UF+OF |
| SSIM(Proposed CE-RevWM) | 0.9969 | 0.9948 | 0.9858 | 0.9631 | —     | —     |
| Calligraphy image       |        |        |        |        |       |       |
| SSIM([21], GA-RevWM)    | UF     | UF     | UF     | UF     | UF    | UF    |
| SSIM(Proposed CE-RevWM) | 0.9991 | 0.9988 | 0.9988 | 0.9988 | —     | —     |

‘—’ denotes the corresponding pure payload cannot be embedded.

## 5. Conclusion

How to achieve an effective tradeoff between the embedding capacity and imperceptibility is an important research issue for reversible watermarking. For this purpose, this paper has reported a new intelligent reversible watermarking algorithm. To preprocess the image with least histogram shift cost against overflow and underflow, two flexible thresholds are applied. Through using the combination between the composite chaos and the average energy of IWT block, the adaptive embedding threshold matrix is generated, accordingly, the PSNR value as high as possible is achieved with a given embedding capacity.

## References

- [1] B. Chen, G. Coatrieux, G. Chen, X. Sun, J. Coatrieux, and H. Shu, “Full 4-D quaternion discrete Fourier transform based watermarking for color images,” *Digital Signal Processing.*, vol. 28, no. 5, pp. 106-119, 2014. [Article \(CrossRef Link\)](#).
- [2] P. Bas and T. Furon, “A new measure of watermarking security: the effective key length,” *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 1, pp. 1306–1317, 2013. [Article \(CrossRef Link\)](#).
- [3] G. Gao, “Composite chaos-based lossless image authentication and tamper localization,” *Multimedia Tools and Applications*, vol. 63, no. 3, pp.947-964, 2013. [Article \(CrossRef Link\)](#).
- [4] Z. Xia, X. Wang, X. Sun, and B. Wang, “Steganalysis of least significant bit matching using multi-order differences,” *Security and Communication Networks*, vol. 7, no. 8, pp. 1283-1291, 2014. [Article \(CrossRef Link\)](#).
- [5] D. Yan, R. Wang, X. Yu, and J. Zhu, “Steganography for MP3 audio by exploiting the rule of window switching,” *Computers & Security*, vol. 31, no. 5, pp. 704-716, 2012. [Article \(CrossRef Link\)](#).
- [6] H. Sadreazami, M. O. Ahmad, and M. N. Swamy, “A study of multiplicative watermark detection in the contourlet domain using alpha-stable distributions,” *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4348-4360, 2014. [Article \(CrossRef Link\)](#).
- [7] Z. Xia, X. Wang, X. Sun, Q. Liu, and N. Xiong, “Steganalysis of LSB matching using differences between nonadjacent pixels,” *Multimedia Tools and Applications*, vol. 75, no. 4, pp. 1947-1962, 2016. [Article \(CrossRef Link\)](#).
- [8] G. Gao and Y. Q. Shi, “Reversible data hiding using controlled contrast enhancement and integer wavelet transform,” *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 2078-2082, 2015. [Article \(CrossRef Link\)](#).
- [9] B. Ou, X. Li, Y. Zhao, R. Ni, and Y. Q. Shi, “Pairwise prediction error expansion for efficient reversible data hiding,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 5010–5021, 2013. [Article \(CrossRef Link\)](#).
- [10] Z. C. Ni, Y. Q. Shi, N. Ansari, and W. Su, “Reversible data hiding,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 3, pp. 354-362, 2006. [Article \(CrossRef Link\)](#).
- [11] J. X. Wang, J. Q. Ni, and J. W. Pan, “A high performance reversible watermarking scheme based on histogram shifting,” *Acta Automatica Sinica*, vol. 38, no. 1, pp.88-96, 2012. [Article \(CrossRef Link\)](#).
- [12] G. Coatrieux, W. Pan, F. Cuppens, and C. Roux, “Reversible watermarking based on invariant image classification and dynamic histogram shifting,” *IEEE Transactions on Information Forensics and*

- Security*, vol. 8, no. 1, pp.111-120, 2013. [Article \(CrossRef Link\)](#).
- [13] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890-896, 2003. [Article \(CrossRef Link\)](#).
- [14] Q. Gu and T. Gao, "A novel reversible robust watermarking algorithm based on chaotic system," *Digital Signal Processing*, vol. 23, no. 5, pp. 213-217, 2013. [Article \(CrossRef Link\)](#).
- [15] D. M. Thodi and J. J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp.721-730, 2007. [Article \(CrossRef Link\)](#).
- [16] I. Dragoi and D. Coltuc, "Local prediction based difference expansion reversible watermarking," *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1779-1790, 2014. [Article \(CrossRef Link\)](#).
- [17] Y. K. Lin, "High capacity reversible data hiding scheme based upon discrete cosine transformation," *Journal of Systems and Software*, vol. 85, no.10, pp. 2395-2404, 2012. [Article \(CrossRef Link\)](#).
- [18] C. C. Chang, P. Y. Pai, C. M. Yeh, and Y. K. Chan, "A high payload frequency-based reversible image hiding method," *Information Sciences*, vol. 180, no.11, pp. 2286-2298, 2010. [Article \(CrossRef Link\)](#).
- [19] S. Zhang, T. Gao, and L. Yang, "A reversible data hiding scheme based on histogram modification in integer DWT domain for BTC compressed images," *International Journal of Network Security*, vol. 18, no. 4, pp. 718-727, 2016.
- [20] G. Xuan, C. Yang, Y. Zhen, Y. Q. Shi, and Z. Ni, "Reversible data hiding using integer wavelet transform and companding technique," *Lecture Notes in Computer Science*, vol. 3304, pp. 115-124, 2005. [Article \(CrossRef Link\)](#).
- [21] M. Arsalan, S. A. Malik, and A. Khan, "Intelligent reversible watermarking in integer wavelet domain for medical images," *Journal of Systems and Software*, vol. 85, no. 4, pp. 883-894, 2012. [Article \(CrossRef Link\)](#).
- [22] Q. Mao, F. Li, C. C. Chang, "Reversible data hiding with oriented and minimized distortions using cascading trellis coding," *Information Sciences*, vol. 317, no.10, pp. 170-180, 2015. [Article \(CrossRef Link\)](#).
- [23] X. Wen, L. Shao, Y. Xue, and W. Fang, "A rapid learning algorithm for vehicle classification," *Information Sciences*, vol. 295, no. 1, pp. 395-406, 2015. [Article \(CrossRef Link\)](#).
- [24] Z. Xia, X. Wang, X. Sun, Q. Liu, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 2, pp. 340-352, 2015. [Article \(CrossRef Link\)](#).
- [25] B. Gu, V. S. Sheng, K. Y. Tay, W. Romano, and S. Li, "Incremental support vector learning for ordinal regression," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 7, pp. 1403-1416, 2015. [Article \(CrossRef Link\)](#).
- [26] B. Gu and V. S. Sheng, "A robust regularization path algorithm for v-support vector classification," *IEEE Transactions on Neural Networks and Learning Systems*, 2016. [Article \(CrossRef Link\)](#).
- [27] B. Gu, V. S. Sheng, Z. Wang, D. Ho, S. Osman, and S. Li, "Incremental learning for v-support vector regression," *Neural Networks*, vol. 67, pp. 140-150, 2015. [Article \(CrossRef Link\)](#).
- [28] X. Tong, M. Cui, "Image encryption based on disturbed composite chaos," *Science China-Information Science*, vol. 39, no. 6, pp. 588-597, 2009.
- [29] S. Pincus, "Approximate entropy (ApEn) as a complexity measure," *Chaos*, vol. 5, no. 1, pp. 110-117, 1995. [Article \(CrossRef Link\)](#).
- [30] L. H. Zhang, X. F. Liao, X. B. Wang, "An image encryption approach based on chaotic maps," *Chaos Soliton Fract*, vol. 24, no.3, pp.759-765, 2005. [Article \(CrossRef Link\)](#).
- [31] A. Rukhin, J. Soto, and J. Nechvatal, "A statistical test suite for random and pseudorandom number generator for cryptographic applications." [Article \(CrossRef Link\)](#).
- [32] A. R. Calderbank, I. Daubechies, W. Sweldens, and B. L. Yeo, "Wavelet transforms that map integers to integers," *Applied and Computational Harmonic Analysis*, vol. 5, no. 3, pp. 332-369, 1998. [Article \(CrossRef Link\)](#).
- [33] G. Xuan, X. Tong, J. Teng, X. Zhang, and Y. Q. Shi, "Optimal histogram-pair and prediction-error based image reversible data hiding," *Lecture Notes in Computer Science*, vol. 7809, pp. 368-383, 2013. [Article \(CrossRef Link\)](#).

- [34] Y. K. Chan, W. T. Chen, S. S. Yu, Y. A. Ho, C. S. Tsai, and Y. P. Chu, "A HDWT-based reversible data hiding method," *Journal of Systems and Software*, vol. 82, no. 3, pp. 411-421, 2009.  
[Article \(CrossRef Link\)](#).



**Guangyong Gao** received the Ph.D. Degree from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2012. Currently he is an associate professor with the School of Information Science and Technology, Jiujiang University. His research interests include Multimedia Information Security, Digital Image Processing and Computer Networks Security.



**Yun-Qing Shi** has been with the New Jersey Institute of Technology, Newark, NJ, USA, since 1987. He received the M.S. degree from Shanghai Jiao Tong University, Shanghai, China, and the Ph.D. degree from University of Pittsburgh, Pittsburgh, PA, USA. His current research interests include digital data hiding, forensics and information assurance, and visual signal processing and communications. He is an author or co-author of 300 papers, one book, and five book chapters, and an editor of ten books. He holds 27 U.S. patents. He received the Innovators Award 2010 by the New Jersey Inventors Hall of Fame for Innovations in Digital Forensics and Security. His U.S. patent 7 457 341 entitled *System and Method for Robust Reversible Data Hiding and Data Recovery in the Spatial Domain* won the 2010 Thomas Alva Edison Patent Award by the Research and Development Council of New Jersey. He served as an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS (II), and a few other journals; he also served as the Technical Program Chair of IEEE ICME in 2007, a Co-Technical Chair of IWDW in 2006, 2007, 2009, 2010, 2011, 2012, and 2013, and IEEE MMSP in 2005, a Co-General Chair of IEEE MMSP in 2002, and a Distinguished Lecturer of IEEE CASS. He is a member of a few IEEE technical committees.



**Xingming Sun** received his BS in mathematics from Hunan Normal University, China, in 1984; his MS in computing science from Dalian University of Science and Technology, China, in 1988; and his PhD in computing science from Fudan University, China, in 2001. He is currently a professor at the Department of Computer and Software, Nanjing University of Information Science and Technology, China. In 2006, he visited the University College London, UK; he was a visiting professor in University of Warwick, UK, between 2008 and 2010. His research interests include network and information security, database security, and natural language processing.



**Caixue Zhou** received the B.S. in Computer Science Department from Fudan University in 1988, Shanghai, China and the M.S. in Space College of Beijing University of Aeronautics and Astronautics in 1991, Beijing, China. He is an Associate Professor in the School of Information Science and Technology, Jiujiang University, Jiujiang, China since 2007. He is a member of the CCF (China Computer Federation), and a member of CACR (Chinese Association for Cryptologic Research). His research interests include applied cryptography, security of computer networks.



**Zongmin Cui** received the B.E. degree from Southeast University in 2002 and the M.S. degree from Huazhong University of Science and Technology in 2006. He received the Ph.D. Degree from Huazhong University of Science and Technology in 2014. He is currently an associate professor with the School of Information Science and Technology, Jiujiang University. His research interests include cloud security, authorization update, key management, access control, and publish/subscribe system.



**Liya Xu** received the B.E degree from Wuhan University in 2006 and the M.S. degree from computer school of Wuhan university in 2010. He received the doctor degree from computer school of Wuhan university in 2014. He is currently an associate professor with the School of Information Science and Technology, Jiujiang University, Jiujiang, China. His research interests include VANET, internet of things, DTN.