Lossless embedding: A visually meaningful image encryption algorithm based on hyperchaos and compressive sensing

Xing-Yuan Wang(王兴元)[1,2], Xiao-Li Wang(王晓丽)[1], Lin Teng(滕琳)[1,†], Dong-Hua Jiang(蒋东华)[3], and Yongjin Xian(咸永锦)[1,4]

[1] *School of Information Science and Technology, Dalian Maritime University, Dalian* 116026, *China*
[2] *Guangxi Key Laboratory of Multi-source Information Mining & Security, Guangxi Normal University, Guilin* 541004, *China*
[3] *School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou* 511400, *China*
[4] *School of Cyber Security, Qilu University of Technology (Shandong Academy of Sciences), Jinan* 250353, *China*

(Received 29 August 2022; revised manuscript received 2 November 2022; accepted manuscript online 9 November 2022)

A novel visually meaningful image encryption algorithm is proposed based on a hyperchaotic system and compressive sensing (CS), which aims to improve the visual security of steganographic image and decrypted quality. First, a dynamic spiral block scrambling is designed to encrypt the sparse matrix generated by performing discrete wavelet transform (DWT) on the plain image. Then, the encrypted image is compressed and quantified to obtain the noise-like cipher image. Then the cipher image is embedded into the alpha channel of the carrier image in portable network graphics (PNG) format to generate the visually meaningful steganographic image. In our scheme, the hyperchaotic Lorenz system controlled by the hash value of plain image is utilized to construct the scrambling matrix, the measurement matrix and the embedding matrix to achieve higher security. In addition, compared with other existing encryption algorithms, the proposed PNG-based embedding method can blindly extract the cipher image, thus effectively reducing the transmission cost and storage space. Finally, the experimental results indicate that the proposed encryption algorithm has very high visual security.

Keywords: chaotic image encryption, compressive sensing, meaningful cipher image, portable network graphics, image encryption algorithm

PACS: 05.45.Gg, 07.05.Pj, 05.45.Jn **DOI:** 10.1088/1674-1056/aca149

1. Introduction

With the rapid development of big data technology, the amount of data transmitted in the network is also increasing rapidly. As the mainstream form of multimedia information transmission, digital image is widely used in industry, military, medical, and other fields. At the same time, with the increasing security requirements of modern applications, the security of digital image has also attracted people's great attention. As an effective means to protect digital image information, a large number of image encryption algorithms have been proposed.[1–3]

At present, image encryption technology is mainly divided into two categories. The first category is the traditional image encryption algorithm. That is, the plain image is processed and converted into a noise-like cipher image. Many scholars propose different image encryption schemes based on quantum computation,[4,5] chaos theory,[6–8] fractal sorting matrix,[9–11] DNA coding,[12,13] fractal control theory,[14–16] and other technologies.[17–19] Specifically, this type of image encryption algorithms is mainly composed of scrambling and diffusion, which encrypt the image by changing pixel position and pixel value, respectively. However, the noise-like cipher image generated by such algorithms are easy to attract the attention of hackers and thus suffer from various malicious attacks. The other is the visual image encryption algorithm,[20–24] which first converts the plain image into a noise-like image, and then embeds it into the carrier image to generate a visually meaningful steganographic image. Compared with traditional image encryption schemes, the steganographic images are not easy to attract hackers' attention during transmission, thereby reducing the possibility of being attacked. Therefore, the research on the visual image encryption algorithm is of great significance.

Bao *et al.*[25] first proposed a visual image encryption algorithm composed of pre-encryption and embedding process, which can not only protect the security of the content carried by digital image, but also reduce the possibility of being attacked. Unfortunately, the volume of steganographic image generated by this scheme is four times that of the plain image, which leads to the extra transmission bandwidth and storage space. However, improved schemes have been proposed. Chai *et al.*[26] used compressive sensing to encrypt and compress the plain image, and then embedded the cipher image into the carrier image to generate steganographic image with the same size. Then, Wang *et al.*[27] proposed an embedding scheme based on least significant bit (LSB) and integer wavelet transform. The embedding and corresponding extracting process in this scheme are completely reversible. Subsequently, in

†Corresponding author. E-mail: tenglin@dlmu.edu.cn
© 2023 Chinese Physical Society and IOP Publishing Ltd

order to improve the robustness of the encryption algorithm, Zhu *et al.*[28] proposed a visually meaningful image encryption algorithm based on singular value decomposition embedding. The cipher image was embedded into the diagonal matrix obtained by preforming singular value decomposition on carrier image. In addition, some embedding methods based on pixel value differencing,[29] histogram shifting,[30] reversible color transform,[31] Schur decomposition,[32] most significant bit[33] and others[34–36] were also proposed. Although the existing visually meaningful image encryption algorithms have greatly improved the visual security, there are also some problems: Embedding the cipher image into the carrier image will destroy the pixel structure of the carrier image, and affect the imperceptibility of the steganographic image. There is truncation error in the encryption process, which will decrease the quality of the decrypted image. The extracting process requires the carrier image, which requires additional storage space and transmission cost.

In order to solve these problems, this paper proposes a visual image encryption algorithm combining hyperchaotic system, compressive sensing and PNG-based embedding. The proposed algorithm consists of two main stages. In the first stage, DWT is applied to plain image. Meanwhile, the hash value of plain image is used to control the hyperchaotic system to generate scrambling and measurement matrix. Then the sparse matrix after threshold processing is encrypted and measured by dynamic spiral block scrambling and compressive sensing to obtain the cipher image. In the second stage, the cipher image is embedded into the alpha channel of the PNG carrier image to obtain a visually meaningful steganographic image. Moreover, the embedding position of cipher image elements is controlled by chaotic sequences. The prominent contributions of this paper are as follows:

(i) A lossless embedding method and dynamic spiral block scrambling are designed for visual image encryption.

(ii) The proposed visually meaningful image encryption algorithm has very high visual security. Additionally, the quality of decrypted image is independent of the carrier image. Therefore, the carrier image can be selected flexibly.

(iii) Our encryption algorithm can effectively resist the chosen-plaintext attacks by associating with the hash value of the plain image.

(iv) The proposed decryption algorithm can extract the cipher image blindly, which effectively reduces the transmission cost and storage space.

The rest of this article is organized as follows. In Section 2 we introduce the relevant knowledge. In Section 3 wedescribes in detail the image encryption and the corresponding decryption algorithm. The simulation experiments and performance analyses are given in Sections 4 and 5. In Section 6, we compare the proposed algorithm with other existing visual image encryption algorithms. Lastly, we present a concise conclusion in Section 7.

2. Relevant knowledge

2.1. Hyperchaotic Lorenz system

In this paper, the hyperchaotic Lorenz system is used to generate the pseudo-random sequences, and its dynamic equation[37] is expressed as follows:

$$\begin{cases} \dot{x} = a(y-x)+w, \\ \dot{y} = cx-y-xz, \\ \dot{z} = xy-bz, \\ \dot{w} = -yz+rw, \end{cases} \tag{1}$$

where a, b, c, and r are the control parameters of the system. When $a=10$, $b=8/3$, $c=28$, and $r=-1$, the system is in the hyper-chaotic state, and the corresponding Lyapunov exponents are $\lambda_1=0.3381$, $\lambda_2=0.1586$, $\lambda_3=0$, $\lambda_4=-15.1725$. In our scheme, four chaotic sequences are generated for encryption and embedding.

2.2. Compressive sensing

Compressive sensing (CS) is used to simultaneously sample and compress signals. According to the theory, if the signal is sparse, it can be reconstructed with less sampling data.[38]

Suppose that a one-dimensional signal $\boldsymbol{x} \in R^n$ can be represented by a set of linear combinations of sparse basis $\boldsymbol{\Psi}^{\mathrm{T}} = \{\psi_1, \psi_2, \ldots, \psi_N\}$. This process is denoted by

$$\boldsymbol{x} = \sum_{i=1}^{N} \psi_i s_i = \boldsymbol{\Psi} \boldsymbol{s}, \tag{2}$$

where $s_i = \{s_1, s_2, \ldots, s_N\}$ is the coefficient vector of signal $\boldsymbol{x}$. When the signal $\boldsymbol{x}$ has only K ($K \ll N$) nonzero coefficients on a certain basis $\boldsymbol{\Psi}$, $\boldsymbol{\Psi}$ is called the sparse basis of the signal $\boldsymbol{x}$, and $\boldsymbol{s}$ is the sparse coefficient.

The signal $\boldsymbol{x}$ is projected onto a set of measurement matrices $\boldsymbol{\Phi}$ to obtain the measurement value $\boldsymbol{y}$. That is, an $M \times N$ ($M < N$) measurement matrix $\boldsymbol{\Phi}$, which is not related to the sparse basis matrix, is used to linearly project the signal $\boldsymbol{x}$ to obtain the linear measurement value $\boldsymbol{y}$. This process is expressed by

$$\boldsymbol{y} = \boldsymbol{\Phi} \boldsymbol{x}. \tag{3}$$

Since the signal $\boldsymbol{x}$ can be sparsely represented: $\boldsymbol{x} = \boldsymbol{\Psi} \boldsymbol{s}$, the above formula can be rewritten as

$$\boldsymbol{y} = \boldsymbol{\Phi} \boldsymbol{x} = \boldsymbol{\Phi} \boldsymbol{\Psi} \boldsymbol{s} = \boldsymbol{\Theta} \boldsymbol{s}, \tag{4}$$

where $\boldsymbol{\Theta}$ is called the sensing matrix. When the matrix $\boldsymbol{\Theta}$ satisfies the RIP criterion,[39] the reconstruction process of the signal $\boldsymbol{x}$ can be transformed into an optimization problem under the minimum norm of l_1,

$$\min \|s\|_1 \quad \text{s.t.} \quad y = \boldsymbol{\Theta} \boldsymbol{s}. \tag{5}$$

At present, the commonly used effective reconstruction algorithms include matching pursuit (MP), orthogonal matching pursuit (OMP), gradient projection (GP), smooth l_0 norm (SL), and so on. Because the smooth l_0 norm has the characteristics of low calculation amount, high estimation accuracy, and strong robustness. It is selected as the reconstruction algorithm in this paper.

2.3. Generating measurement matrix

When the signal is sparse in the transform domain, the original signal can be compressed by the measurement matrix $\boldsymbol{\Phi}$. The hyperchaotic Lorenz system is used to generate measurement matrix in this paper. The specific generation steps are as follows:

Step 1 The initial values x, y, z and w are used to iterate the hyperchaotic Lorenz system for $1000+mNd$ times, and the first 1000 values are discarded to obtain a chaotic sequence X,

$$X = \{x_1, x_2, x_3, \ldots, x_{mNd}\}, \tag{6}$$

where parameter d is the sampling interval.

Step 2 The chaotic sequence X is sampled equidistantly to obtain X', This process is denoted in Eq. (7).

$$X' = X_{100+kd}, k = 0, 1, \ldots, mN-1. \tag{7}$$

Step 3 To map the range of sequence values to $(-1, 1)$, the sequence X' is processed by using the following equation to obtain a new sequence X'':

$$X'' = 1 - 2X'. \tag{8}$$

Step 4 The obtained sequence X'' is reordered in a column-wise manner, and then the measurement matrix $\boldsymbol{\Phi}_{m\times N}$ is constructed as

$$\boldsymbol{\Phi} = \sqrt{\frac{2}{m}} \begin{bmatrix} X''_0 & X''_m & \cdots & X''_{mN-m} \\ X''_1 & X''_{m+1} & \cdots & X''_{mN-m+1} \\ \vdots & \vdots & \vdots & \vdots \\ X''_{m-1} & X''_{2m-1} & \cdots & X''_{mN} \end{bmatrix}. \tag{9}$$

2.4. Dynamic spiral block scrambling

Spiral scanning can be used to scramble the plain image, which can interfere with the high correlation between image pixels to increase the security level of the encryption algorithm. However, the traditional spiral scrambling has its limitation. After scrambling, the value of the special position in the matrix remains unchanged, such as the position of the number 9 in Fig. 1. In addition, some classical scrambling algorithms, such as Arnold transform, have periodicity, which means that the image will return to the original image after a certain number of transformations. In order to solve these problems and to achieve better scrambling effect, a new dynamic spiral block scrambling algorithm is proposed in this paper. This method not only ensures the security of the algorithm, but also changes the positions of all pixels in only one round of encryption. The specific steps are as follows:

Step 1 Select the appropriate length N_0 to divide the image with size of $M\times N$ into N_S sub-blocks with size of $N_0\times N_0$, the number of sub-blocks is expressed by

$$N_S = \left\lceil \frac{M}{N_0} \right\rceil \times \left\lceil \frac{N}{N_0} \right\rceil. \tag{10}$$

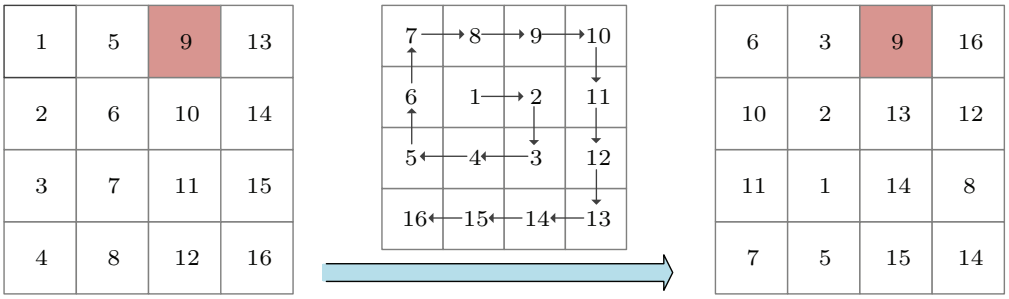


Fig. 1. 4×4 matrix transferred spiral scrambling process

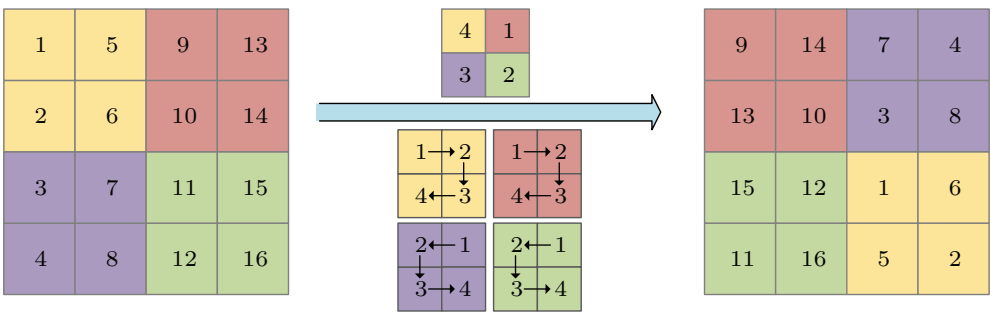


Fig. 2. A 4×4 matrix transferred by dynamic spiral block scrambling via a 2×2 sub-block

Here, $\lceil x \rceil$ represents the smallest integer value not less than x. If N_0 cannot divide the length and width of the image, the image is filled in the shortest sequence, which is randomly composed of integers between 0 and 255, and the number of sub-blocks is determined.

Step 2 The chaotic sequence Y with length of N_S is used to scramble the pixels in the sub-block. If Y_i is odd, a clockwise spiral scrambling is performed on the i-th sub-block matrix. On the contrary, a counterclockwise spiral scrambling is performed.

$$Y = \text{ceil}(Y \times 255), \tag{11}$$

where ceil (x) represents the smallest integer value not less than x.

Step 3 Sort the chaotic sequence Y with length of N_S in ascending order to obtain the scrambling sequence S_1, and use S_1 to exchange the positions of the sub-blocks,

$$[\sim, S_1] = \text{sort}(Y). \tag{12}$$

Due to the different pseudo-random sequences generated, the scrambled matrix is also different. The algorithm can change the position of all pixels with only one scrambling. For example, a 4×4 matrix is divided into four 2×2 sub-blocks, and then the process of dynamic spiral block scrambling is shown in Fig. 3.

2.5. Portable network graphics

Portable network graphics (PNG) is a bitmap graphics format that can represent images lossless. It consists of an 8-byte PNG file signature field and more than 3 subsequent data blocks (IHDR, IDAT, IEND). PNG is designed to replace the two file formats GIF and TIFF. At the same time, it also adds some features that the GIF file format does not have, such as supporting for transparency. PNG can define 256 transparency levels for the image, allowing the edge of the color image to blend smoothly with any background, and because of this, the jagged edges of the image are completely eliminated. Because of the characteristics of small size, lossless compression and better network transmission display, PNG is widely used as information transmission media in the network.

3. Algorithm description

3.1. The encryption process

The flow chart of the encryption scheme proposed in this paper is shown in Fig. 3. The algorithm mainly includes two stages. In the first stage, the plain image is encrypted and compressed by dynamic spiral block scrambling and compressive sensing to obtain the cipher image. Then, in the second stage, the cipher image is embedded into the alpha channel of the PNG carrier image to generate the steganographic image.

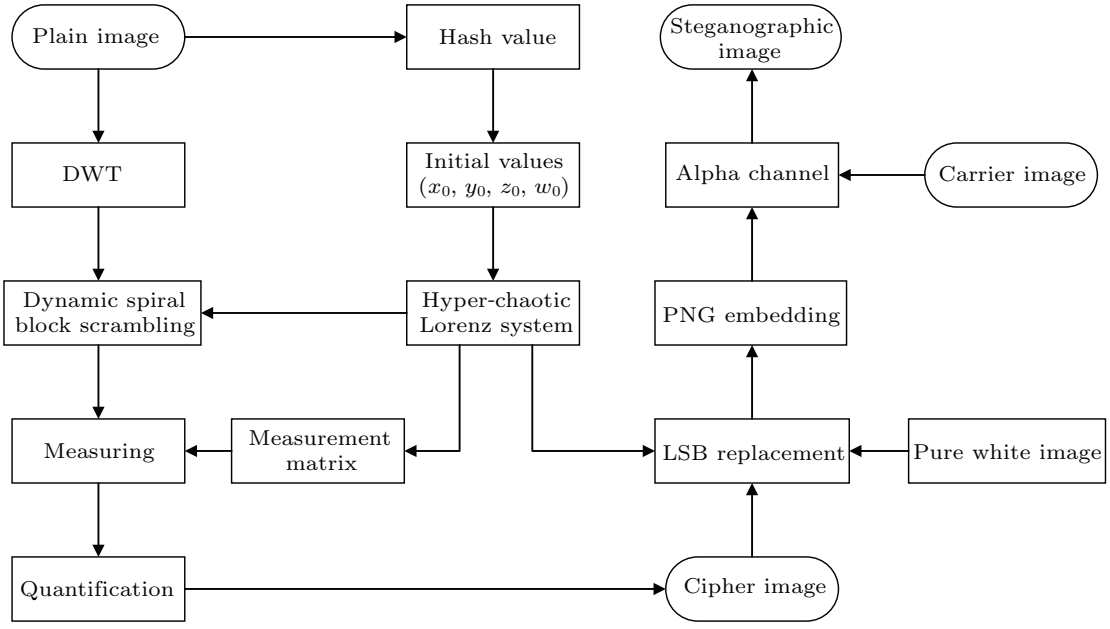


Fig. 3. The flow chart of the proposed encryption algorithm.

In the encryption process, the hash value of plain image is used to generate the initial values of the hyperchaotic Lorenz system, and then the hyperchaotic system is utilized to construct measurement matrix, scrambling sequence and embedding sequence. Assume that the size of the plain image $\boldsymbol{P}$ and the carrier image $\boldsymbol{C}$ are both $M \times N$.

3.1.1. Generating the cipher image based on dynamic spiral block scrambling and CS

Step 1 Perform the hash algorithm on the pixel values of the plain image $\boldsymbol{P}$ to generate 256-bit hash value and represent it as hexadecimal number H. Each 16-bit hash value in H is

processed according to Eq. (13) to generate k_1, k_2, k_3 and k_4,

$$\begin{cases} k_1 = \dfrac{\text{hex2dec}(H(1:16))}{2^{64}}, \\ k_2 = \dfrac{\text{hex2dec}(H(17:32))}{2^{64}}, \\ k_3 = \dfrac{\text{hex2dec}(H(33:48))}{2^{64}}, \\ k_4 = \dfrac{\text{hex2dec}(H(49:64))}{2^{64}}. \end{cases} \tag{13}$$

Step 2 Calculate the average value of all pixels of the plain image $\boldsymbol{P}$ and normalize to get r. The formula is given in the following. Then k_1, k_2, k_3, k_4, and r are used to calculate the initial values x, y, z, w of the hyperchaotic system. This process is also given in the following:

$$r = \frac{\left(\sum_{i=1}^{M}\sum_{j=1}^{N}\boldsymbol{P}(i,j)\right)}{(255\times M\times N)}, \tag{14}$$

$$\begin{cases} x_0 = ((k_1 + r) \bmod 1), \\ y_0 = ((k_2 + r) \bmod 1), \\ z_0 = ((k_3 + r) \bmod 1), \\ w_0 = ((k_4 + r) \bmod 1). \end{cases} \tag{15}$$

Step 3 Use the initial values x_0, y_0, z_0, and w_0 to iterate the hyperchaotic Lorenz system 1000 times and discard these values (to avoid transient effects). Continue to iterate to get the chaotic sequences $\boldsymbol{X}$, $\boldsymbol{Y}$, $\boldsymbol{Z}$, and $\boldsymbol{W}$. We have

$$\begin{cases} X = \{x_1, x_2, \ldots, x_{mNd}\}, \\ Y = \{y_1, y_2, \ldots, y_{N_S}\}, \\ Z = \{z_1, z_2, \ldots, z_{mN}\}, \\ W = \{w_1, w_2, \ldots, w_{MN}\}. \end{cases} \tag{16}$$

Step 4 Use the discrete wavelet transform to sparse the plain image $\boldsymbol{P}$ to obtain the sparse coefficient matrix $\boldsymbol{P}_1$ with the same size. This process is expressed as

$$\boldsymbol{P}_1 = \boldsymbol{\Psi} \times \boldsymbol{P} \times \boldsymbol{\Psi}', \tag{17}$$

where $\boldsymbol{\Psi}$ is the orthogonal wavelet matrix, and $\boldsymbol{\Psi}'$ is its transposed matrix.

Step 5 If the absolute values of the coefficient matrix $\boldsymbol{P}_1$ are less than or equal to the setting threshold, these values are forced to be set to 0 to obtain the matrix $\boldsymbol{P}_2$. The chaotic sequence Y and the scrambling sequence S_1 are used to perform the dynamic spiral block scrambling on the matrix $\boldsymbol{P}_2$, and the scrambling coefficient matrix $\boldsymbol{P}_3$ is obtained.

Step 6 The chaotic sequence generated by the hyperchaotic Lorenz system is used to generate the measurement matrix $\boldsymbol{\Phi}$ with size of $m \times N$, where $m = \lfloor CR \times M \rfloor$, and CR is the compression rate set by the user. The specific construction process is described in Subsection 2.3.

Step 7 Use measurement matrix $\boldsymbol{\Phi}$ to measure matrix $\boldsymbol{P}_3$, the compressed matrix $\boldsymbol{P}_4$ with size of $m \times N$ is obtained. The measurement formula is

$$\boldsymbol{P}_4 = \boldsymbol{\Phi} \times \boldsymbol{P}_3. \tag{18}$$

Step 8 Quantize the measurement value matrix $\boldsymbol{P}_4$ to the range of 0–255 according to the following equation, the cipher image $\boldsymbol{P}_5$ is obtained.

$$\boldsymbol{P}_5 = \text{floor}(255 \times (\boldsymbol{P}_4 - \min)/(\max - \min)), \tag{19}$$

where max and min represent the maximum and minimum of matrix $\boldsymbol{P}_4$, respectively, and floor (x) represents the largest integer not greater than x.

3.1.2. Embedding the cipher image into the carrier image

Step 1 Convert the cipher image $\boldsymbol{P}_5$ into a binary one-dimensional array with size of $m \times N$, denote as E. Take the adjacent two bits of each element in E as a group, store them successively in the low two bits of four binary arrays with size of $m \times N$, namely E_1, E_2, E_3, and E_4, and set the value of the remaining bits as 0.

Step 2 Construct a matrix $\boldsymbol{F}$ whose size of $M \times N$ and all the values are 255, and divide it into four one-dimensional arrays F_1, F_2, F_3, F_4 with size of$m \times N$ according to

$$\begin{aligned} F_1 &= \boldsymbol{F}(1 : M \times N/4), \\ F_2 &= \boldsymbol{F}(M \times N/4 + 1 : M \times N/2), \\ F_3 &= \boldsymbol{F}(M \times N/2 + 1 : 3 \times M \times N/4), \\ F_4 &= \boldsymbol{F}(3 \times M \times N/4 + 1 : M \times N). \end{aligned} \tag{20}$$

Step 3 Convert the elements in the one-dimensional array F_1, F_2, F_3, F_4 into binary, and set the value of the lower 2 bits of each element to 0 to get F_1', F_2', F_3', F_4'.

Step 4 Sort the chaotic sequence W with length of $m \times N$ to get the random replacement sequence S_3. The formula is given as follows:

$$[\sim, S_3] = \text{sort}(W). \tag{21}$$

Step 5 Replace the lower 2 bits of $F_1'(S_3(i))$, $F_2'(S_3(i_7))$, $F_3'(S_3(i_7))$, $F_4'(S_3(i))$ with the lower 2 bits of $E_1(i)$, $E_2(i)$, $E_3(i)$, $E_4(i)$. The process can be expressed by

$$\begin{aligned} F_1'(S_3(i_7))F_1'(S_3(i_8)) &= E_1(i_7)E_1(i_8), \\ F_2'(S_3(i_7))F_2'(S_3(i_8)) &= E_2(i_7)E_2(i_8), \\ F_3'(S_3(i_7))F_3'(S_3(i_8)) &= E_3(i_7)E_3(i_8), \\ F_4'(S_3(i_7))F_4'(S_3(i_8)) &= E_4(i_7)E_4(i_8). \end{aligned} \tag{22}$$

Step 6 Convert the elements in F_1', F_2', F_3', F_4' to decimal, then compose them into a one-dimensional array F' with size of $M \times N$, and finally reshape the one-dimensional array F' into an $M \times N$ matrix $\boldsymbol{F}''$. The formulas for this step are described as follows:

$$F' = [F_1', F_2', F_3', F_4'], \tag{23}$$

$$\boldsymbol{F}'' = \text{reshape}(F', M, N). \tag{24}$$

Step 7 Write $\boldsymbol{F}''$ into the alpha channel of the PNG format carrier image, and finally get the PNG format steganographic image $\boldsymbol{S}$ with the same size.

3.2. The decryption process

The decryption process is the reverse process of the encryption process, mainly including extraction and reconstruction. It is worth mentioning that the embedding process and the corresponding extraction process are completely reversible. During decryption, key parameters, such as 256-bit hash value, r, and N, must be sent to the receiver. Figure 4 is the decryption flow chart, the decryption steps are given in the following.

Step 1 Extract the cipher information from the alpha channel of the PNG steganographic image $\boldsymbol{S}$, and the matrix $\boldsymbol{F}''$ is obtained.

Step 2 The matrix $\boldsymbol{F}''$ is sequentially divided into four one-dimensional binary arrays F_1', F_2', F_3', F_4' with size of $m \times N$. The sequence S_3 is used to perform the inverse random replacement operation to obtain E_1, E_2, E_3, E_4, the formula is given as follows:

$$\begin{aligned}
E_1(i_7)E_1(i_8) &= F_1'(S_3(i_7))F_1'(S_3(i_8)),\\
E_2(i_7)E_2(i_8) &= F_2'(S_3(i_7))F_2'(S_3(i_8)),\\
E_3(i_7)E_3(i_8) &= F_3'(S_3(i_7))F_3'(S_3(i_8)),\\
E_4(i_7)E_4(i_8) &= F_4'(S_3(i_7))F_4'(S_3(i_8)).
\end{aligned} \tag{25}$$

Step 3 The lower 2 bits of each element in E_1, E_2, E_3, E_4 are sequentially integrated into the one-dimensional binary array E with size of $m \times N$. Convert the elements in E to decimal, and reshape E into an $m \times N$ matrix $\boldsymbol{P}_5$ according to the following equation. $\boldsymbol{P}_5$ is the cipher image

$$\boldsymbol{P}_5 = \text{reshape}(E, m, N). \tag{26}$$

Step 4 The inverse quantization process of matrix $\boldsymbol{P}_5$ is performed according to the following equation to obtain the measurement value matrix $\boldsymbol{RP}_4$:

$$\boldsymbol{RP}_4 = \frac{\boldsymbol{P}_5(\max - \min)}{255} + \min. \tag{27}$$

Step 5 Generate the measurement matrix $\boldsymbol{\Phi}$ according to the method in Subsection 2.3, and then use the smooth l_0 norm reconstruction algorithm to obtain $\boldsymbol{RP}_3$ from $\boldsymbol{RP}_4$. This process can be expressed by

$$\boldsymbol{RP}_3 = SL_0(\boldsymbol{RP}_4, \boldsymbol{\Phi}). \tag{28}$$

Step 6 Use the sequence Y and S_1 to perform inverse dynamic spiral block scrambling on matrix $\boldsymbol{RP}_3$ to generate the matrix $\boldsymbol{RP}_1$. Then inverse wavelet discrete transform (IDWT) is applied to $\boldsymbol{RP}_1$ to obtain the final decrypted image $\boldsymbol{RP}$.

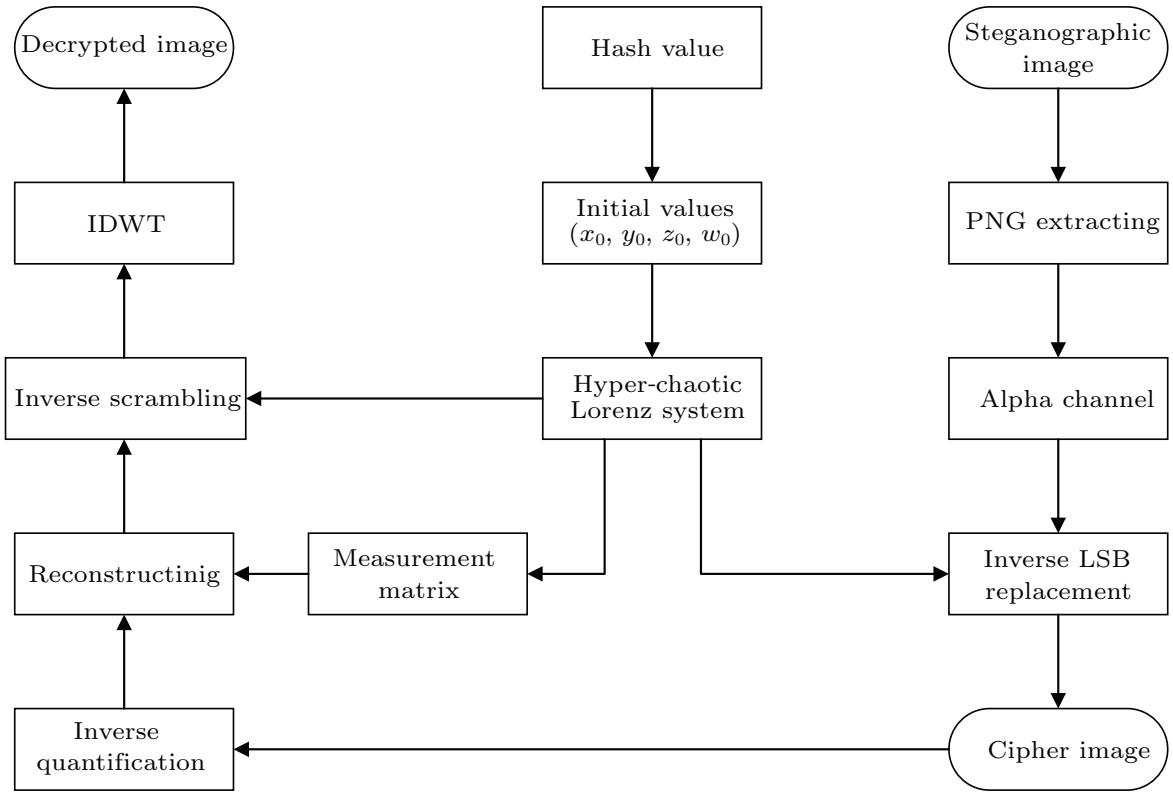


Fig. 4. The flow chart of the proposed decryption algorithm.

4. Simulation results

To verify validity of the proposed algorithm, we carry out experiments on a desktop computer (2.9 GHz i5-9400F CPU, 8G RAM), and simulation platform is Matlab R2020a. The parameters during the encryption process are set as follows: $d = 25$, $TS = 25$, $CR = 0.25$, and $N = 2$.

4.1. Encryption and decryption results

Eight gray images with the size of 512×512 are selected as test images in experiment, the results of encryption and decryption are given in Fig. 5. Each group of simulation experiments is represented by a column in the graph. The third row is the noise-like secret image obtained by dynamic spiral block scrambling and compressive sensing technology. After embedding the carrier image, the generated steganographic image is visually basically consistent with the carrier image, and has very high imperceptibility. To make the experimental results more convincing, the peak signal-to-noise ratio (PSNR)[40] and the mean structural similarity (MSSIM)[41] are used to quantitatively analyze the imperceptibility of steganographic image and the quality of decrypted image. The experimental data obtained is shown in Table 1, where $\mathrm{PSNR_{dec}}$ represents the PSNR between the plain image and the decrypted image, and $\mathrm{PSNR_{ste}}$ and $\mathrm{MSSIM_{ste}}$ represent the PSNR and MSSIM between the carrier image and the steganographic image. As shown in the table, the $\mathrm{PSNR_{ste}}$ data are all greater than 48 dB, and the $\mathrm{MSSIM_{ste}}$ data are approximately equal to one. It further verifies that the steganographic image obtained by this algorithm has very high visual security. In addition, the value of $\mathrm{PSNR_{dec}}$ is also greater than 35 dB. Because the embedding and extracting process is completely reversible, high-quality decrypted images can be obtained. In a word, the experimental results show that the proposed algorithm generates steganographic image with high imperceptibility, and the decrypted image quality is also very satisfactory.

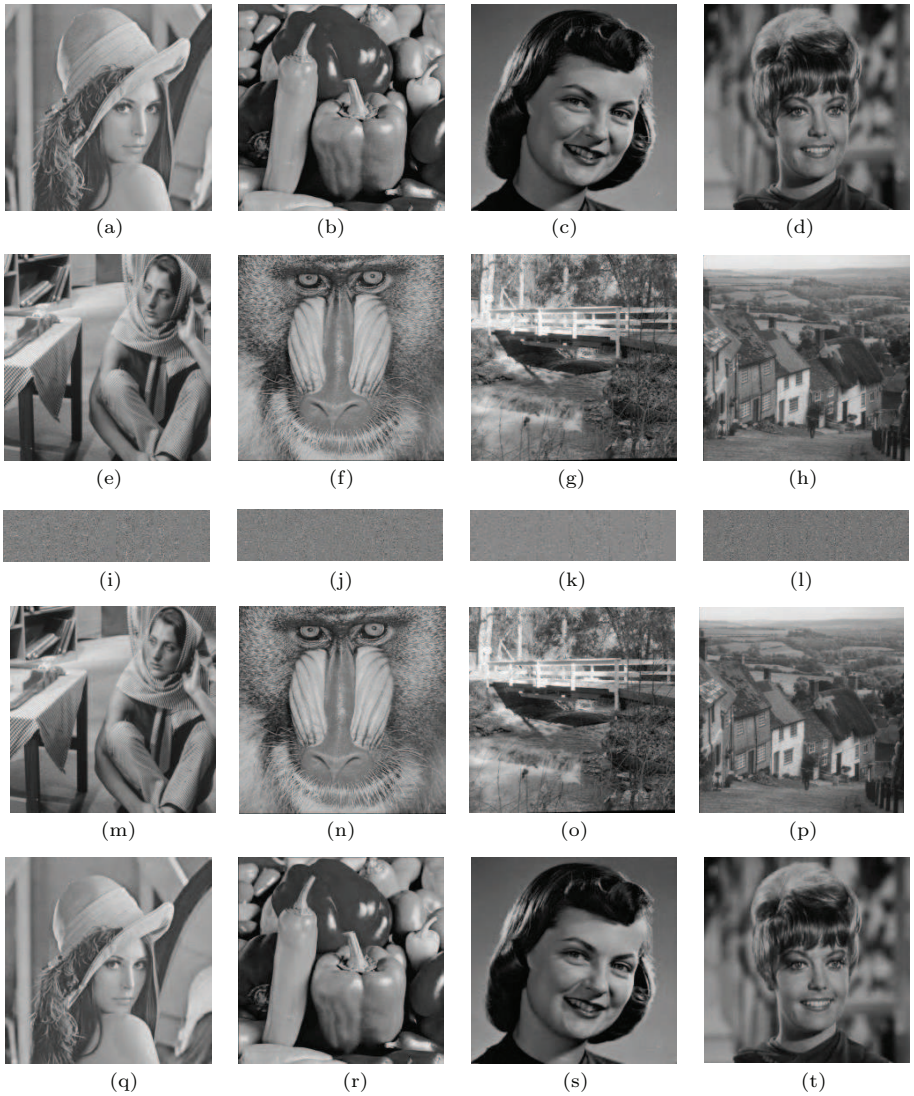


Fig. 5. Simulation results: (a)–(d) are the four plain images Lena, Peppers, Girl and Zelda; (e)–(h) are the four carrier images Barbara, Baboon, Bridge and Goldhill; (i)–(l) are the corresponding secret images, respectively; (m)–(p) are the corresponding steganographic images; (q)–(t) are the respective decrypted images.

Table 1. PSNR and MSSIM values of simulation results.

Plain image	Carrier image	$PSNR_{dec}$ (dB)	$PSNR_{ste}$ (dB)	$MSSIM_{ste}$
Lena	Barbara	36.5235	48.3918	0.9999
Zelda	Goldhill	37.9694	49.0024	0.9999
Girl	Bridge	37.0430	48.2771	0.9999
Peppers	Baboon	35.6943	48.4264	0.9999

4.2. Influence of different carrier images on simulation results

In some visual image encryption algorithm based on compressive sensing, the choice of carrier image has a great influence on the quality of decrypted image. In other words, if the carrier image is selected improperly, the quality of the decrypted image will be poor. In addition, different carrier images have a great influence on the imperceptibility of steganographic images. Therefore, the carrier image cannot be selected flexibly. By contrast, the PNG-based embedding method proposed in this paper is to embed the cipher image into the alpha channel of the PNG carrier image, which does not destroy the pixel structure of the carrier image. In order to measure the impact of different carrier images on the quality of the decrypted images in this algorithm, we encrypt the Zelda image with size of 512×512 and then embed it into five different carrier images respectively. The experimental results are shown in Fig. 6 and Table 2. It can be seen from the experimental data that different carrier images have no effect on the quality of the decrypted images. The $PSNR_{dec}$ data are above 37 dB, which indicates that the quality of the decrypted images are very good. In addition, the $PSNR_{ste}$ data are all greater than 48 dB, and the $MSSIM_{ste}$ data are approximately equal to one. It shows that the imperceptibility of steganographic images is very high, and the effect of embedding the same cipher information into different carrier images on the imperceptibility of steganographic images is very small.



Fig. 6. Simulation results for plain image Zelda with different carrier images: (a)–(e) are five carrier images Baboon, Goldhill, Couple, Barbara and Bridge; (f)–(j) are the respective steganographic images; (k)–(o) are the corresponding decrypted images.

Table 2. PSNR and MSSIM values for different carrier images.

Plain image	Carrier image	$PSNR_{dec}$ (dB)	$PSNR_{ste}$ (dB)	$MSSIM_{ste}$
Zelda	Baboon	37.9674	48.5528	0.9999
	Goldhill	37.9674	49.0042	0.9999
	Couple	37.9674	48.6097	0.9999
	Barbara	37.9674	48.5854	0.9999
	Bridge	37.9674	48.7857	0.9999

4.3. Influence of important parameters on encryption and decryption performance

In the algorithm proposed in this paper, different parameters may have different effects on encryption and decryption, such as the threshold TS. Next, we analyze the impact of the threshold TS on the quality of the decrypted image. We encrypt the plain images Girl and Woman which sizes are both 512×512, and then embed them into the carrier images Goldhill and Barbara through PNG-based embedding method. At the same time, SL_0 and OMP reconstruction methods are used to analyze the impact of threshold TS on the experimental effect, and the results are shown in Fig. 7. It can be seen from the figure that no matter which reconstruction method is adopted, the PSNR shows a trend of first increasing and then decreasing. Therefore, in order to achieve better decryption quality, the threshold TS should be set at about 20. After the threshold exceeds 20, the quality of the decrypted image will gradually decline. The threshold value has a great influence on improving the reconstruction performance of the image. In addition, it can also be seen from the figure that the SL_0 reconstruction method is generally better than OMP.

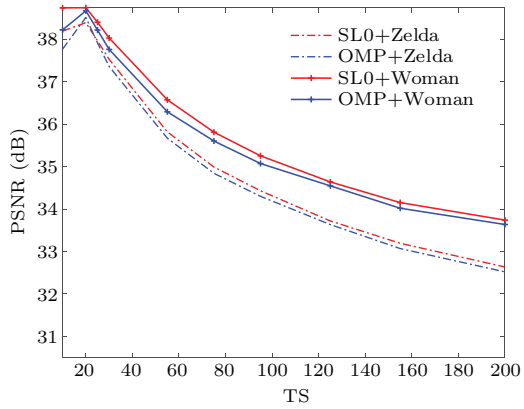


Fig. 7. PSNR vs. TS with different reconstruction methods.

Table 3. Encryption time and PSNR for different sizes of sub-blocks.

Sizes of sub-blocks	Encryption time (s)	PSNR (dB)
2×2	1.479920	36.5235
4×4	1.387481	35.7721
8×8	1.360635	35.4089

In addition, the number of sub-blocks in the dynamic spiral block scrambling algorithm will also affect the encryption and decryption effects. Taking the plain image Lena with the size of 512×512 as an example, the PSNR is used to quantitatively analyze the quality of the decrypted image. The influence of sub-blocks size on the quality of the decrypted image and encryption time is shown in Table 3. Experimental results show that the smaller the sub-block size is, the longer encryption time the algorithm needs. If the sub-block size is 2×2, the encryption time of the algorithm will be relatively long. When the sub-block size is 8×8, the decrypted image quality decreases slightly in comparison.

5. Performance analyses

In this section, we mainly analyze the performance of the proposed algorithm, including key space, plaintext sensitivity, robustness analysis and running efficiency analysis. The parameters used in the experiments are the same as those described in Subsection 4.1.

5.1. Key security analysis

The key space is an important indicator to measure whether an encryption algorithm can resist brute force attacks. Generally speaking, the key space of an encryption algorithm needs to reach 2^{100}.[42] This paper performs hash algorithm on the pixel values of plain image $\boldsymbol{P}$ to generate a 256-bit hash value, which is used to generate the initial values x, y, z, w of the hyperchaotic system. Assume that the computing accuracy of the computer is 10^{-14}. Then the total key space is greater than $(10^{14})^4 \times 2^{256} \gg 2^{100}$. In addition, there are other keys in the encryption process, such as the sub-block size N_0 and the threshold TS. Therefore, the total key space is much larger than 2^{100}, which is enough to resist various brute force attacks.

5.2. Plaintext sensitivity analysis

A good encryption algorithm should have high plaintext sensitivity, which can be used to measure whether the encryption algorithm can resist the chosen-plaintext attacks. The encryption scheme proposed in this paper uses the hash value of the plain image to generate the initial values of the hyperchaotic system, so the encryption scheme can well resist the chosen-plaintext attacks. The mean structural similarity and the number of pixel change rate (mathematical expression is shown in Eq. (29)) are introduced to quantitatively evaluate the plaintext sensitivity of the algorithm proposed in this paper.

$$\mathrm{NPCR} = \frac{1}{hw}\sum_{i=1}^{h}\sum_{j=1}^{w}\mathbf{D}_{i,j} \times 100\%, \tag{29}$$

$$\boldsymbol{D}_{i,j} = \begin{cases} 1, & \boldsymbol{C}1_{i,j} \neq \boldsymbol{C}2_{i,j}, \\ 0, & \boldsymbol{C}1_{i,j} \neq \boldsymbol{C}2_{i,j}. \end{cases} \tag{30}$$

In Eq. (30), $\boldsymbol{C}1$ and $\boldsymbol{C}2$ are two images with size of $h \times w$.

Table 4. Simulation results of the plaintext sensitivity with carrier image Goldhill.

Plain image	Coordinate	NPCR	MSSIM
Lena	(125, 333)+	99.0829	0.08822
Girl	(71, 55)+	99.0341	0.119549
Peppers	(123,323)+	99.2599	0.119932
Zelda	(376, 256)−	98.9822	0.093705

The experimental results are shown in Table 4. Four plain images with size of 512×512 and the corresponding modified plain images are encrypted, and then embed them into the steganographic image Goldhill. In the first row of Table 4, (125, 333)+ represents that the pixel values of the pixel point (125, 333) in the image Lena are added by one. As can be seen

from the experimental data, when the pixel values of the original plain image are slightly modified, the NPCRs between the two generated steganographic images are approximately equal to one, and the MSSIMs are close to zero. This shows that the algorithm proposed in this paper has very high plaintext sensitivity.

5.3. Robustness analysis

5.3.1. Noise attack

Considering that steganographic image will be affected by various noises during transmission, such as salt and pepper noise (SPN), Gaussian noise (GN) and speckle noise (SN), the quality of decrypted image will be reduced. In order to better test the ability to resist noise attacks of the proposed scheme, the plain image Girl with size of 512×512 is compressed and encrypted, and then embed it into the carrier image Bridge. SPN, GN and SN of different intensities are artificially added to the alpha channel of the PNG steganographic image. The experimental results are shown in Fig. 8. From the quality of the decrypted image, GN has the greatest effect on the encryption scheme, but the figure contour information in the decrypted image can still be recognized. Numerically, when the noise intensity is 0.00001%, 0.00003%, and 0.00005%, the PSNR values are 37.0430 dB, 35.0528 dB, and 29.3540 dB, respectively. In contrast, SPN and SN have little effects on the encryption algorithm. Under the same noise intensity, the PSNR values remain at 37.0430 dB, which further indicates that the proposed scheme can well resist SPN and SN. Conclusively, the proposed encryption scheme has a strong capability to resist SPN and SN, also it can resist GN to some extent.



(a) 0% SPN (b) 0.00001% SPN (c) 0.00003% SPN (d) 0.00005% SPN

(a) 0% GN (b) 0.00001% GN (c) 0.00003% GN (d) 0.00005% GN

(a) 0% SN (b) 0.00001% SN (c) 0.00003% SN (d) 0.00005%SN

Fig. 8. Robustness test results against noise attack for Girl.

5.3.2. Cropping attack

Steganographic images may encounter malicious cropping attacks by hackers during transmission. Therefore, the proposed encryption algorithm also needs to be robust enough. In this section, we test the ability of the proposed encryption scheme to resist cropping attacks. In the test, the steganographic images with 32×32, 64×64, 100×100 and 120×120 data losses are illustrated in Figs. 9(a)–9(d), and the corresponding decrypted images are shown in Figs. 9(e)–9(h), the PSNR values are 32.5064 dB, 31.6406 dB, 29.7184 dB, and 29.0634 dB, respectively. It can be seen that when the steganographic image has 120×120 data loss, the peak signal-to-noise ratio between the decrypted image and plain image can still reach 29 dB, and the contour information of the figure in the decrypted image can still be clearly identified. The experimental results show that the proposed image encryption has a strong ability to resist cropping attacks.

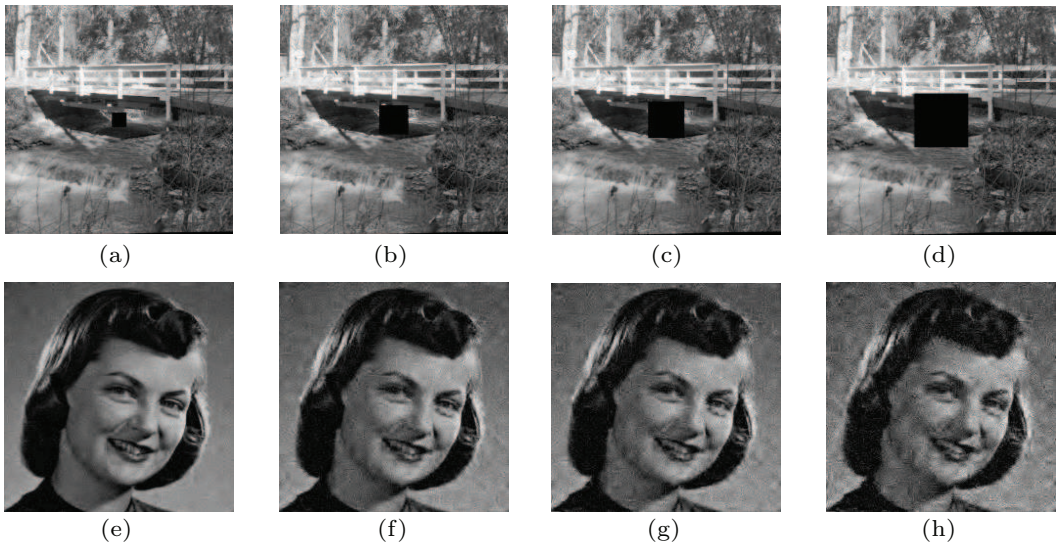


Fig. 9. Robustness test results against cropping attack for Girl: (a)–(d) steganographic images with cropped 32 ×32, 64 ×64, 100×100, and 120 ×120, respectively; (e)–(h) decrypted images of (a)–(d)

5.4. Running efficiency analysis

The running efficiency of the algorithm is also one of the indicators to measure the performance of the encryption scheme. In order to realize the real-time image processing, it is necessary to improve the efficiency of the algorithm as much as possible. In this subsection, we analyze the encryption time of the proposed algorithm, and select the plain image Lena with the size of 512×512 and 256×256 for experiment. When the sub-block sizes of dynamic spiral block scrambling are 2×2 and 4×4, the experimental results are shown in Table 5 and Table 6, respectively. Here "encryption" represents the encryption and compression process, and "embedding" represents embedding the cipher image into the carrier image. It can be concluded from the experimental data that, firstly, the embedding process takes up most of the time of the whole encryption process. However, for Lena with size of 512×512, PNG-based embedding process only takes 0.070259 s, and most of the time is spent on LSB replacement. Secondly, the sub-block size of the dynamic spiral block scrambling has a great impact on the Encryption process time. It can be seen from the comparison that if we want to improve the encryption efficiency of the algorithm, the larger sub-blocks should be selected.

Table 5. Encryption time for different images via a 2×2 sub-block (unit: s).

Image size	Encryption	Embedding	Total
256×256	0.120960	0.337443	0.458403
512×512	0.161777	1.318143	1.479920

Table 6. Encryption time for different images via a 4×4 sub-block (unit: s).

Image size	Encryption	Embedding	Total
256×256	0.035319	0.337443	0.372762
512×512	0.069338	1.318143	1.387481

5.5. Chosen-plaintext attack analysis

UP to now, many encryption algorithms have been attacked by chosen-plaintext attacks. The main reason is that different plain images correspond to the same key stream in encryption. However, in this paper, first the average value and hash value of plain image are calculated and used to generate the initial values of the chaotic system. Then, chaotic sequences generated by hyperchaotic Lorenz system are used for dynamic spiral block scrambling, measurement matrix generation and PNG-based embedding. In this way, different encryption keys are generated for different images, thus improving the ability to resist plaintext attack.

To better prove the proposed algorithm so that it can resist the chosen-plaintext attack, we also carry out experimental verification. This attack can be portrayed as[23]

$$\boldsymbol{O}_1(i,j) \oplus \boldsymbol{O}_2(i,j) \neq \boldsymbol{C}_1(i,j) \oplus \boldsymbol{C}_2(i,j), \tag{31}$$

where $\boldsymbol{O}_1$ and $\boldsymbol{O}_2$ are plain images, $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ are their corresponding cipher images. If this equation is applied to any i and j, it means that the test method can resist the chosen-plaintext attack.

To intuitively observe more test results, the plaintext attack of three pairs of gray image pairs is tested. In Fig. 10, the first and second columns are plain images, the fourth and fifth columns are their corresponding secret images, and the third and sixth columns represent the XOR of two plain images and the XOR of two secret images, respectively. It is evident that the equation is satisfied, so the algorithm has strong resistance towards the chosen plaintext analysis. In addition, we first perform the XOR operation on two plain images, and then perform the XOR operation on their cipher images. We compare the two images obtained after XOR, calculate the number of different pixels between them, and the proportion of these different pixels in the total pixel number of the cipher image. The statistical results are shown in Table 7. From the above discussion, it is clear that the hacker cannot obtain the key by applying a chosen plaintext attack.

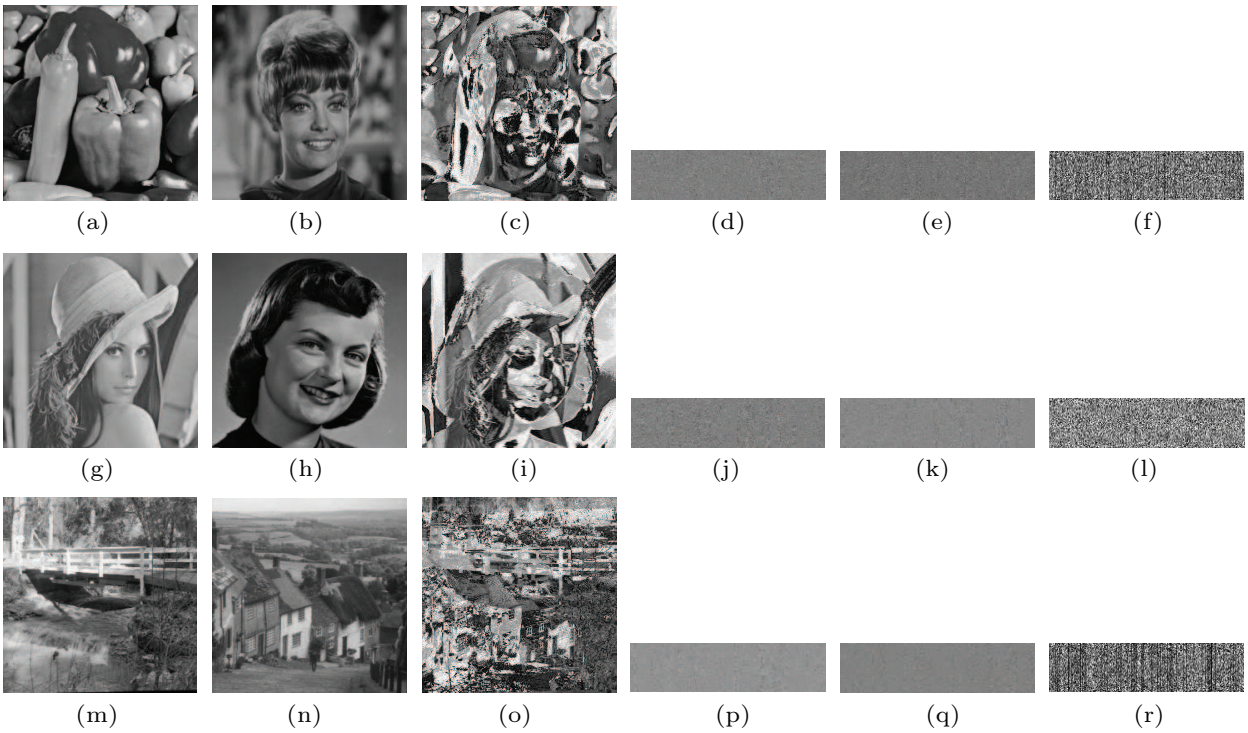


Fig. 10. Image-by-pixel XOR operation of two images: (a) plain image Peppers, (b) plain image Zelda, (c) (a)⊕(b), (d) secret image Peppers, (e) secret image Zelda, (f) (d)⊕(e); (g) plain image Lena, (h) plain image Girl, (i) (g)⊕(h), (j) secret image Lena, (k) secret image Girl, (l) (j)⊕(k); (m) plain image Bridge, (n) plain image Goldhill, (o) (m)⊕(n), (p) secret image Bridge, (q) secret image Goldhill, (r) (p)⊕(q).

Table 7. Statistical analysis of chosen-plaintext attack.

O_1	O_2	Number of different pixels	Percentage (%)
Peppers	Zelda	65303	99.64%
Lena	Girl	65263	99.58%
Bridge	Goldhill	65257	99.57%

6. Comparison with related schemes

In this section, we compare the image encryption algorithm proposed in this paper with existing related algorithms, mainly from the four aspects of visual security, decrypted image quality, robustness, and encryption efficiency. The experimental data are given in the following. In order to make a fair comparison, the experimental data to be compared is directly quoted from the source paper or obtained through simulation experiments.

6.1. Visual security

For visual image encryption algorithms, the imperceptibility of steganographic images is an important indicator to measure its security. The higher the imperceptibility of the steganographic image is, the greater the similarity between the steganographic image and the carrier image is, then the higher the visual security of the encryption algorithm is. In order to quantitatively measure the imperceptibility of steganographic images, we compare the $\mathrm{PSNR_{ste}}$ and $\mathrm{MSSIM_{ste}}$ with the experimental results of Refs. [26,27]. The results are shown in Table 8.

It can be seen from the comparison data that the average $\mathrm{PSNR_{ste}}$ value proposed in this paper is as high as 48.6207 dB, while the average $\mathrm{PSNR_{ste}}$ value of Ref. [26] is only 23.8436 dB. The proposed encryption scheme can also improve the average $\mathrm{PSNR_{ste}}$ value of Ref. [27] by 37.80%. At the same time, the average $\mathrm{MSSIM_{ste}}$ value of the algorithm proposed in this paper has also reached 0.9999, which is much larger than the $\mathrm{MSSIM_{ste}}$ value of Refs. [26,27]. In conclusion, the comparison results fully confirm that the algorithm proposed in this paper improves the imperceptibility of steganographic images and has a high degree of visual security.

Table 8. Comparison of PSNR and MSSIM values of steganographic images.

Plain image (carrier image)	$\mathrm{PSNR_{ste}}$ (dB)			$\mathrm{MSSIM_{ste}}$		
	Ref. [26]	Ref. [27]	Present	Ref. [26]	Ref. [27]	Present
Lena (Peppers)	18.5136	32.3513	49.0066	0.6726	0.9257	0.9999
Jet (Baboon)	23.3967	37.1058	48.4655	0.6991	0.9833	0.9999
Girl (Goldhill)	28.2318	36.1125	48.2414	0.7021	0.9666	0.9999
Barbara (Bridge)	25.2321	35.5629	48.7694	0.7337	0.9783	0.9999
Average	**23.8436**	**35.2831**	**48.6207**	**0.7019**	**0.9635**	**0.9999**

6.2. Decrypted image quality

The embedding process and the corresponding extraction process of the encryption algorithm proposed in this paper are completely reversible. $PSNR_{dec}$ is used to quantitatively compare the quality of the decrypted images, and the experimental data obtained are shown in Table 9. It can be seen from the comparison results that the decrypted image quality of the proposed encryption scheme is much higher than the other, which further confirms that the decrypted image quality of the proposed encryption scheme is very high.

Table 9. PSNR values for comparison of different decrypted images.

Plain image (carrier image)	$PSNR_{dec}$ (dB)		
	Ref. [26]	Ref. [27]	Present
Lena (Peppers)	30.0862	33.4204	36.5235
Jet (Baboon)	28.9067	31.4662	33.4416
Girl (Goldhill)	29.3820	34.2626	37.0430
Barbara (Bridge)	25.7491	28.4422	31.1695
Average	28.5310	31.8979	34.5444

6.3. Robustness comparison

In this subsection, we compare the robustness of the proposed algorithm with the algorithm based on IWT+LSB embedding[27] and Dynamic LSB embedding.[43] The experimental data comes from Ref. [28]. The plain image Girl is encrypted and embedded into the carrier image Goldhill by different encryption schemes. Then add SPN, GN, SN attack and cropping attack of different intensities to the steganographic image. The experimental data obtained are listed in Tables 10 and 11, where I-LSB and D-LSB separately are obtained using the IWT+LSB and dynamic LSB embedding methods.

From Table 10, we can see that no matter which encryption algorithm is used, GN has the greatest impact on the cryptosystem. However, compared with the algorithms of Refs. [27,43], the encryption algorithm proposed in this paper can better resist GN attack. The PSNR values are enhanced approximately 13–17 dB at the same intensity. All the encryption algorithms have strong ability to resist SPN attack. All three encryption algorithms can resist SN attack to a certain extent. However, when the intensity increases, the algorithm proposed in this paper has better decryption quality. In addition, as can be seen from Table 11, the data loss increases from 16×16 to 80×80, and the quality of the decrypted image generated by our scheme drops by about 5.3 dB, which is less than those in Refs. [27,43]. When the cropping mask becomes larger, the encryption algorithm proposed in this paper provides performance gain about 11 dB. To sum up, the proposed image encryption algorithm based on the PNG embedding method can effectively improve the robustness against noise attack and cropping attack.

Table 10. Comparison of resistance capability of noise attack.

Noise type	Attack intensity	$PSNR_{dec}$(dB)		
		I-LSB	D-LSB	Present
SPN	0.0001%	33.44	33.44	37.0430
	0.0003%	33.26	33.44	37.0413
	0.0005%	33.02	33.44	37.0408
GN	0.0001%	15.32	14.75	28.0919
	0.0003%	10.19	9.35	27.5209
	0.0005%	9.35	8.54	27.3775
SN	0.0001%	33.44	33.44	37.0430
	0.0003%	19.10	22.50	27.4399
	0.0005%	15.44	17.30	27.3960

Table 11. Comparison of resistance capability of cropping attack.

Attack intensity	$PSNR_{dec}$(dB)		
	I-LSB	D-LSB	Present
16×16	28.98	28.95	35.0733
32×32	24.92	24.58	32.5064
64×64	20.13	19.73	31.6406
80×80	19.62	18.01	29.7184

6.4. Encryption efficiency

To encrypt the plain image in real time, the encryption time should be as short as possible, the encryption time should be as short as possible. Comparing the encryption efficiency of the algorithm proposed in this paper with Refs. [27,43], the results are given in Table 12.

Table 12. The encryption efficiency comparison with other algorithms (unit: s).

Item	Lena (512×512)			Lena (256×256)		
	Ref. [27]	Ref. [43]	Present	Ref. [27]	Ref. [43]	Present
Encryption	0.116	0.53	0.1617	0.071	0.10	0.1209
Embedding	0.193	2.00	1.3181	0.083	0.24	0.3374
Total	0.309	2.53	1.4798	0.154	0.35	0.4583

The sub-block size of scrambling algorithm is 2×2. As can be seen from the table, when the size of the plain image is 256×256, the encryption time of all algorithms is very short. However, when the size of the plain image is 512×512, the encryption time of Ref. [43] increases dramatically. To sum up, the encryption efficiency of the proposed algorithm is second only to Ref. [27], while the encryption efficiency of the algorithm proposed in Ref. [43] is the lowest.

7. Conclusions

This paper proposes a novel visually meaningful image encryption algorithm combining hyperchaotic system, compressive sensing and PNG-based embedding. In our encryp-

tion scheme, dynamic spiral block scrambling and compressive sensing are used to encrypt and compress the plain image. Then the cipher image is embedded into the alpha channel of the PNG carrier image. Different from the existing visual image encryption algorithms, the PNG-based embedding method does not embed cipher image into the pixel, so it does not destroy the pixel structure of the carrier image. The embedding process and the corresponding extraction process are completely reversible, which can avoid truncation errors during embedding and improve the quality of decrypted image. In addition, the proposed decryption algorithm realizes blind extraction, which effectively reduces transmission cost and storage space. Meanwhile, the hash value of the plain image is used to generate the keys, which is beneficial to improve the ability of the proposed algorithm to resist chosen-plaintext attacks. Finally, the experimental results show that the proposed image encryption algorithm has very good visual security, decryption quality and robustness.

It is worth noting that, although existing visual image encryption schemes have good security performance, their embedding positions are usually fixed or controlled by keys. If the embedding position is not appropriate, the imperceptibility of steganographic image may be affected. Therefore, in the future work, we will devote ourselves to studying an adaptive embedding scheme to further improve the security of the algorithm.

Acknowledgements

Project supported by the National Natural Science Foundation of China (Grant No. 61672124), the Password Theory Project of the 13th Five-Year Plan National Cryptography Development Fund (Grant No. MMJJ20170203), Liaoning Province Science and Technology Innovation Leading Talents Program Project (Grant No. XLYC1802013), Key R&D Projects of Liaoning Province (Grant No. 2019020105-JH2/103), Jinan City '20 Universities' Funding Projects Introducing Innovation Team Program (Grant No. 2019GXRC031), and Research Fund of Guangxi Key Lab of Multi-source Information Mining & Security (Grant No. MIMS20-M-02).

References

[1] Wang X Y and Liu P B 2022 *IEEE Trans. Circuit. Syst.* **69** 1291
[2] Chen J J, Yan D W, Duan S K and Wang L D 2020 *Chin. Phys. B* **29** 110504
[3] Yan X P, Wang X Y and Xian Y J 2022 *Chin. Phys. B* **31** 080504
[4] Zhou N R, Yan X Y, Liang H R, Tao X Y and Li Y G 2018 *Quantum Inf. Process.* **17** 338
[5] Yang Y G, Tian J, Lei H, Zhou Y H and Shi W M 2016 *Inf. Sci.* **345** 257
[6] Fang P F, Liu H, Wu C M and Liu M 2022 *Chin. Phys. B* **31** 040501
[7] Wang R, Li M Y and Luo H J 2022 *Chin. Phys. B* **31** 080508
[8] Liu S T and Zhang L 2021 *Surface Chaos and Its Applications* (Singapore: Springer) pp. 30–32
[9] Xian Y J, Wang X Y and Teng L 2022 *IEEE Trans. Circuit. Syst. Vid.* **32** 4028
[10] Xian Y J, Wang X Y, Wang X Y, Li Q and Yan X P 2022 *IEEE Trans. Circuit. Syst.* **69** 3320
[11] Xian Y J, Wang X Y, Zhang Y Q, Wang X Y and Du X H 2021 *Chin. Phys. B* **30** 060508
[12] Wen W Y, Wei K K, Zhang Y S, Fang Y M and Li M 2020 *Nonlinear Dyn.* **99** 1587
[13] Chen J X and Zhou Y C 2020 *Inf. Sci.* **520** 130
[14] Liu S T and Wang P 2018 *Fractal Control Theory* (Singapore: Springer Nature) pp. 97–111
[15] Liu S T, Zhang Y P and Liu C A 2020 *Fractal Control and Its Applications* (Switzerland: Springer Nature) pp. 61–80
[16] Liu S T, Wang Y P, Bi Z M and Wang Y 2021 *Mathematical Principle and Fractal Analysis of Mesoscale Eddy* (Singapore: Springer) pp. 121–125
[17] Li Q, Wang X Y, Ma B, Wang X Y, Wang C P, Gao S and Shi Y Q 2022 *IEEE Trans. Circuit. Syst. Vid.* **32** 5695
[18] Shi H and Wang L D 2019 *Acta Phys. Sin.* **68** 200501 (in Chinese)
[19] Wang X G, Li M, Yu N N, Xi S X, Wang X L and Lang L Y 2019 *Acta Phys. Sin.* **68** 240503 (in Chinese)
[20] Kanso A and Ghebleh M 2017 *Opt. Laser Eng.* **90** 196
[21] Kumar V and Kumar D 2017 *Multimed. Tools Appl.* **77** 13279
[22] Wang X Y, Liu C and Jiang D H 2021 *Inf. Sci.* **574** 505
[23] Wang X Y, Liu C and Jiang D H 2022 *Inf. Sci.* **610** 300
[24] Armijo-Correa J O, Murguia J S, Mejia-Carlos M, Arce-Guevara V E and Aboytes-Gonzalez J A 2020 *Opt. Laser Technol.* **127** 106165
[25] Bao L and Zhou Y C 2015 *Inf. Sci.* **324** 197
[26] Chai X L, Gan Z H, Chen Y R and Zhang Y S 2017 *Signal Process.* **134** 35
[27] Wang H, Xiao D, Li M, Xiang Y P and Li X Y 2019 *Signal Process.* **155** 218
[28] Zhu L Y, Song H S, Zhang X, Yan M D, Zhang T, Wang X Y and Xu J 2020 *Signal Process.* **175** 107629
[29] Hussain M, Wahab A W A, Ho A T S, Javed N and Jung K H 2017 *Signal Process.-Image* **50** 44
[30] Tai W L, Yeh C M and Chang C C 2009 *IEEE Trans. Circuit. Syst. Vid.* **19** 906
[31] Ping P, Fu J, Mao Y C, Xu F and Gao J 2019 *IEEE Access* **7** 170168
[32] Ye G D, Pan C, Dong Y X, Jiao K X and Huang X L 2021 *Trans. Emerg. Telecommun. Technol.* **32** e4071
[33] Miri A and Faez K 2017 *Multimed. Tools Appl.* **77** 13133
[34] Malik A, Wang X H, Chen T L, Yang T L, Khan A N, Wu H Z, Chen Y L and Hu Y 2019 *J. Inf. Secur. Appl.* **48** 102374
[35] Wen W Y, Hong Y K, Feng Y M, Li M and Li M 2020 *Signal Process.* **173** 107580
[36] Yang Y G, Zou L, Zhou Y H and Shi W M 2020 *Optik* **213** 164422
[37] Wang X Y and Wang M J 2008 *Physica A* **387** 3751
[38] Candes E J, Romberg J and Tao T 2006 *IEEE Trans. Inf. Theory* **52** 489
[39] Donoho D L 2006 *IEEE Trans. Inf. Theory* **52** 1289
[40] Musanna F, Dangwal D and Kumar S 2020 *Multimed. Tools Appl.* **79** 25115
[41] Wang Z, Bovik C, Sheikh H R and Simoncelli E P 2004 *IEEE Trans. Image Process.* **13** 600
[42] Alarez G and Li S J 2006 *Int. J. Bifur. Chaos* **16** 2129
[43] Chai X L, Wu H Y, Gan Z H, Zhang Y S, Chen Y R and Nixon K W 2020 *Opt. Laser Eng.* **124** 105837