Visually secure image encryption using adaptive sparsification and parallel compressive sensing

感觉abstract第一句话应该是visually secure image encryption开头，基于CS的方法应该放到后面

Zhang Kuiyuan

CS has recently been shown as an effective technique for visually secure image encryption, where the images are concurrently encrypted and compressed.

Specifically, we show that the proposed adaptive sparsification can ..

In this work, we propose

in terms of

这里应该缩写CS？

significantly

Abstract

Compressive sensing (CS) is a suitable technique for image encryption and compression. Recently, some visually secure image encryption schemes have been developed using CS to concurrently encrypt and compress images. In these schemes, a plain-image is first encrypted and compressed, and then embedded into a carrier image. Because the cipher image is visually meaningful, the plain-image can be dually protected. However, existing ~~visually secure image encryption~~ schemes have many weaknesses in the quality of reconstructed and cipher images and efficiency. ~~To solve these weaknesses, this study~~ proposes a new visually secure image encryption scheme using adaptive sparsification and parallel compressive sensing. The adaptive sparsification can greatly improve the quality of reconstructed image by utilizing the separable wavelet transform and column-based adaptive thresholding. The parallel CS with random-order measurement matrices is adapted to enhance the processing efficiency. Besides, matrix encoding technique is employed to embedding the secret image into carrier image, which can greatly reduce the number of changed bits. Simulation results and security analysis show that our proposed scheme can encrypt plain-images to be cipher-images with a high security level. Comparison results demonstrate that it can achieve a higher efficiency, higher quality of reconstructed image and less data loss of carrier image than some newly developed schemes.

the

Keywords: Image encryption; Image compression; Separable wavelet transform; Parallel compressive sensing; Matrix encoding.

Further, a matrix ..is

be是多余的？

also

adopted

,while the proposed parallel CS ,,,, can enhance ..tremendously

a higher efficiency有点问题

1. Introduction

As a typical kind of multimedia data, the digital images are widely used to deliver information every moment. Because a digital image can contain many potential information, it may cause a serious information security incident when some secret images, e.g. military images, are acquired by ~~some~~ unauthorized accesses. Therefore, it is very important to protect the contents of secret images.

Among all the technologies of protecting images, the encryption is the most straightforward and effective one. Many image encryption algorithms have been developed and these algorithms can be divided into three kinds. The first kind only encrypts a plain-image to be an unrecognizable image with same size using different techniques including chaos theory [1, 2, 3, 4], DNA coding [5, 6, 7], quantum transformation [8, 9], cellular automata [10, 11], domain transformation [12, 13] and etc. When encrypting a plain-image to be an unrecognisable cipher-image, the cipher-image is random-like and independent from the plain-image. Only with the correct secret key, one can recover the

plain-image. Without the correct secret key, one cannot obtain any useful information from the cipher-image. Thus, these encryption algorithms can achieve a high security level. However, this kind of algorithms have some obvious weaknesses. First, the encryption structures of some algorithms have low security levels and thus the encrypted results can be broken [14, 15, 16, 17]. Besides, the cipher-images in these encryption algorithms usually have the same sizes with the plain-images and this leads to low encryption efficiency. Because many natural images have high data redundancy, it is meaningful to reduce the data redundancy when encrypting them.

The second kind of encryption algorithms encrypt a plain-image to be an unrecognizable image with reduced size. To concurrently perform image compression and encryption, many researchers have introduced the compressive sensing (CS) technology into the image encryption. For example, the authors in [18, 19] firstly compressed a plain-image using the CS, and then encrypted the compressed image by scrambling and diffusing image pixels. To improve the security level, the authors in [20] further proposed a parallel CS technique to resist chosen plain attack. These encryption algorithms can concurrently compress and encrypt a plain-image and th in many scenes. However, they also transform plain images to be unrecognisable cipher imag images easy to attract the attentions of attackers.

To overcome the weaknesses of the previous two kinds of encryption algorithms, the third kind of encryption algorithms aim to encrypt a plain image to be a cipher image with visual security [21, 22]. These encryption algorithms usually include two stages: encryption stage and embedding stage. The encryption stage encrypts a plain image to be a secret image, while the embedding stage embeds the secret image into a carrier image to generate the final visually meaningful cipher image. For example, the scheme in [23] first encrypts a plain image to be a unrecognisable secret image, and then embeds the secret image into a carrier image by replacing the partial of the carrier image. To reduce the embedding size, Chai *et al*. proposed an encryption scheme using CS and discrete wavelt transform (DWT) [24]. In this encryption scheme, a plain image is first compressed using CS, and then encrypted to be a secret image, and finally embedded into a carrier image that has the same size with the plain image. To improve the security level and quality of reconstructed image, the authors in [25] modified the encryption structure in [24] by introducing parallel CS counter mode and integer wavelet transformation. Although these existing visually secure image encryption schemes can achieve a relatively high performance, they still have many performance limitati of plain-image, quality of reconstructed image and processing efficiency.

In order to overcome the performance limitations of existing image encryption algorithms, this paper proposes a new visually secure image encryption scheme using adaptive sparsification and parallel compressive sensing (PCS). First, a plain-image is decomposed by separable wavelet transform (SWT) [26] and scrambled by the 2D cat map [27]. Second, the scrambled image is sampled using PCS with a threshold for each column, and the measurement matrices for each column are generated by a chaotic system. Finally, after quantifying and diffusing, the secret image is embedded to a carrier image using matrix encoding. The contributions and novelty of this paper can be summarized as follows.

(1) To improve the sparsity performance, we propose the adaptive sparsification strategy, which utilizes SWT and column-based adaptive thresholding.

(2) To improve the efficiency, we generate a random-order measurement matrix for each column in the data sampling.

(3) To reduce the data loss of carrier image, we introduce the matrix coding technique to embed the secret image to the carrier image.

(4) Comparison results demonstrate that our proposed scheme can achieve a higher efficiency, higher quality of reconstructed image and less data loss of carrier images than some newly developed schemes.

The rest of this paper is organized as follows. Section 2 introduces some related works and discusses their properties. Section 3 presents the proposed visually secure encryption scheme. Section 4 simulates the proposed scheme and analyzes its performance. Section 5 evaluates the security of the proposed scheme and compares it with some recently developed schemes. Section 6 gives a conclusion of this paper.

2. The Related work and Preliminaries

This section first introduces the CS theory, and then analyzes some representative visually secure encryption schemes using CS, and finally presents some techniques that are used to design new encryption scheme in Section 3.

2.1. CS Theory

The CS theory specifies that when sampling a sparse signal below the Nyquist rate, the original signal can be completely recovered from the sampled data [28, 29]. It tells that a sparse signal can be represented by a small data set that is far smaller than it. Suppose a sparse signal $\mathbf{x}$ is of size $N \times 1$ and a measurement matrix $\mathbf{\Phi}$ is of size $M \times N$ ($M \ll N$), the sample process can be defined as

another data?

$$\mathbf{y} = \mathbf{\Phi} \cdot \mathbf{x}, \tag{1}$$

where $\mathbf{y}$ is the measurements vector with size $M \times 1$. The sparse signal $\mathbf{x}$ is called K-sparse when it has K non-zero entries. To completely recovery of original signal, the K, M and N should satisfy that

$$M \geq cK \log_2(N/K), \tag{2}$$

solvig the following optimization problem

where c is constant number.

When recovering the original signal, the estimation of $\mathbf{x}$, denoted as $\hat{\mathbf{x}}$, can be calculated by

$$\min \|\hat{\mathbf{x}}\|_p \quad \text{s. t.} \quad \mathbf{y} = \mathbf{\Phi} \cdot \hat{\mathbf{x}} \tag{3}$$

where $p = 0$ or 1, and $\|\cdot\|_p$ denotes the l_0-norm or l_1-norm. Many effective reconstruction methods have been approved, which includes basic pursuit, orthogonal matching pursuit (OMP) [30] and smoothed l_0 norm (SL0) [31].

有的公式后面有标点

Usually, a natural signal is not sparse. Thus, to deal with a natural signal using CS theory, one should first transform the signal into a frequency domain to get its sparse representation and then perform the CS to the obtained sparse signal. Thus, for a natural signal $\mathbf{x}$, the sample process can be defined as

$$\mathbf{y} = \mathbf{\Phi} \cdot \mathbf{s} = \mathbf{\Phi} \cdot \mathbf{\Psi} \cdot \mathbf{x}, \tag{4}$$

where $\mathbf{\Psi}$ is a sparse transformation matrix, $\mathbf{s}$ is the sparse representation of signal $\mathbf{x}$.

2.2. *Visually Secure Image Encryption Using CS*

第一句话好像没有铺垫，不知道这个claim怎么来的

Because the CS can concurrently compress and encrypt an digital image, it is widely used in image security algorithms. Recently, many visually secure image encryption schemes have been developed using CS [24, 25, 32]. These algorithms first encrypts a plain image to be a secret image with reduced size, and then embeds the secret image into a carrier image to generate a cipher image. Because the cipher-image has the same visual effect with the carrier image, the plain image can be dually protected. ~~Generally speaking~~, a general framework of these algorithms can be plotted as Fig. 1. It shows that the procedure can be divided into three stages: sparsification, compressive sampling and embedding. The sparsification transforms a plain-image to be a spare signal. The compressive sampling processes the spare signal using the CS theory and the embedding hides the secret image to a carrier image.

a similar visual?

where the XX is divided into three stages:

sparse?

Since

is presented in

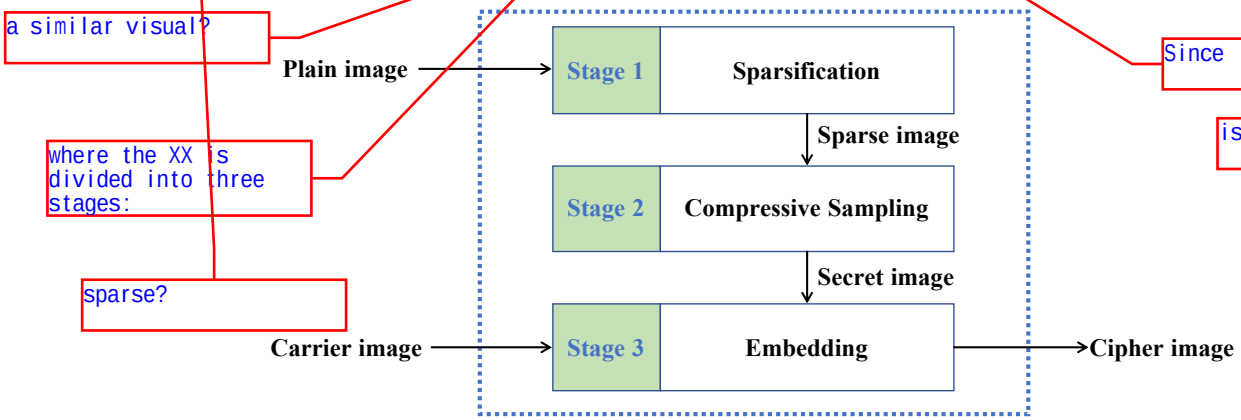


Figure 1: A general framework of the visually secure image encryption schemes using CS.

2.2.1. *Sparsification*

A natural image should first be transformed to be a sparse signal before sampled via CS theory. One of the most effective method is to transform the image from spatial domain to frequency domain using some techniques such as the wavelet transform and Fourier transform. Since a digital image has much data redundancy, its frequency spectrum has many elements that are close to 0. After setting these elements to 0 via quantifying with a threshold, a sparse signal can be generated.

It is obvious that the quantization process is ~~a~~ lossy ~~operation~~. Thus, the quality of reconstructed result is directly affected by the sparsification strategy. For example, the authors in [24, 25, 32] first employ the discrete wavelet

highly

transform (DWT) to transform an image into the frequency domain, and then shuffle the image data positions using some techniques, and finally use a global threshold to quantify the image data. Although these existing sparsification strategies can achieve a good result, they still have some weaknesses. The sparsification performance using DWT is not very high, and the quantization using a global threshold can produce different sparsity in every column of shuffled image. These two weak points will result in low quality of reconstructed image and limits the compression ratio.

具体哪方面

shortcomings, disadvantages

2.2.2. *Compressive Sampling*

In this stage, the CS theory is used to compressively sample the sparse image to reduce the image size. According to the CS theory in Eq. (1), a measurement matrix is required when compressively sampling a sparse image. To solve the low security level of the secret image caused by a fixed measurement matrix for the whole sparse image [20], researchers usually use changeable measurement matrices generated by a chaotic system. Specifically, the sampling is performed column-by-column or block-by-block. Generally speaking, the column-by-column sampling can result in a better quality of constructed image but a slower reconstruction speed than the block-by-block sampling [33].

Recently, to further improve the security level of the secret image, researchers developed more effective techniques to generate the measurement matrix. For example, the authors in [25] generated different measurement matrices to sample every column of the sparse image. Even using the same secret key, the produced measurement matrices are different. The authors in [32] first generated different structurally random matrices for every column, and then combined a chaotic sequence with a random number to determine which measurement matrix is used. However, the processing speed is greatly slowed down when generating different measurement matrices for each column of the image ~~data~~.

degraded

如果是修饰matrix 应该是structural

is then

在讲别人文章的时候，好像一会是过去式，一会是一般式

好像不是很准确， 前面也不是这样表述的

2.2.3. *Embedding*

After compressive sampling, a secret image is generated ~~and it then is~~ embedded into a carrier image to obtain visually secure cipher image. The cipher image is similar to the carrier image and thus can eliminate the attentions of attackers. To get a better visual effect, the carrier image has several times bigger size than the secret image. Usually, a pre-processing is performed to the carrier image to further improve the embedding space or reduce the data loss. For example, the authors in [24] decomposed the carrier image into several coefficient matrices using DWT and embedded the secret image using pixel substitution. Finally, a modified carrier image is generated by applying the inverse DWT to the modified coefficient matrices. However, this strategy can cause low quality of reconstructed image, because the original DWT is defined in the fractional domain and doss loss will happen when converting fractional number to integer number. To solve this problem, the authors in [25] uses the integer wavelet transform (IWT) to instead the DWT. Since the IWT is an inversive operation and thus the selection of carrier image has no influence on the quality of the constructed image. Besides, to further improve the embedding effect, the authors in [32] first divided the secret and the carrier images into the same number of non-overlapping blocks, and then hides each block of secret image in a block of carrier image. These strategies can significantly reduce the the data loss of the carrier image and obtain a

good visual effect. However, they also result in a low-quality and unstable cipher image, because the pixels of carrier image are directly replaced by the pixels of secret image.

2.3. Our Contributions

To further improve the performance of the visually secure image encryption, we introduce some new techniques to improve the quality of cipher and reconstructed images and processing efficiency.

2.3.1. Adaptive Sparsification

To improve the sparsity performance, an adaptive sparsification is introduced and it includes three operations: SWT, matrix confusion, and column-based adaptive thresholding (CBAT). The SWT is used to decompose the plain image to get its sparse representation in wavelet domain. First, the orthogonal wavelet transform matrix is calculated as

$$\mathbf{W} = \mathbf{P}_n\mathbf{P}_{n-1}\ldots\mathbf{P}_2\mathbf{P}_1, \tag{5}$$

where the $\mathbf{P}_i$ $(i = 1, 2, \ldots, n)$ with size $N \times N$ is calculated as

$$\mathbf{P}_i = \begin{bmatrix} \begin{bmatrix} H_{(N/2^i)\times(N/2^{i-1})} \\ G_{(N/2^i)\times(N/2^{i-1})} \end{bmatrix} & 0 \\ 0 & \mathbf{I} \end{bmatrix}, \tag{6}$$

where H and G are the low and high pass filters of the wavelet base, respectively, the maximum value of i is $\lfloor \log_2(N) \rfloor$ ($\lfloor \cdot \rfloor$ is to obtain the smallest integer that isnot smaller than $\cdot$) and $\mathbf{I}$ is an identity matrix. The DWT has high processing efficiency but low sparsity performance, while the wavelet packet transform (WPT) has high sparsity performance but low processing efficiency. The SWT can well balance the trade-off between the performance and efficiency. It has the same computation complexity and much better sparsity performance than the DWT.

After the image decomposition by SWT, the elements in the coefficient matrix can be divided into principal components and secondary components. The matrix confusion randomly shuffle the elements in the coefficient matrix such that the principal components can be uniformly distributed in each column.

Finally, the CBAT method is proposed to adaptively set thresholds for different images. The CBAT ensures that every column of an image has the same sparsity and can be almost completely reconstructed. For a compression rate CR and column length N, the sparsity for each column is calculated as

$$K = \lfloor \theta \cdot \omega \cdot CR^{\omega} \cdot e^{-\theta \cdot CR^{\omega}} \cdot N \rfloor, \tag{7}$$

where θ and ω are determined by the reconstruction method. Then the threshold for a column is the K-th largest absolute value in the column. The elements whose absolute values are smaller than the threshold will be set to zero.

2.3.2. Random-Order Measurement Matrix

A random-order measurement matrix is generated to improve the generation efficiency of measurement matrix. Because the parallel compressive sensing (PCS) can concurrently process all the columns of an image, a number of N measurement matrices should be generated when the image has N columns.

Since the 2D Logistic-adjusted-Sine map (2D-LASM) chaotic map introduced in [34] can generate chaotic signals with uniformly distribution, it is used to generate the N measurement matrices. Its equations are defined as

$$\begin{cases} x_{i+1} = \sin(\pi\mu(y_i + 3)x_i(1 - x_i)), \\ y_{i+1} = \sin(\pi\mu(x_{i+1} + 3)y_i(1 - y_i), \end{cases} \tag{8}$$

where the parameter $\mu \in [0, 1]$ and (x_0, y_0) are initial values. To generate N measurement matrices of size $M \times N$, a chaotic matrix $\mathbf{S}$ with size $M \times 2N$ is firstly generated by iterating the 2D-LASM with given initial state (x_0, y_0, μ). Then, each of the N measurement matrices can be generated by selecting N columns from $\mathbf{S}$ according to the chaotic sequences. Finally, the elements in all the measurement matrices are converted into 1 or -1 by

$$\mathbf{\Phi}_i(r, c) = \begin{cases} 1, & \mathbf{\Phi}_i(r, c) \geq 0.5; \\ -1, & \mathbf{\Phi}_i(r, c) < 0.5. \end{cases} \tag{9}$$

Algorithm 1 shows the generation of N measurement matrices $\mathbf{\Phi}_1, \mathbf{\Phi}_2, \cdots, \mathbf{\Phi}_N$.

Algorithm 1: Generation of N measurement matrices.

Input: The initial state (x_0, y_0, μ) and image size $M \times N$.

Output: The N measurement matrices $\mathbf{\Phi}_1, \mathbf{\Phi}_2, \cdots, \mathbf{\Phi}_N$.

Generate a chaotic sequence $\mathbf{X} = \{x_1, x_2, \ldots, x_{2MN+2NN}\}$ using the 2D-LASM with (x_0, y_0, μ);
Initialize $\mathbf{W} = \mathbf{X}(1 : 2MN)$ and rearrange $\mathbf{W}$ as size $M \times 2N$;
Initialize $\mathbf{P} = \mathbf{X}(2MN + 1 : 2MN + 2NN)$ and rearrange $\mathbf{P}$ as size $N \times 2N$;
Initialize $\mathbf{\Phi}_1, \mathbf{\Phi}_2, \cdots, \mathbf{\Phi}_N \in \mathbb{R}^{M \times N}$;
for $i = 1 : N$ **do**
    $[\mathbf{P}', \mathbf{C}]$ = SortC($\mathbf{P}(i, :)$) {Sort the i-th row and $\mathbf{C}$ is the index vector.};
    for $j = 1 : N$ **do**
        $\mathbf{\Phi}_i(:, j) = \mathbf{W}(:, \mathbf{C}(j))$;
    end
    Convert the elements of $\mathbf{\Phi}_i$ into 1 or -1 by Eq. (9).
end

2.3.3. Matrix Encoding Embedding

To achieve a better embedding performance, the matrix encoding [35] is introduced to embed the secret image into carrier image. Using more bits, the matrix encoding can represent a number of information bits with acceptable

data loss. It can be described using a triple (n, k, t), where n is the representing bit number, k is the bit number to be represented and t is the maximum changed bit number. Suppose that a codeword $b = \{b_1 b_2 \dots b_n\}$ is the bits that can be changed in a block, $x = \{x_1 x_2 \dots x_k\}$ contains the secret bits and $b' = \{b'_1 b'_2 \dots b'_n\}$ is the modified codeword having embedded secret bits. Then the encoding process can be described as

Step 1: A function f is defined as

$$f(b) = (b_1 \times 1) \oplus (b_2 \times 2) \oplus \cdots \oplus (b_n \times n), \quad (f(b) \in [0, 2^k - 1]) \tag{10}$$

where $\oplus$ means bitwise xor operation. From this function, we can extract k bits data from a codeword b.

Step 2: Find the position where the bit needs to be changed by

$$s = f(b) \oplus x \tag{11}$$

Step 3: The rule to change the codeword b is illustrated as

$$b' = \begin{cases} b, & s = 0 \\ \{b_1, b_2, \dots, 1 - b_i \dots, b_n\}, & s = i \end{cases} \tag{12}$$

Replace b by b'.

Step 4: Repeat *Step 1* to *Step 3* until $f(b) = x$.

In this paper, the matrix encoding with $(n = 3, k = 2, t = 1)$ is used to embed a secret image into a carrier image. This indicates that each pixel in the carrier image is regarded as a block and the least three significant bits are the changeable bits to embed the bits of the secret image.

3. Visually Secure Image Encryption Scheme

This section presents a new visually secure image encryption scheme. In the encryption process, the plain image is first encrypted to be a secret image, which is then embedded into a carrier image to get visually secure cipher image. In the decryption process, the secret image is first extracted from the cipher image, and then decrypted to obtain the plain image. Fig. 2 show the structure of the proposed encryption scheme. It includes two stages: encryption and embedding. In the encryption stage, the SWT is used to decompose the plain image, the 2D cat map is used to randomly shuffle the pixels, and the column-based adaptive thresholding is to generate sparse image, ensuring that each column has the same number of zero. The secret key is used to generate the initial states, which are employed by the 2D-LASM to generate the measurements matrices, and parameters in diffusion and matrix coding. The PCS sampling is performed to the sparse image in the column-wise manner. After performing the quantification and diffusion to the sampling result, a secret image can be obtained. In the embedding stage, the matrix encoding

technique is used to embed the secret image into a carrier image under the control parameters generated by the 2D-LASM. Since each step in the encryption process is reversible, the decryption scheme is the combination of the inverse operation of the encryption and its structure is shown in Fig. 3, where the reconstruction stage is the inverse operation of the encryption stage and the extracting stage is the inverse operation of the embedding stage.

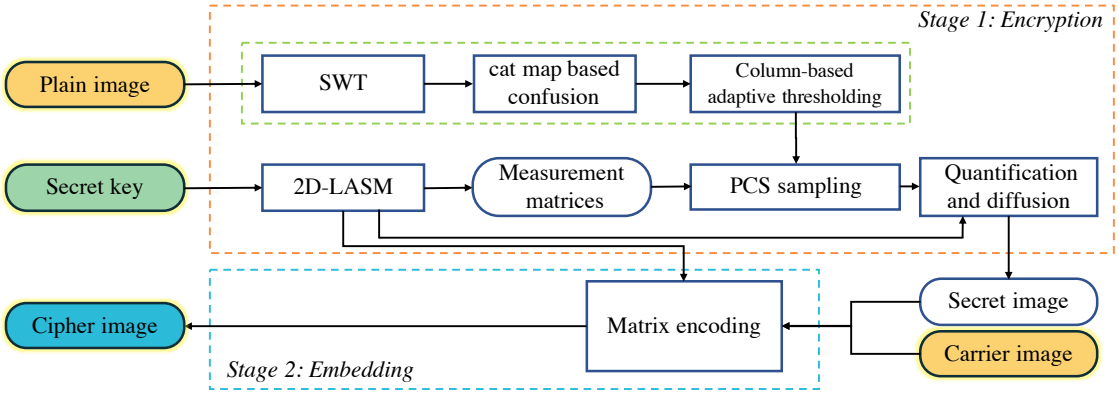


Figure 2: The structure of the proposed visually secure image encryption scheme.

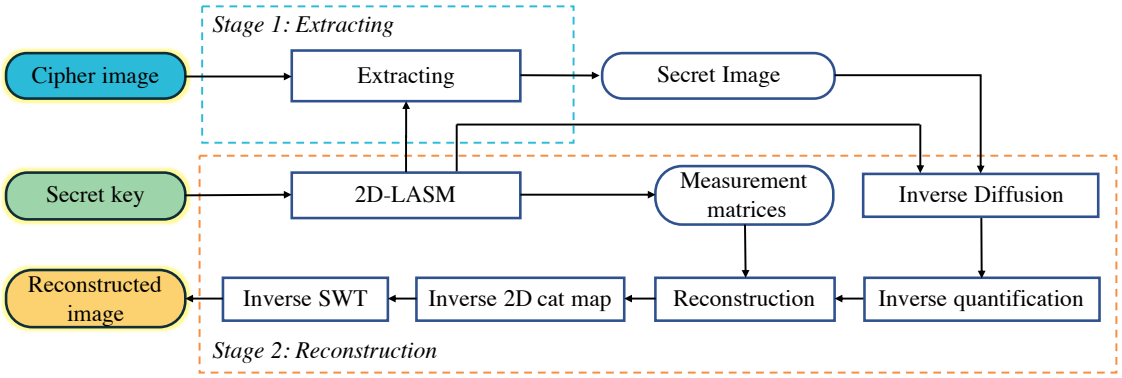


Figure 3: The structure of decryption scheme.

空格

3.1. Secret Key

The secret key $\mathbf{K}$ is composed of 256 bits that used to generate the initial states of the 2D-LASM. Firstly, a hash function SHA-256 is performed to the secret key to enhance the security level. Then the hashed result contains six parts, namely $\mathbf{K}' = \{x_0, y_0, \mu, \gamma_1, \gamma_2, \gamma_3\}$, where (x_0, y_0, μ) are original initial states for 2D-LASM, $(\gamma_1, \gamma_2, \gamma_3)$ are interference parameters. Algorithm 2details the generation procedures of initial states of the 2D-LASM. Three groups of initial states $(x_0^{(1)}, y_0^{(1)}, \mu^{(1)})$, $(x_0^{(2)}, y_0^{(2)}, \mu^{(2)})$ and $(x_0^{(3)}, y_0^{(3)}, \mu^{(3)})$ can be obtained, and they are used in the generating the measurement matrices, and the parameters in diffusion and matrix coding, respectively.

process of

Algorithm 2: The generation of initial states for 2D-LASM

Input: Secret key $\mathbf{K}$ with length of 256 bits

Output: Initial states $(x_0^{(1)}, y_0^{(1)}, \mu^{(1)})$, $(x_0^{(2)}, y_0^{(2)}, \mu^{(2)})$ and $(x_0^{(3)}, y_0^{(3)}, \mu^{(3)})$

$\mathbf{K}' = \text{SHA256}(\mathbf{K})$;

$x_0 = (\sum_{i=1}^{64} \mathbf{K}'[i] \times 2^{i-1})/2^{64}$;

$y_0 = (\sum_{i=65}^{128} \mathbf{K}'[i] \times 2^{i-65})/2^{64}$;

$\mu = (\sum_{i=129}^{192} \mathbf{K}'[i] \times 2^{i-129})/2^{64}$;

$\gamma_1 = (\sum_{i=193}^{213} \mathbf{K}'[i] \times 2^{i-193})/2^{21}$;

$\gamma_2 = (\sum_{i=214}^{234} \mathbf{K}'[i] \times 2^{i-214})/2^{21}$;

$\gamma_3 = (\sum_{i=235}^{256} \mathbf{K}'[i] \times 2^{i-235})/2^{22}$;

for $i = 1 : 3$ **do**

  $x_0^{(i)} = ((x_0 + x_0 \times 2^{i\times 5} \times \gamma_i) \mod 1) + 10^{-5}$;

  $y_0^{(i)} = ((y_0 + y_0 \times 2^{i\times 5} \times \gamma_i) \mod 1) + 10^{-5}$;

  $\mu^{(i)} = ((\mu + \mu \times 2^{i\times 5} \times \gamma_i) \mod 0.4) + 0.5$;

end

3.2. *Encryption and Reconstruction*

Here, we describe the encryption process in the forward operation, and the related reconstruction process in the backward operation. Suppose the plain image $\mathbf{P}$ to be encrypted is of size $N \times N$. The detailed steps is described as follows:

Step 1: Apply the SWT introduced in Section 2.3.1 on $\mathbf{P}$. The coefficient matrix $\mathbf{P1}$ with size $N \times N$ is calculated as

$$\mathbf{P1} = \Psi \times \mathbf{P} \times \Psi^T, \tag{13}$$

where Ψ is the orthogonal wavelet matrix computed by Eq. (5) with layer $\lfloor \log(N) \rfloor$. The inverse operation to recover the plain image $\mathbf{P}$ is

$$\mathbf{P} = \Psi^T \times \mathbf{P1} \times \Psi. \tag{14}$$

Step 2: The 2D cat map is used to randomly shuffle the pixel positions of the image $\mathbf{P1}$, and it is defined as

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \mod N \tag{15}$$

where (x, y) is a pixel position in the original image and (x', y') is a pixel position in the shuffled image, a and b are two parameters. Iterate the 2D cat map c times to the image $\mathbf{P1}$, a totally shuffled image $\mathbf{P2}$ can be generated with randomly-distribution pixels. Note that the parameters a, b and c can affect the reconstruction quality. The

then

image **P1** can be recovered from **P2** by iterating the inverse 2D cat map c times using the same parameters, and the inverse 2D cat map is define as

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} ab+1 & -a \\ -b & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} \mod N \tag{16}$$

Step 3: Apply the CBAT presented in Section 2.3.1 to **P2** to obtain the sparse image. Predefine the compression ratio of the plain image as CR. Generate N measurement matrices, $\mathbf{\Phi}_1, \mathbf{\Phi}_2, \cdots, \mathbf{\Phi}_N$, using the Algorithm 1, where $M = N \times CR$ and the initial state of 2D-LASM is $(x_0^{(1)}, y_0^{(1)}, \mu^{(1)})$.

Step 4: The i-th measurement matrix $\mathbf{\Phi}_i$ is used to sample the i-th column of **P2** using the parallel CS. Then the compressed image **P3** with size $M \times N$ is generated. The image **P2** can be reconstructed from **P3** via different CS reconstruction methods.

Step 5: Quantify the pixel values of **P3** to be integer of range $[0, 255]$ and the quantified matrix **P4** is obtained. The quantification process is calculated as

$$\mathbf{P4} = \left\langle \frac{\mathbf{P3} - P_{\min}}{P_{\max} - P_{\min}} \times 255 \right\rangle, \tag{17}$$

where $P_{\min}$ and $P_{\max}$ are the minimum and maximum values of **P3**, respectively, and $\langle \cdot \rangle$ is to get the nearest integer. After the quantification, an image with integer pixel value **P4** is obtained. The inverse operation of quantification is defined as

$$\mathbf{P3} = \frac{\mathbf{P4} \times (P_{\max} - P_{\min})}{255} + P_{\min} \tag{18}$$

Step 6: A diffusion operation is developed to randomly change the pixel value and spread the little change to the whole image. Specifically, a chaotic sequence $X = \{x_i\}_{i=1}^{MN}$ is generated by 2D-LASM with initial state $(x_0^{(2)}, y_0^{(2)}, \mu^{(2)})$. Then convert the sequence $\mathbf{X}$ into integers $\mathbf{V} = \{v_1, v_2, \cdots, v_{MN}\}$ by

$$v_i = \langle x_i \times 2^{30} \rangle \mod 256 \tag{19}$$

Let $P4_i$ and s_i donate the ith element of **P4** and the ith element of secret image **S**. Then s_i can be calculated by

$$s_i = \begin{cases} P4_i \oplus v_i & i = 1 \\ P4_i \oplus v_i \oplus s_{i-1} & i > 1 \end{cases} \tag{20}$$

Reshape the **S** as size $M \times N$ and the secret image is obtained. The reservable operation is

$$P4_i = \begin{cases} s_i \oplus v_i & i = 1 \\ s_i \oplus v_i \oplus s_{i-1} & i > 1 \end{cases} \tag{21}$$

3.3. Embedding and Extracting

After the plain-image is encrypted to be a secret image, the secret image is then embedded into a carrier image to further enhance the security level. Then a visually secure cipher image can be obtained by embedding the secret image $\mathbf{S}$ into carrier image $\mathbf{Q}$. Suppose the size of the carrier image is $M_2 \times N_2$. To completely recover the secret image, the sizes of the carrier image and the secret image should satisfy that

$$M_2 \times N_2 \geq M \times N \times 4 \tag{22}$$

Before the embedding, two chaotic sequences $\mathbf{X} = \{x_1, x_2, \cdots, x_{M_2N_2}\}$ and $\mathbf{Y} = \{y_1, y_2, \cdots, y_{M_2N_2}\}$ are generated using the 2D-LASM with initial state $(x_0^{(3)}, y_0^{(3)}, \mu^{(3)})$. Select the last MN elements of $\mathbf{Y}$ to form another sequence $\mathbf{Y}' = \{y'_1, y'_2, \cdots, y'_{MN}\}$. Sort the sequence $\mathbf{X}$ and $\mathbf{Y}'$ in increasing order and generate two index vectors $\mathbf{I}_x$ and $\mathbf{I}_y$. Then rearrange the secret image $\mathbf{S}$ as a vector $\{s_1, s_2, \cdots, s_{MN}\}$. By decomposing each pixel into 8 bits, a binary matrix with size $MN \times 8$ can be generated, and rearrange it as $\mathbf{S}'$ with size $4MN \times 2$. Finally, the binary matrix $\mathbf{S}'$ can be embedded into the carrier image $\mathbf{Q}$ by matrix encoding under the control of $\mathbf{I}_x, \mathbf{I}_y$. Algorithm 3 shows the pseudo-code of embedding the secret $\mathbf{S}$ into $\mathbf{Q}$ using matrix encoding.

Algorithm 3: The procedure of embedding a secret image into a carrier image.

Input: The secret image $\mathbf{S}$ with size $M \times N$, the carrier image $\mathbf{Q}$ with size $M_2 \times N_2$, the index vectors $\mathbf{I_x}$ and $\mathbf{I_y}$.

Output: The visually secure cipher image $\mathbf{C}$

Decompose each pixel of $\mathbf{S}$ into 8 bits, and rearrange its size to obtain a binary matrix $\mathbf{S}'$ with size $4MN \times 2$;
Decompose each pixel of $\mathbf{Q}$ into 8 bits, and rearrange its size to obtain a binary matrix $\mathbf{Q}'$ with size $M_2N_2 \times 8$;
for $i = 1 : 4MN$ **do**
    $x = \mathbf{I_x}(i)$ and $y = \mathbf{I_y}(i)$;
    $a = [\mathbf{Q}'(x, 6) \cdot 1] \oplus [\mathbf{Q}'(x, 7) \cdot 2] \oplus [\mathbf{Q}'(x, 8) \cdot 3]$;
    $b = a \oplus [\mathbf{S}'(y, 1) \cdot 2 + \mathbf{S}'(y, 2)]$;
    if $b! = 0$ then change $\mathbf{Q}'(x, b + 5)$
end
Convert each row of $\mathbf{Q}'$ to be a decimal integer;
Rearrange $\mathbf{Q}'$ to obtain the cipher image $\mathbf{C}$ with size $M_2 \times N_2$;

Using the inverse operation of matrix coding, the secret image $\mathbf{S}$ can be completely extracted from the cipher image $\mathbf{C}$ using the same index matrices $\mathbf{I}_x$ and $\mathbf{I}_y$.

3.4. Discussion

Because many effictive techniques such as the adaptive sparsification, random-order measurement matrix and matrix coding are introduced in our proposed scheme, the scheme is able to protect a plain image with high security

level and achieve many advantages.

First, the compression ratio and reconstruction quality of the original image can be greatly enhanced, due to the adaptive sparsification. The SWT can decompose the plain image with high sparsity performance and processing efficiency, the 2D-LASM can generate chaotic sequences with uniform distribution, and the CBAT can ensure that every column of an image has the same sparsity and can be reconstructed with acceptable data loss. Thus, one can exactly setting the compression ratio to the original image and obtain a high-quality reconstructed result.

Second, because of the matrix coding, the cipher image can achieve a high quality and has a similar data loss for different carrier images. Suppose the carrier image $\mathbf{I}$ and cipher image $\mathbf{C}$ are with sizes $M_2 \times N_2$, and the size of the secret image to be embedded is $M_2 \times N_2/4$, which is maximum value of satisfying the requirements of matrix coding in Eq. (22). According to the embedding process described in Algorithm 3, each pixel in $\mathbf{I}$ has the same 25% probabilities to change one of its last three bits, or keep no change. This indicates that $|\mathbf{I}(i, j) - \mathbf{C}(i, j)|$ can be 0, 1, 2, and 4 with the same 25% probabilities. Then the difference between the carrier image and the cipher image, denoted by the Peak Signal Noise Ratio (PSNR) [36], can be calculated as

$$\begin{aligned} PSNR &= 10 \times \log_{10}\left(\frac{255^2 \times M_2 \times N_2}{\sum_{i=1}^{M_2}\sum_{j=1}^{N_2}(\mathbf{I}(i,j) - \mathbf{C}(i,j))^2}\right) \\ &= 10 \times \log_{10}\left(\frac{255^2 \times M_2 \times N_2}{\sum_{i=1}^{M_2}\sum_{j=1}^{N_2} 0.25 \times (0^2 + 1^2 + 2^2 + 4^2)}\right) \approx 40.9292. \end{aligned} \tag{23}$$

This theoretically indicate that the embedding process has a greatly high performance and the cipher image has few data loss compared with the carrier image.

Thirdly, the scheme can well balance the trade-off between the size of the carrier image and the compression ratio CR. From Eq. (22), one can obtain that when the size of the carrier image is fixed, the compression ratio has a maximum value, and when the compression ratio is fixed, the size of the carrier image has a minimum value. Thus, one is flexible to set the compression ratio and the size of the carrier image. Besides, because the embedding process is a completely reversible operation, the secret image can be completely extracted from the cipher image without data loss.

Finally, the proposed scheme can protect the plain image with a high security level. This is because the PCS sampling, diffusion and matrix coding are all under the control of the chaotic sequences generated by the 2D-LASM. The initial state of the 2D-LASM is generated by the secret key and the 2D-LASM is extremely sensitive to the change of initial state and it can generate chaotic sequences with uniform distribution. Thus, the cipher image has a high security level and can resist the commonly used security attacks.

4. Simulation Results and Analysis

This section simulates the proposed visually secure image encryption scheme and analyzes its performance. To show a relatively fair simulation results, ten classical and widely used images are tested in our experiments and these

while

images include ~~the images~~ ”Brain”, ”Finger”, ”Girl”, ”Bridge”, ”Barbara”, ”Peppers”, ”Lena”, ”Jet”, ”Airplane”, and ”Baboon”. The former two images have the size of 256×256, while the latter eight images have the size of 512×512.

4.1. Simulation Results

This subsection simulates the proposed scheme using different images as the plain images and carrier images. The secret key is randomly generated as ’5B24E5F4C1257EA9B12CD5821F085B87E404A078E21C08F145E5B187C3B33E7E0’, the filter used in SWT is the sym8, the compression ratio CR is set as 0.25 and the reconstruction method in PCS is the SL0 with parameters $\theta = 0.3445$ and $\omega = 1.404$ for CBAT.

(a) (b) (c) (d) (e)

Figure 4: Encryption and decryption results. The images in these five rows represent the plain images, secret images, carrier images, cipher images and reconstructed images, respectively.

Fig. 4 shows the simulation results and each column is an individual experiment. Because the compression ratio $CR = 0.25$, the secret images are the 25% large with the plain image and they are noise-like. After embedding the

secret images into the carrier images, the generated cipher images have the same visual effects with the carrier images. This can well protect the secret images because the meaningful images can greatly reduce the attentions of the attacks. Because the embedding process is completely invertible and the reconstruction of PCS has a high performance, the reconstructed images have high quality and visually same effects with the plain images.

To qualitatively assess the performance of our proposed scheme, we use the PSNR presented in Eq. (23) and mean structural similarity (MSSIM) [37] to measure the quality of cipher images and reconstructed images. Among these two indexes, MSSIM can qualitatively describe the structural similarity between two images, and is calculated by

$$\begin{cases} C_1 = (k_1 \times L)^2 \\ C_2 = (k_2 \times L)^2 \\ \mathrm{SSIM}(x, y) = \dfrac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \\ \mathrm{MSSIM}(X, Y) = \frac{1}{M}\sum_{k=1}^{M} \mathrm{SSIM}(x_k, y_k) \end{cases} \tag{24}$$

where $k_1 = 0.01, k_2 = 0.03$, L is the gray level of image and $L = 255$, X and Y represent two images, x and y are blocks of images X and Y respectively, μ_α and σ_α ($\alpha = x, y$) are the average value and variance value of block α respectively, σ_{xy} stands for the covariance of blocks x and y, the total number M of image blocks is 64.

that

Table 1 shows the test results. As can be seen, all the PSNR and MSSIM values between the cipher and carrier images are above 40.9 and 0.99, respectively. This demonstrates the cipher images have high similarity with the carrier images. Besides, all the reconstructed images also achieve high PSNR and MSSIM values from the plain images, and this indicates that the reconstructed images have good quality. Thus, our proposed scheme not only can concurrently compress and encrypt a plain image, but also generate cipher image with high quality to ensure a visual security. With these significant properties, our proposed scheme has the potential to satisfy the requirements of visual security, compression ratio and reconstruction quality in practical applications.

Table 1: The PSNR and MSSIM values between the cipher and carrier images, and between the reconstructed and plain images.

Plain images	Carrier images	Cipher and carrier images		Reconstructed and plain images	
		PSNR	MSSIM	PSNR	MSSIM
Brain	Finger	40.9187	0.9967	38.1003	0.9502
Girl	Bridge	40.9089	0.9973	37.3884	0.9686
Barbara	Peppers	40.9391	0.9917	30.2668	0.9372
Lena	Jet	40.9480	0.9944	35.4482	0.9661
Airplane	Baboon	40.9289	0.9968	34.6343	0.9581

4.2. Reconstruction Quality Against Compression Ratio

The quality of the reconstructed images is highly related to the compression ratio. To investigate the relationship between the quality of constructed images and the compression ratio CR, we simulate our scheme using two different

these

draw

construction methods in the inverse operation of the PCS, namely the OMP and SL0. According to our experiments, the best reconstruction quality can be achieved when the parameters of CBAT are setting as $\theta = 0.2576$, $\omega = 1.265$ in OMP, and setting as $\theta = 0.3445$, $\omega = 1.404$ in SL0. The images "Barbara", "Airplane", "Girl" and "Lena" are used as the plain image, respectively, and the compression ratio is set as $CR = \{0.1, 0.2, \cdots, 0.8\}$.

Fig. 5 shows the PSNR values between the reconstructed and plain images with different CR and different reconstruction methods. From the figures, we can ge the following conclusions. (1) When setting the PSNR=30 db and SL0 as the reconstruction method, the minimum CR is about 0.1, 0.25, 0.15 and 0.15 for the images "Girl", "Barbara", "Lena" and "Airplane", respectively. This indicates that the proposed scheme can achieve a high performance to concurrently compress and encrypt an image. (2) The quality of the reconstructed image is highly related to the reconstruction method. For the reconstruction method OMP, the PSNRs between the reconstructed and original images growth fast with the increment of the CR. However, for the reconstruction method SL0, the PSNR keeps almost the same or even decrease slightly when the CR is bigger than 0.5. (3) When the CR is smaller than 0.5, the SL0 method can reconstruct the images with a higher quality than the OMP method. In the contract, when the CR is bigger than 0.5, the OMP is more effective. Thus, the OMP reconstruction is more suitable for the light compression applications while the SL0 is more suitable for the heavy compression applications.

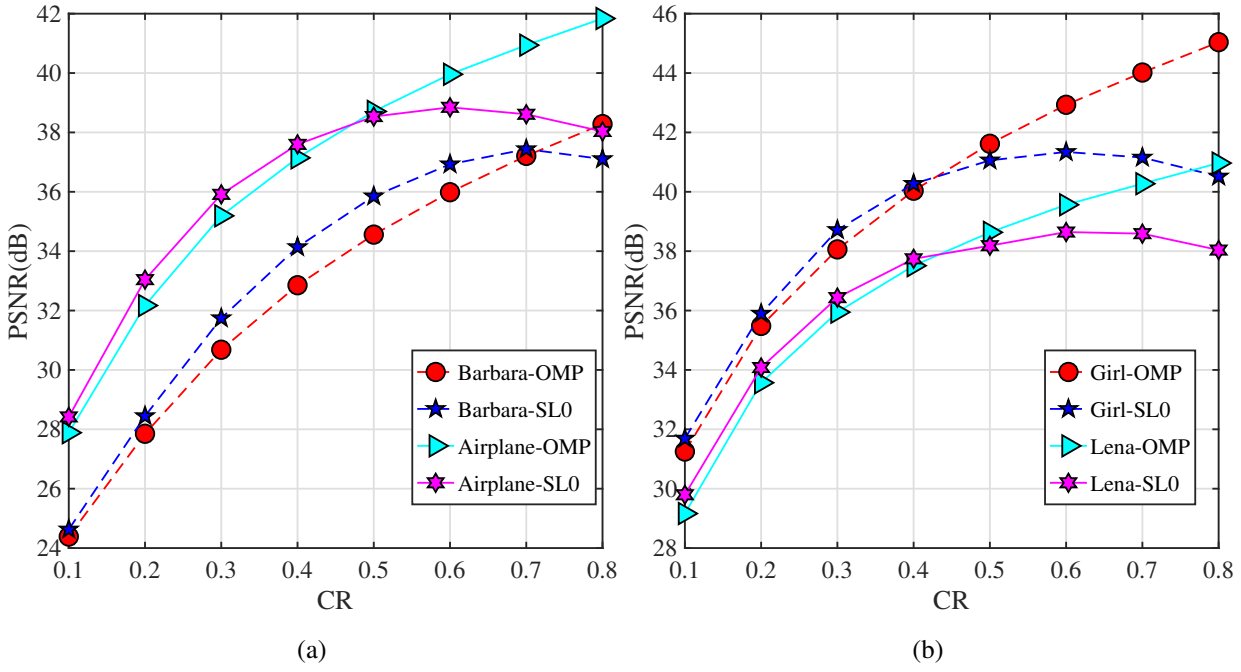


Figure 5: The PSNR values between the reconstructed and plain images with different CR and different reconstruction methods. (a) Images "Barbara" and "Airplane"; (b) images "Girl" and "Lena".

4.3. Carrier Image Against Reconstructed Image

Here, we test the performance of the proposed scheme for different carrier images. Fig. 6 shows the visual quality of the cipher images and the reconstructed images using the images ”Bridge”, ”Peppers”, ”Jet” and ”Baboon” as the carrier images, respectively. The plain image is the image ”Lena”. One can see that with different carrier images, the cipher images always have high visual quality and are similar to the corresponding carrier images. Besides, the reconstructed images can achieve a similar visual quality for different carrier images.

(a) (b) (c) (d)

Figure 6: The quality of the constructed images against different different carrier images. (a)-(d) are four independent experiments, where the first, second and third rows are the carrier images, cipher images and reconstructed images.

To qualitatively test effects of the carrier images on the quality of the cipher and reconstructed images, we calculate the PSNRs and MSSIMs between the cipher and carrier images, and between the reconstructed and the plain images. Table 2 shows the PSNR values and MSSIM values. As can be observed, all the PSNR values and MSSIM values between the reconstructed and original images are the same for different carrier images. This is because the embedding process is a completely reversible operation, the selections of the carrier images cannot affect the quality of the reconstructed image. Besides, all the PSNR values between the cipher and carrier images are very close to the

depends

theoretical value calculated in Eq. (23). The little differences are cause by the statistic errors in the matrix coding. These indicate that the proposed scheme is robust for the carrier image. Many similar schemes don't have this property and their performance ~~are~~ highly dependent on the selections of carrier images. Our proposed scheme can overcome this weakness and one is flexible to select any digital image as the carrier image to achieve a high performance.

Table 2: The quality of the cipher and reconstructed images affected by different carrier images, where the plain image is the image "Lena".

Carrier image	Cipher and carrier images		Reconstructed and plain images	
	PSNR (db)	MSSIM	PSNR (db)	MSSIM
Bridge	40.9280	0.9973	35.4482	0.9661
Peppers	40.9154	0.9916	35.4482	0.9661
Jet	40.9480	0.9944	35.4482	0.9661
Baboon	40.9455	0.9967	35.4482	0.9661

5. Security Analysis and Performance Comparison

This section analyzes the security level of the proposed encryption scheme and compares its performance with some newly developed schemes.

5.1. Key Security Analysis

An effective image encryption algorithm should have large enough key space to resist brute-force attack, and the ideal key space should be bigger than 2^{100} [38]. The secret key in our proposed scheme consist of 256 bits and its key space is 2^{256}, which is sufficient to satisfy the requirement of key space. Meanwhile, the secret key should be high sensitive in both the encryption and decryption processes.

To test the sensitivity of the secret key, a secret key $\mathbf{K}_1$ is randomly generated to encrypt the plain image "Lena" and subsequently embed the secret image into a carrier image. By randomly changing one bit of the $\mathbf{K}_1$ in different positions, three new secret keys $\mathbf{K}_2, \mathbf{K}_3, \mathbf{K}_4$ are obtained as follows.

$$\mathbf{K}_1 = \text{7A09E5F4B5241E49B12CD5521E085A87F414A078E51C08D14535B487CBB3347A0},$$
$$\mathbf{K}_2 = \text{7A09E5F4B5241E49B12CD5521E085A87F414A078E51C08D14535B487CBB3347A}\mathbf{1},$$
$$\mathbf{K}_3 = \text{7A09E5F4B5241E49B12CD5521E085A87F414A078E5}\mathbf{3}\text{C08D14535B487CBB3347A0},$$
$$\mathbf{K}_4 = \text{7A09E5F4B5241E4}\mathbf{B}\text{B12CD5521E085A87F414A078E51C08D14535B487CBB3347A0}$$

The four secret keys are separately used to decrypt the same cipher image, and the decryption results are shown in Fig. 7. As can be seen, the decryption results with incorrect secret key are totally different with that with correct secret key. Without secret key, one can't get any useful information about the plain image. Besides, the number of

pixel change rate (NPCR) [39] is used to calculate the difference between the reconstructed images decrypted by the correct and incorrect keys. For two images $\mathbf{O}_1$ and $\mathbf{O}_2$ with the same size $M \times N$, their NPCR is defined as

$$NPCR = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\delta_{\mathbf{O}_1(i,j),\mathbf{O}_2(i,j)}}{MN} \times 100\%, \tag{25}$$

where $\delta_{\mathbf{O}_1(i,j),\mathbf{O}_2(i,j)}$ is 1 if $\mathbf{O}_1(i,j) = \mathbf{O}_2(i,j)$; otherwise it is 0. The NPCRs between the correct reconstructed image in Fig (a) and incorrect reconstructed images in Fig (b)-(d) are 0.9998, 0.9986 and 0.9986, respectively. This demonstrates that the proposed scheme has high sensitivity on the secret key.

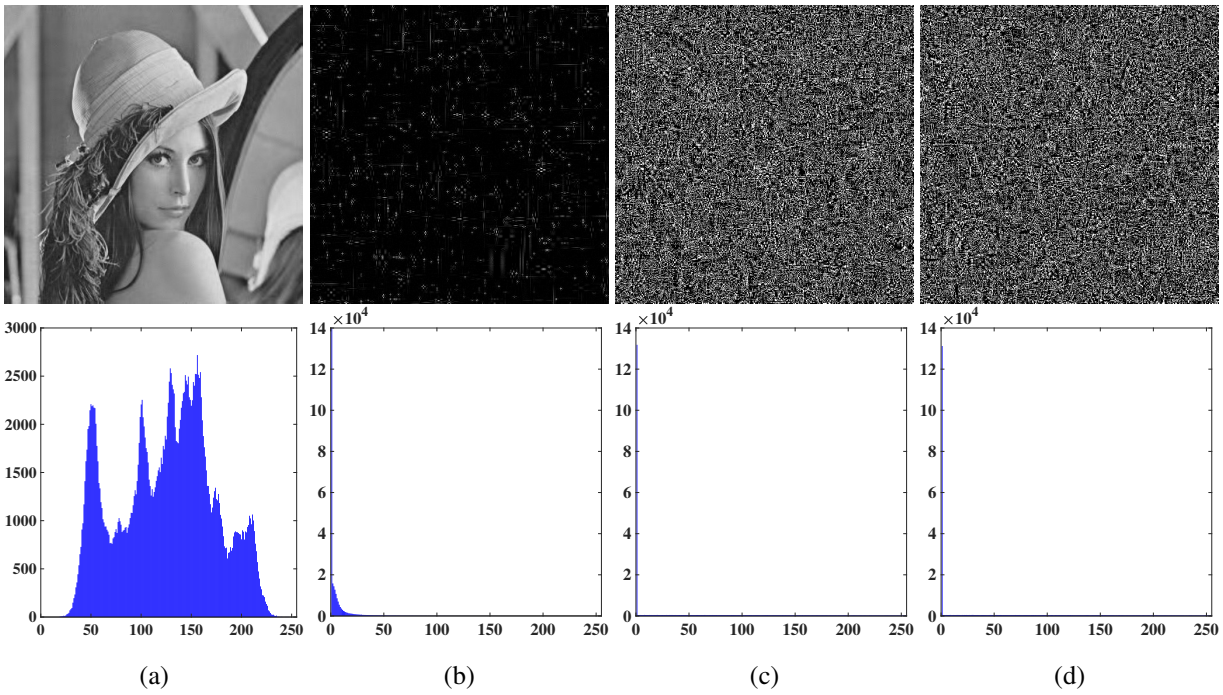


Figure 7: Key sensitivity analysis. Decryption results with (a) correct secret key $\mathbf{K}_1$ and (b)-(d) three incorrect secret keys $\mathbf{K}_2, \mathbf{K}_3, \mathbf{K}_4$.

5.2. *Histogram Analysis*

The histogram of an image can directly reflect its statistical properties. For a natural image, its histogram usually has many patterns and thus one can obtain much useful information from its histogram. From the information theory, one can get the least information when all the pixels distribute uniformly. Thus, an effective encryption algorithm should have the ability to generate secret images with uniform-distribution. Here, we use information entropy to measure the pixel distributions of the secret images, and the entropy H of a signal s is calculated as

$$H(s) = -\sum_{i=0}^{n} P(s_i)\log_2 P(s_i) \tag{26}$$

Where $P(s_i)$ is the probability of the ith possible value s_i. For an 8-bit grayscale image, n is 255 and the maximum entropy is 8 when the pixels are absolutely uniform-distributed.

We successively subject images "Peppers" and "Lena", "Baboon" and "Girl", "Jet" and "Airplane" to the proposed scheme. Fig. 8 shows the histograms of plain images and their secret images, and it is clear that the histograms of the plain images have some patterns while the histograms of the secret images are uniform-distributed. One cannot get any useful information from the histograms of secret image. As for the information entropy, all secret images has the entropy that is very close to 8.

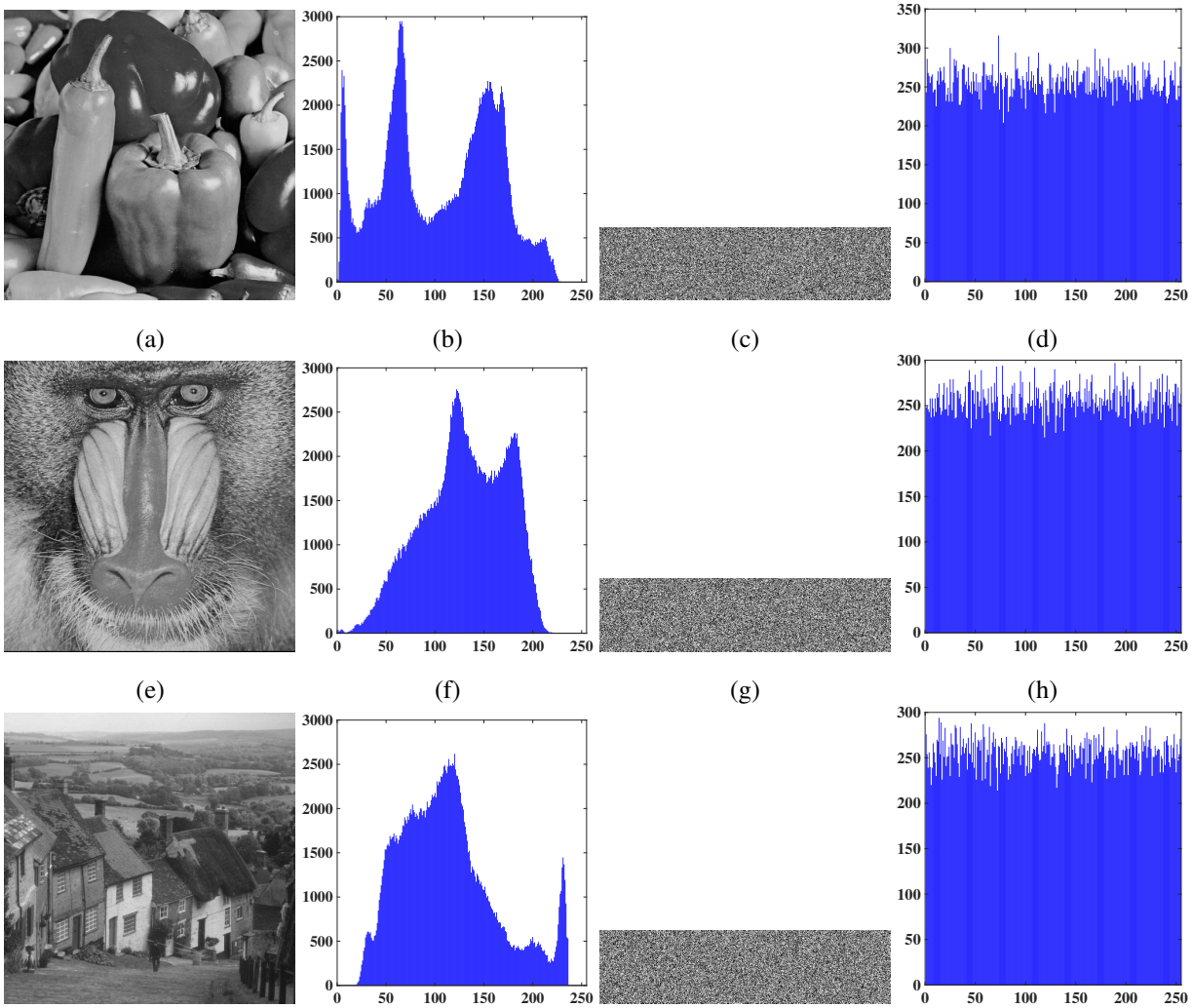


Figure 8: Histogram analysis about the plain images and their corresponding secret images. Pictures in thses four columns represents plain images, histograms of plain images, secret images and histograms of secret images, respectively. The entropies of these three plain images are 7.5715, 7.3579 and 7.4778, while the entropies of their secret images are 7.9969, 7.9972 and 7.9974 respectively.

To achieve a higher security level, the cipher images are expected to have similar visual effects with the carrier images. Fig. 9 shows the histograms of the carrier images and their corresponding cipher images. One can see that the visual effects and histograms between the carrier image and its cipher image are quite similar. To qualitatively test the

difference between the carrier and cipher images, the histogram intersection [40] is used to describe their similarity. For two histograms X and Y with N bins, their distance of histogram intersection is defined as

$$HI(X,Y) = \frac{\sum_{k=1}^{N} min(X_k, Y_k)}{\sum_{k=1}^{N} Y_k} \tag{27}$$

A larger value mean bigger similarity of the two histograms. Table 3 shows the distances of histogram intersection between the carrier images and cipher images. One can see that all the distances of histogram intersection are close to 1. This demonstrates the high similarity betwen the carrier images and its cipher images.

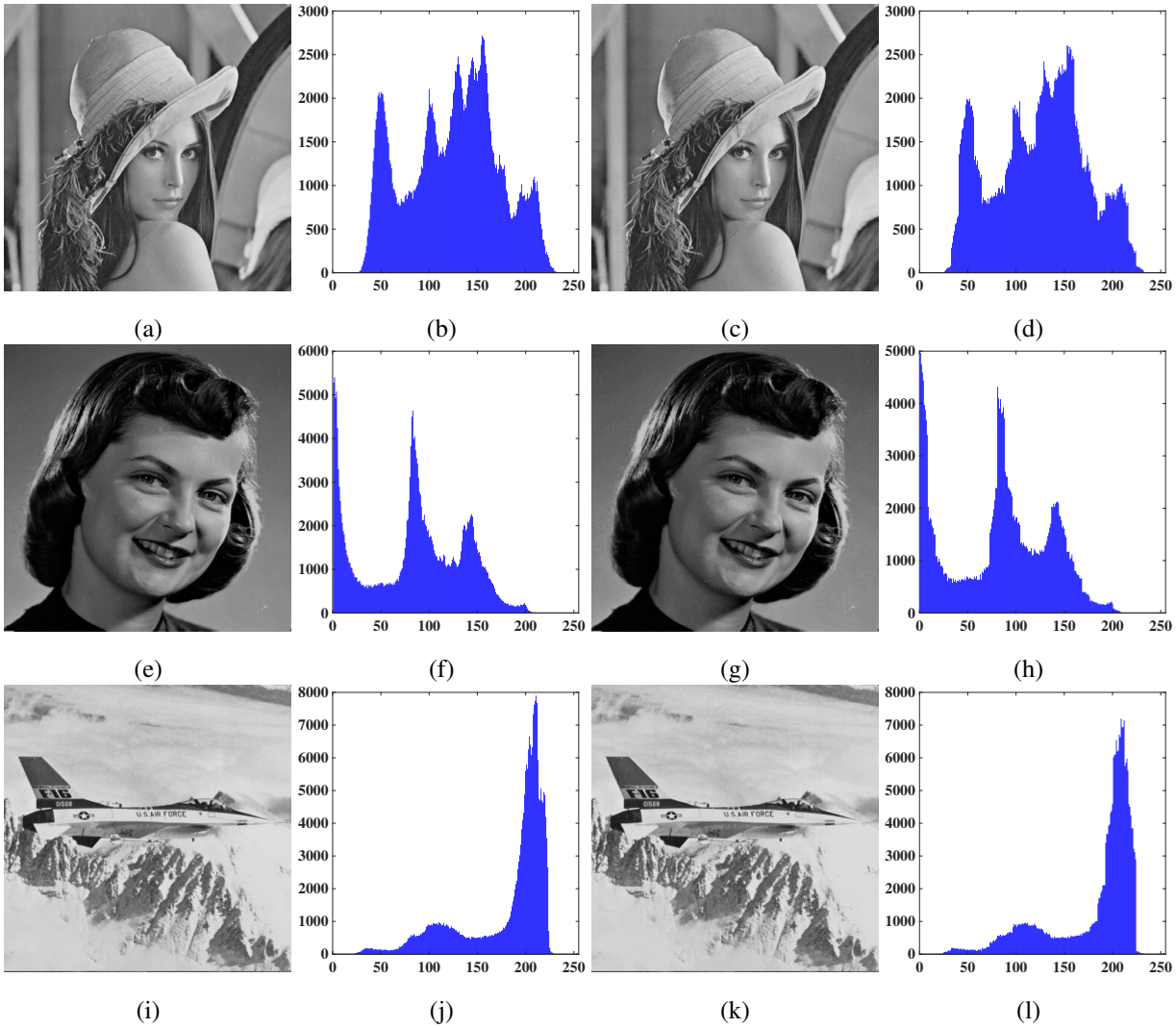


Figure 9: Histogram analysis about the carrier images and their corresponding cipher images. Pictures in thses four columns represents carrier images, histograms of carrier images, cipher images and histograms of cipher images, respectively.

Table 3: The distances of histogram intersection between the carrier and cipher images.

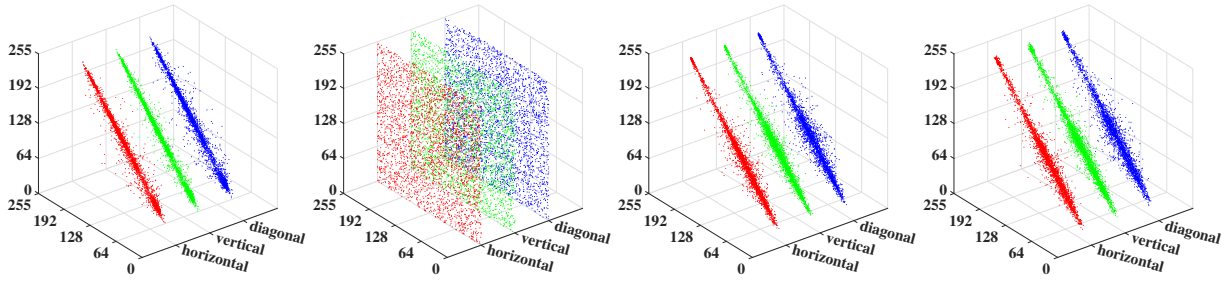
Plain images	Peppers	Baboon	Jet
Carrier images	Lena	Girl	Airplane
Distance	0.9702	0.9610	0.9572

5.3. Adjacent Pixel Correlation

A natural image has strong correlation among the adjacent pixels. This correlation can benefit to the reconstruction of original image without secret key. Thus, a secret image is expected to have weak correlation among the adjacent pixels. Here, we evaluate the correlation of adjacent pixels using correlation coefficient. To calculate the correlation coefficient of an image, 3000 pixels are randomly selected in the image, and then the correlation coefficients between these pixels with their adjacent pixels in horizontal, vertical and diagonal direction are calculated. Assume $X = \{x_i\}_{i=1}^{3000}$ and $Y = \{y_i\}_{i=1}^{3000}$ are two sequences of pixels and every pair (x_i, y_i) are adjacent pixels, the correlation coefficient of X, Y can be calculated as

$$CC_{XY} = \frac{Cov(X, Y)}{\sigma_X \sigma_Y}, \tag{28}$$

where σ_X donates the standard deviation of X and $Cov(X, Y)$ is the covariance of X and Y. Fig. 10 plots the adjacent pixel pairs of the plain, secret, carrier and cipher images, and Table 4 shows the numeral results. The used plain image is the image "Lena" and carrier image is the image "Jet". As can bee seen, the adjacent pixel pairs of the secret image are uniformly distributed in the whole phase plane, which is shown in Fig. 10(b), and their correlation coefficients are close to 0. This indicates that the adjacent pixels in the secret images have weak correlation. Besides, the cipher image has similar correlation coefficients with the carrier image, which implies the good similarity between the cipher and carrier image.



(a) Plain image (b) Secret image (c) Carrier image (d) Cipher image

Figure 10: Adjacent pixel pairs along the horizontal, vertical and diagonal directions in the (a) plain image, (b) secret image, (c) carrier image and (d) cipher image.

5.4. Ability of Resisting Noise and Data Loss

Since almost all the transmission channels are noise channels, the cipher images should have the strong ability to resisting noise and data loss. This indicates that even a cipher image is blurred by the noise or has data loss,

Table 4: Correlation coefficients along the horizontal, vertical and diagonal directions in the plain image, secret image, carrier image and cipher image.

	Plain image	Secret image	Carrier image	Cipher image
Horizontal	0.9714	-0.0069	0.9682	0.9689
Vertical	0.9856	0.0047	0.9743	0.9716
Diagonal	0.9634	-0.0494	0.9489	0.9489

against

the decryption process can still recover the most information of the original image. Here, we test this ability of our proposed scheme.

Fig. 11 shows the experimental results with different percentages of data loss and salt and pepper noise. The image "Girl" is used as the plain image while the image "Jet" is used as the carrier image. First, encrypt the plain image using the carrier image. Then, cause a cropping to the cipher image or blur the images. Finally, decrypt the cipher images with the correct key. One can see that the reconstructed images are still meaningful and readable. This indicates that even the data of the cipher image is changed in a certain level, the proposed scheme can still recover most information in the original image. Thus, the proposed scheme has a good robustness on noise pollution and cropping attack.

(a) (a) (b) (c) (d)

Figure 11: Simulation results for the ability of resisting data loss and noise. The first row are the cipher images, while the second row are the corresponding reconstructed images. (a) the original cipher image (b) a 32×32 block cropping; (c) a 64×64 block cropping; (d) 0.1% salt and pepper noise; (e) 1% salt and pepper noise.

5.5. Efficiency Analysis

The fast increment of digital images requires a high efficiency to the encryption process. Here, we theoretically analyze the time complexity of our proposed scheme and experimentally test its speed. Suppose the plain image is of size $N \times N$, the secret image is of size $M \times N$, and the label c_i in subsequently description represents a constant

number. In the encryption stage, the total time complexity for the SWT, 2D cat map confusion and CBAT is $O(c_1N^2)$. For PCS sampling, time mainly costs in the generation of measurement matrices, and the total time complexity is $O(c_2N^2\log(N))$. Since the quantification and diffusion are the linear operations, their time complexity is $O(c_3MN)$. In the embedding stage, because the generating and sorting of chaotic sequences require a proceeding time, the time complexity for the matrix coding in the embedding stage is $O(c_4N^2\log(N^2))$. Among above complexities, $O(N^2\log(N^2))$ has the maximum magnitude, which will determine the actual running time of our proposed encryption scheme. Thus, the total computational complexity of encryption process is $O(cN^2\log(N^2))$. Meanwhile, the time complexity in the decryption process is highly determined by the reconstruction method.

To test the actual running time, the experiment is performed on a computer with Inter(R) Core(TM) i7-8700 @ 3.2GHz. Table 5 lists the encryption and decryption times for images with sizes 256×256, 512×512 and 1024×1024. It can be observed that the average encryption times is 0.0838s, 0.3749s and 2.2120s for images with different size 256×256, 512×512 and 1024×1024 respectively. This is roughly consistent with the theoretical results. On the other hand, the average decyption times are 0.3292s, 2.3768s and 62.8956s for images with different size 256×256, 512×512 and 1024×1024. With the increment of image size, the running time of decryption process grows fast. Thus, our proposed encryption scheme can achieve a high encryption efficiency.

后面这个thus连不起来好像

Table 5: The encryption and decryption times (second) of our proposed scheme for images with different sizes.

	Image size	Lena	Girl	Peppers	Baboon	Barbara	Airplane	Average
Encryption time	256×256	0.0832	0.0828	0.0865	0.0835	0.0841	0.0822	0.0838
	512×512	0.3672	0.3810	0.3761	0.3726	0.3798	0.3724	0.3749
	1024×1024	2.1966	2.2074	2.2761	2.1680	2.2313	2.1921	2.2120
Decryption time	256×256	0.2976	0.3484	0.3180	0.3058	0.3827	0.3224	0.3292
	512×512	2.3626	2.4828	2.3186	2.1993	2.5995	2.2980	2.3768
	1024×1024	65.1505	58.8444	58.8796	66.0253	65.8899	62.5836	62.8956

5.6. Comparison with Latest Schemes

To show the superiority of our proposed encryption scheme, we compare it with some other latest CS-based encryption schemes introduced in [18, 19, 22, 24, 25, 41, 42, 43]. The comparisons are performed from the aspects of the quality of reconstructed image, the quality of cipher image, and the efficiency. To provide a relatively fair comparison, the results of the competing schemes are all directly referenced from the original papers.

The quality of reconstructed image is the most important performance in an image encryption algorithm. Table 6 shows the comparisons of the PSNR values between the plain image and the reconstructed images by different encryption schemes. Since the results in [18, 22, 24] are shown in graphs, we estimate their PSNR valuess from the graphs. The N/A indicates that the corresponding value was not provided. As can been seen, under the same com-

pression ratio CR, the proposed scheme can achieve the largest PSRN values for different images. This indicates that the reconstructed images by the proposed scheme have the highest quality.

Table 6: Comparison of the PSNR values between the plain and reconstructed images in different encryption schemes.

Images	CR	[18]	[19]	[22]	[24]	[25]	[41]	[42]	[43]	Proposed
Lena512	0.25	28.5	N/A	28.5	29.0562	33.4204	N/A	N/A	31.4240	35.4482
	0.5	33.5276	34.5560	33	35	N/A	N/A	N/A	32.9660	38.1840
Lena256	0.25	N/A	N/A	N/A	N/A	N/A	26.0600	26.5600	N/A	29.3184
	0.5	N/A	N/A	N/A	N/A	N/A	29.8200	29.8300	N/A	35.4534
	0.25	28	N/A	N/A	30.2	N/A	N/A	N/A	30.6809	34.7331
		33	31.5132	N/A	34			N/A	31.9825	37.2845

are

proposed

For a visually secure image encryption scheme, the cipher image is expected to have a high quality. We also compare the quality of cipher images between our prosed scheme with the visually secure schemes introduced in [24, 25, 32]. Table 7 shows the comparisons of the PSNR and MSSIM values between the cipher image and the carrier images in different encryption schemes. One can see that the PSNR values of our proposed scheme for different images all approximate to 40.90dB, which is much larger than the PSRN values of other encryption schemes. Besides, the MSSIM values of our proposed scheme is also larger than that of the other schemes. These demonstrate that our proposed scheme can generate a cipher image that is quite similar with the carrier image.

are

Table 7: Comparison of the PSNR and SSIM values between the cipher and carrier images in different encryption schemes.

Plain images	Carrier images	[24]		[25]		[32]		Proposed	
		PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM
Lena	Peppers	18.5136	0.6726	32.3513	0.9257	35.1347	N/A	40.9186	0.9916
Jet	Baboon	23.3967	0.6991	37.8967	0.9833	36.4906	N/A	40.9336	0.9968
Brain	Cameraman	24.8700	0.6488	34.8967	0.9381	35.3534	N/A	40.9145	0.9603
Girl	Airplane	28.2318	0.7021	36.1125	0.9666	36.2169	N/A	40.9113	0.9895
Barbara	Bridge	25.2321	0.7337	35.5629	0.9783	36.1070	N/A	40.9072	0.9973
Average		24.0488	0.6913	35.2058	0.9584	35.8692	N/A	40.9170	0.9871

Finally, we compare the encryption and decryption efficiencies between our proposed scheme and schemes introduced in [19, 24, 25, 41, 42]. The images "Finger" and "Baboon" with size 256×256 are used as the plain images. Table 8 shows the encryption and decryption times of different encryption schemes, respectively. The results show that the encryption and decryption speeds of our proposed scheme are much faster than the other five schemes, indicating the high efficiency of the proposed scheme.

Table 8: Comparison results of encryption and decryption times (second) for images with size 256×256.

Schemes	Finger256		Baboon256		Average	
	Encryption	Decryption	Encryption	Decryption	Encryption	Decryption
[19]	N/A	N/A	0.4296	0.9544	0.4296	0.9544
[24]	0.1159	2.2295	0.1181	2.3325	0.1170	2.2760
[25]	0.1544	2.2489	0.1523	2.2870	0.1533	2.2679
[41]	0.4536	1.1374	0.4607	1.1413	0.4572	1.1393
[42]	0.3356	1.5216	0.3319	1.5342	0.3333	1.5279
Proposed	0.0629	0.3074	0.0618	0.3116	[illegible]	[illegible]

adopted

6. Conclusion

This paper proposed a visually secure image encryption scheme using adaptive sparsification and parallel compressive sensing. The scheme includes the encryption and embedding stages. The encryption stage first decomposes a plain image using SWT and scrambles the image using the 2D cat map, and then samples the scrambled image using PCS with a threshold for each column, and finally quantifies and diffuses the image to obtain a secret image. The embedding stage embeds the secret image into a carrier image using the matrix coding. The adaptive sparsification can greatly improve the quality of reconstructed image by utilizing the separable wavelet transform and column-based adaptive thresholds. The parallel CS with random-order measurement matrices is adapted to enhance the processing efficiency. Besides, the matrix encoding ensures can result in superior visual effect of the cipher image and doesn't affect the quality of reconstructed image. Experiment results demonstrate the high security and robustness of our proposed scheme. Comparison results show that our proposed scheme has better performance than some other latest scheme in the quality of reconstructed image, quality of cipher image, and efficiency.

also

References

[1] W. Liu, K. Sun, C. Zhu, A fast image encryption algorithm based on chaotic map, Optics and Lasers in Engineering 84 (2016) 26–36.

[2] C. Li, G. Luo, K. Qin, C. Li, An image encryption scheme based on chaotic tent map, Nonlinear Dynamics 87 (1) (2017) 127–133.

[3] C. Pak, L. Huang, A new color image encryption using combination of the 1D chaotic map, Signal Processing 138 (2017) 129–137.

[4] E. Y. Xie, C. Li, S. Yu, J. Lü, On the cryptanalysis of Fridrich's chaotic image encryption scheme, Signal processing 132 (2017) 150–154.

[5] P. Zhen, G. Zhao, L. Min, X. Jin, Chaos-based image encryption scheme combining DNA coding and entropy, Multimedia Tools and Applications 75 (11) (2016) 6303–6319.

[6] R. Guesmi, M. A. B. Farah, A. Kachouri, M. Samet, A novel chaos-based image encryption using DNA sequence operation and Secure Hash Algorithm SHA-2, Nonlinear Dynamics 83 (3) (2016) 1123–1136.

[7] X. Chai, Y. Chen, L. Broyde, A novel chaos-based image encryption algorithm using DNA sequence operations, Optics and Lasers in engineering 88 (2017) 197–213.

[8] L.-H. Gong, X.-T. He, S. Cheng, T.-X. Hua, N.-R. Zhou, Quantum image encryption algorithm based on quantum image XOR operations, International Journal of Theoretical Physics 55 (7) (2016) 3234–3250.

[9] N. Zhou, Y. Hu, L. Gong, G. Li, Quantum image encryption scheme with iterative generalized Arnold transforms and quantum image cycle shift operations, Quantum Information Processing 16 (6) (2017) 164.

[10] A. Y. Niyat, M. H. Moattar, M. N. Torshiz, Color image encryption based on hybrid hyper-chaotic system and cellular automata, Optics and Lasers in Engineering 90 (2017) 225–237.

[11] Y. Wang, Y. Zhao, Q. Zhou, Z. Lin, Image encryption using partitioned cellular automata, Neurocomputing 275 (2018) 1318–1332.

[12] Y. Luo, M. Du, J. Liu, A symmetrical image encryption scheme in wavelet and time domain, Communications in Nonlinear Science and Numerical Simulation 20 (2) (2015) 447 – 460.

[13] S. Liansheng, X. Meiting, T. Ailing, Multiple-image encryption based on phase mask multiplexing in fractional Fourier transform domain, Optics letters 38 (11) (2013) 1996–1998.

[14] Y. Zhang, Y. Li, W. Wen, Y. Wu, J.-x. Chen, Deciphering an image cipher based on 3-cell chaotic map and biological operations, Nonlinear Dynamics 82 (4) (2015) 1831–1837.

[15] E. Y. Xie, C. Li, S. Yu, J. Lü, On the cryptanalysis of Fridrich's chaotic image encryption scheme, Signal processing 132 (2017) 150–154.

[16] J. Chen, F. Han, W. Qian, Y.-D. Yao, Z.-l. Zhu, Cryptanalysis and improvement in an image encryption scheme using combination of the 1D chaotic map, Nonlinear Dynamics 93 (4) (2018) 2399–2413.

[17] H. Wang, D. Xiao, X. Chen, H. Huang, Cryptanalysis and enhancements of image encryption using combination of the 1D chaotic map, Signal processing 144 (2018) 444–452.

[18] J. Chen, Y. Zhang, L. Qi, C. Fu, L. Xu, Exploiting chaos-based compressed sensing and cryptographic algorithm for image encryption and compression, Optics & Laser Technology 99 (2018) 238–248.

[19] Y. Luo, J. Lin, J. Liu, D. Wei, L. Cao, R. Zhou, Y. Cao, X. Ding, A robust image encryption algorithm based on Chua's circuit and compressive sensing, Signal Processing 161 (Nonlinear Dyn 83 4 2016) (2019) 227–247.

[20] R. Fay, C. Ruland, Compressive sensing encryption modes and their security, in: 2016 11th International Conference for Internet Technology and Secured Transactions (ICITST), IEEE, 2016, pp. 119–126.

[21] A. Kanso, M. Ghebleh, An algorithm for encryption of secret images into meaningful images, Optics and lasers in engineering 90 (2017) 196–208.

[22] W. Wen, Y. Hong, Y. Fang, M. Li, M. Li, A visually secure image encryption scheme based on semi-tensor product compressed sensing, Signal Processing 173 (2020) 107580.

[23] L. Bao, Y. Zhou, Image encryption: Generating visually meaningful encrypted images, Information Sciences 324 (2015) 197–207.

[24] X. Chai, Z. Gan, Y. Chen, Y. Zhang, A visually secure image encryption scheme based on compressive sensing, Signal Processing 134 (2017) 35–51.

[25] H. Wang, D. Xiao, M. Li, Y. Xiang, X. Li, A visually secure image encryption scheme based on parallel compressive sensing, Signal Processing 155 (IEEE Transactions on Emerging Topics in Computing 1 1 2013) (2019) 218–232.

[26] H. Wang, J. Vieira, 2-D wavelet transforms in the form of matrices and application in compressed sensing, 2010 8th World Congress on Intelligent Control and Automation (2010) 35–39.

[27] G. Chen, Y. Mao, C. K. Chui, A symmetric image encryption scheme based on 3D chaotic cat maps, Chaos, Solitons & Fractals 21 (3) (2004) 749–761.

[28] D. L. Donoho, Compressed sensing, IEEE Transactions on Information Theory 52 (4) (2006) 1289–1306.

[29] E. J. Candes, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information, IEEE Transactions on Information Theory 52 (2) (2006) 489–509.

[30] J. A. Tropp, A. C. Gilbert, Signal recovery from random measurements via orthogonal matching pursuit, IEEE Transactions on information theory 53 (12) (2007) 4655–4666.

[31] G. H. Mohimani, M. Babaie-Zadeh, C. Jutten, Fast sparse representation based on smoothed ℓ^0 norm, in: International Conference on Independent Component Analysis and Signal Separation, Springer, 2007, pp. 389–396.

[32] P. Ping, J. Fu, Y. Mao, F. Xu, J. Gao, Meaningful Encryption: Generating Visually Meaningful Encrypted Images by Compressive Sensing

and Reversible Color Transformation, IEEE Access 7 (2019) 170168–170184.

[33] J. E. Fowler, S. Mun, E. W. Tramel, Block-Based Compressed Sensing of Images and Video, Foundations and Trends® in Signal Processing 4 (4) (2012) 297–416.

[34] Z. Hua, Y. Zhou, Image encryption using 2D Logistic-adjusted-Sine map, Information Sciences 339 (2016) 237–253.

[35] R. Crandall, Some notes on steganography, Posted on steganography mailing list (1998) 1–6.

[36] J. Korhonen, J. You, Peak signal-to-noise ratio revisited: Is simple beautiful?, in: 2012 Fourth International Workshop on Quality of Multimedia Experience, IEEE, 2012, pp. 37–38.

[37] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE transactions on image processing 13 (4) (2004) 600–612.

[38] G. Alvarez, S. Li, Some basic cryptographic requirements for chaos-based cryptosystems, International journal of bifurcation and chaos 16 (08) (2006) 2129–2151.

[39] Y. Wu, J. P. Noonan, S. Agaian, et al., NPCR and UACI randomness tests for image encryption, Cyber journals: multidisciplinary journals in science and technology, Journal of Selected Areas in Telecommunications (JSAT) 1 (2) (2011) 31–38.

[40] M. J. Swain, D. H. Ballard, Color indexing, International journal of computer vision 7 (1) (1991) 11–32.

[41] X. Chai, X. Zheng, Z. Gan, D. Han, Y. Chen, An image encryption algorithm based on chaotic system and compressive sensing, Signal Processing 148 (2018) 124–144.

[42] X. Chai, H. Wu, Z. Gan, Y. Zhang, Y. Chen, K. W. Nixon, An efficient visually meaningful image compression and encryption scheme based on compressive sensing and dynamic LSB embedding, Optics and Lasers in Engineering 124 (2020) 105837.

[43] Z. Gan, X. Chai, J. Zhang, Y. Zhang, Y. Chen, An effective image compression–encryption scheme based on compressive sensing (CS) and game of life (GOL), Neural Computing and Applications (2020) 1–29.