A Novel Image Encryption Algorithm Based on Plaintext-related Hybrid Modulation Map

Mingzhe Liu[1], Feixiang Zhao[1], Xin Jiang[1], Xianghe Liu[1], Yining Liu[2]
[1] State Key Laboratory of Geohazard Prevention and Geoenvironment Protection,
Chengdu University of Technology, China
[2] Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, China
liumz@cdut.edu.cn, zhaofeixiang@cdut.edu.cn, jiangxin@cdut.edu.cn, liuxianghe@cdut.edu.cn, ynliu@guet.edu.cn

Abstract

Derived from 1D-Tent map and 1D-Logistic map, a plaintext-related hybrid modulation map (PHMM) which modulated by variance contribution rate of singular values of plaintext image is proposed in this paper. The performance of PHMM was verified by the bifurcation diagram, Lyapunov exponent (LE) and Spearman correlation test. Based on PHMM, a pixel scrambling method called non-repetitive chaotic displacement (NRCD) is proposed. Then a novel image encryption algorithm is proposed based on the combination of NRCD and bit-plane reconstruction. In this algorithm, the permutation and diffusion processes are completed simultaneously. This algorithm has better performance in the tests of statistical characteristics, sensitivity, secret key and efficiency.

Keywords: Image encryption, Singular value decomposition, Chaotic map, Permutation, Diffusion

1 Introduction

Image encryption transmission on the Internet has been recognized as an effective way to protect the data privacy and verification. There are many kinds of algorithms in the field of image encryption. The main tools used by these algorithms include chaotic map [1-5, 11-13, 16-17, 19-23, 27-28], DNA computing [14-15, 26], cellular automata [6-7], wavelet transmission [8-10, 29], neural networks [30-32] and compressive sensing [33-34, 37].

The extreme initial value sensitivity and high randomness of chaotic system make chaotic map the most popular tool in digital image encryption algorithms. Fridrich [1] proposed the first image encryption algorithm based on chaotic map in 1998. After that, a large number of digital image encryption algorithms based on chaotic maps were proposed [1-5, 11-13, 16-17, 19-23, 27-28].

Compared to high-dimensional (HD) chaotic maps, one-dimensional (1D) chaotic maps are easier to detect because they have fewer parameters and state values [18]. Algorithms that use 1D chaotic maps [16-17] are easier to crack than algorithms that use HD chaotic maps [1-5, 19-23].

The chaotic sequences generated by HD chaotic maps used in existing algorithms are only related to the initial values and system parameters, and are independent of the plaintext images [1-5, 11-13, 16-17, 19-23]. Derived from 1D-Tent map and 1D-Logistic map, plaintext-related hybrid modulation map (PHMM) which modulated by variance contribution rate of singular values of plaintext image is proposed in this paper. Driven by the same secret key pair, the chaotic sequence generated by this map is different for different plaintext images. This feature greatly enhances security. Meanwhile, based on PHMM, a pixel position scrambling method called non-repetitive chaotic displacement (NRCD) is proposed. With NRCD, all pixel positions will be changed efficiently after only one round of permutation process.

In traditional image encryption algorithms, permutation and diffusion processes are often independent of each other, and in order to make the algorithm having plaintext-related property, diffusion process often involves a large number of *XOR* operations and floating-point operations [19-20, 24]. These reduce the execution efficiency of algorithms to a certain extent. In the proposed algorithm, the permutation process and the diffusion process are highly coupled by the combined use of NRCD and bit-plane reconstruction, meanwhile the *XOR* operation and the floating-point operation are avoided under the premise of ensuring the plaintext related property. Therefore, the algorithm has higher execution efficiency compared to other algorithms.

The rest of the article is structured as follows. The proposed chaotic map and its chaotic behavior is introduced in Section 2. Detailed steps of the proposed image encryption algorithm are presented in Section 3. The performance analysis of the algorithm are

[*] Corresponding Author: Mingzhe Liu; E-mail: liumz@cdut.edu.cn

DOI: 10.3966/160792642019122007012

presented in Section 4. Section 5 presents the extended encryption algorithms for binary image and color image. The final section concludes this paper.

2 Plaintext-related Hybrid Modulation Map (PHMM)

2.1 1D-Tent Map and 1D-Logistic Map

1D-Tent [35] and 1D-Logistic [36] are two classic one-dimensional chaotic maps, and they are widely used in many fields due to their good ergodic properties. 1D-Tent map is given to the formula (1), 1D-Logistic map is given to the formula (2).

$$x_{n+1} = \begin{cases} \mu x_n, & x_n < 0.5 \\ \mu(1-x_n), & 0.5 \le x_n \end{cases} \tag{1}$$

$$x_{n+1} = q x_n (1 - x_n), 3.567 \le q \le 4 \tag{2}$$

$\{x_0, x_1, x_2, \ldots\}$ is the generated chaotic sequence. μ and q is the system parameters.

2.2 Plaintext-related Hybrid Modulation Map

Based on the high coupling of 1D-Tent map and 1D-Logistic map, a new 2D chaotic map, PHMM is proposed. In PHMM, the plaintext-related features are implemented by timely addition of plaintext-related information. The equation is shown in formula (3).

$$\begin{cases} x_{n+1} = \begin{cases} \mu(x_n + \alpha|\sin y_n|)(pris_{\text{mod}(n,M)+1} + 0.2), & x_n < 0.5 \\ \mu(1 - x_n + \alpha|\sin y_n|)(pris_{\text{mod}(n,M)+1} + 0.2), & 0.5 \le x_n \end{cases} \\ y_{n+1} = q(y_n + \alpha|\sin x_{n+1}|)(1 - y_n - \alpha|\sin x_{n+1}|)(pris_{\text{mod}(n,M)+1} + 0.2) \end{cases} \tag{3}$$

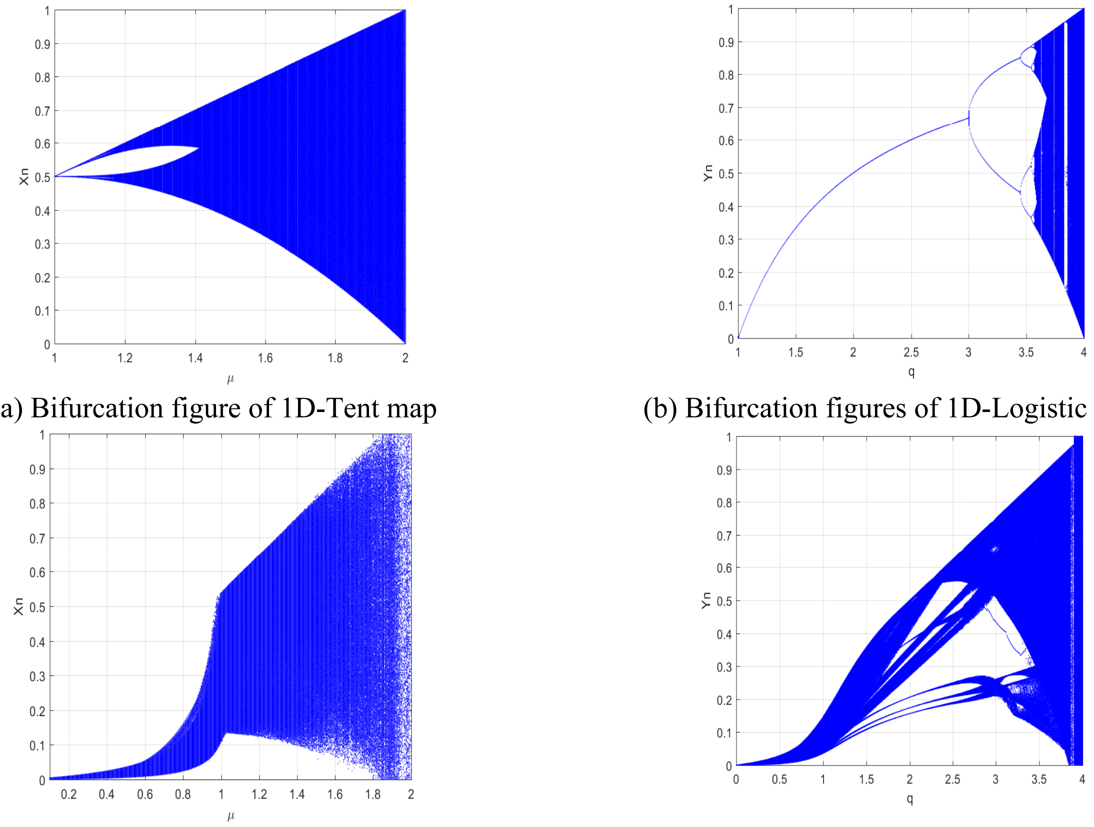
Where $pris_n$ indicates the n-*th* element in the plaintext-related information set ($PRIS \in R^M$), which is called an impurity parameter. α is the coupling parameter. μ and q are system parameters. By introducing more parameters, PHMM achieves better chaotic characteristics.

2.3 Comparison

The above three maps were compared by bifurcation diagram, Lyapunov exponent (LE) and Spearman correlation test. The x dimension of PHMM is compared with 1D-Tent map, and the y dimension is compared with 1D-Logistic map.

2.3.1 Bifurcation Diagram and Lyapunov Exponent

Through the bifurcation diagram, the chaotic region of the chaotic map can be roughly determined. In order to qualitatively measure the chaotic performance of PHMM, the bifurcation diagram of PHMM and the bifurcation diagram of two basic 1D maps are shown in Figure 1. It is obvious that PHMM expands the chaotic region in two dimensions.



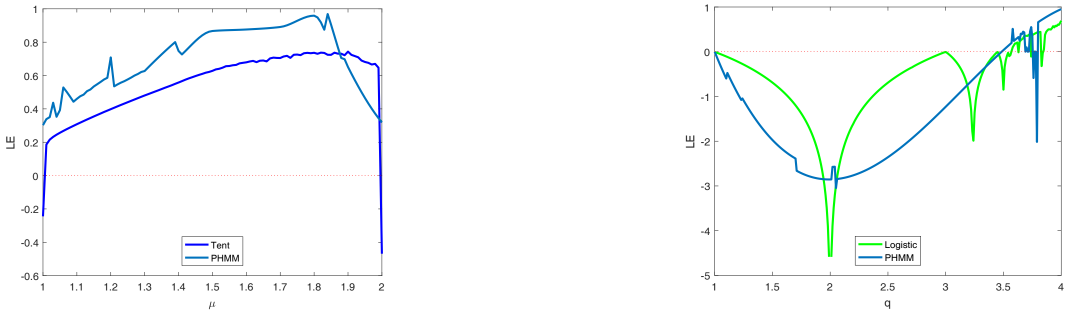
(a) Bifurcation figure of 1D-Tent map

(b) Bifurcation figures of 1D-Logistic map

(c)Bifurcation figure of x_n in PHMM $(\alpha = 0.05, q = 4)$ (d) Bifurcation figure of y_n in PHMM $(\alpha = 0.05, \mu = 1.5)$

Figure 1.

A positive LE indicates that the system is in a chaotic state at this time. In order to quantitatively measure the chaotic performance of PHMM, the LE spectrum of PHMM and the LE spectrums of two basic 1D maps were compared respectively. The comparison results are shown in Figure 2. The range of parameter values μ and q that bring the system into a chaotic state is increased. Therefore, PHMM has better chaotic performance in two dimensions than the two basic chaotic maps, respectively.



(a) Lyapunov exponent spectrums of PHMM $(\alpha = 0.05, q = 4)$ and 1D-Tent map

(b) Lyapunov exponent spectrums of PHMM $(\alpha = 0.05, \mu = 1.5)$ and 1D-Logistic map

Figure 2.

2.3.2 Spearman Correlation Test

In statistics, the Spearman correlation coefficient is a commonly used nonparametric indicator that measures the dependence of two sequences. It uses a monotonic equation to evaluate the correlation of two statistical variables. Given two sequences $X = \{x_0, x_1, x_2, \dots, x_N\}$ and $Y = \{y_0, y_1, \dots, y_N\}$, its calculation formula is given to the formula (4).

$$\rho = 1 - \frac{6\Sigma d_i^2}{n(n^2 - 1)} \tag{4}$$

Where $d_i = label1_i - label2_i$. $label1_i$ is the position of x_i in X when X is arranged in ascending order and $label2_i$ is the position of y_i in Y when Y is arranged in ascending order. The larger $|\rho|$, the greater the difference between the two sequences.

Extreme initial state sensitivity is one of the characteristics of chaotic map. In order to quantitatively measure this characteristic, this paper introduces the Spearman correlation test. The test method is: set a set of initial parameters and obtain a chaotic sequence, then add a very small increment to one of the parameters and then obtain another chaotic sequence. Finally, the correlation coefficient of two chaotic sequences is obtained by Spearman correlation test. The results of the comparison are listed in Table 1. Clearly, PHMM has a more pronounced initial state sensitivity than two basic one-dimensional maps.

Table 1. Spearman correlation coefficients of chaotic maps

	Tent Map μ =1.9000 $\Delta\mu$ =0.0001	PHMM μ =1.9000 $\Delta\mu$ =0.0001	Logistic Map q =3.9000 Δq =0.0001	PHMM =3.9000 Δq =0.0001
$\lvert\rho\rvert$	0.0071	0.0133	0.0096	0.0170

3 The Proposed Image Encryption and Decryption Algorithm

3.1 Key Structure

The secret key used in the proposed algorithm are shown in Figure 3. It is a 320-bit sequence that is used to generate two initial values (x_0, y_0), couple parameter α , and system parameters (μ, q) for the PHMM. These five parameters are decimals generated by different bits in the sequence by the IEEE standard 754.

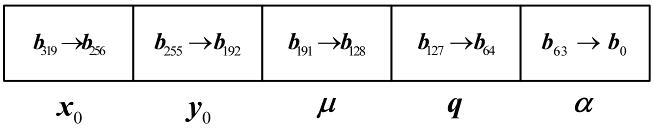

$b_{319} \rightarrow b_{256}$
$b_{255} \rightarrow b_{192}$
$b_{191} \rightarrow b_{128}$
$b_{127} \rightarrow b_{64}$
$b_{63} \rightarrow b_0$
x_0
y_0
μ
q
α

Figure 3. Secret key structure

3.2 Non-repetitive Chaotic Displacement (NRCD)

In order to destroy the high correlation between adjacent pixels in the original image, scrambling is an indispensable part of the image encryption algorithm. Images that are scrambled by traditional scrambling processes often have some pixels that have not changed in position, and this situation reduces the anti-decipherability of the encrypted image to a certain extent. Based on PHMM, a scrambling algorithm, non-repetitive chaotic displacement (NRCD) is proposed. This algorithm can change the position of all the pixels after one round of scrambling.

Given an image O to be encrypted, set its size to M×N. Set a series of parameters and initial values then drive PHMM to generate two chaotic sequences $X=\{x_0, x_1, \ldots, x_M\}$ and $Y=\{y_0, y_1, \ldots, y_N\}$. The detailed operational flow of NRCD is shown in Algorithm 1. The inverse process of NRCD is shown in Algorithm 2.

3.3 Bit-plane Reconstruction

The grayscale image has eight bit planes (from B_0 to B_7), and the pixel value of the i-th row and the j-th column can be written as:

$$p(i, j)=\sum_{n=0}^{7} B_n(i, j)\times 2^n \quad \textbf{(5)}$$

The formula for the proportion of the image information contained in n-th bit plane is as follows:

$$weight_n=\frac{2^n}{255}\times 100\%, n=0,1,\ldots,7 \quad \textbf{(6)}$$

It can be clearly seen that the proportion of each bit plane from B_0 to B_7 in the original image increases exponentially. For example, B_7 accounts for more than 50% and B_0 does not exceed 0.5%. Therefore, proper rearrangement of bit planes makes it easy to hide most of the information of the image. In this paper, the way of bit-planes reconstruction is shown in Figure 4.

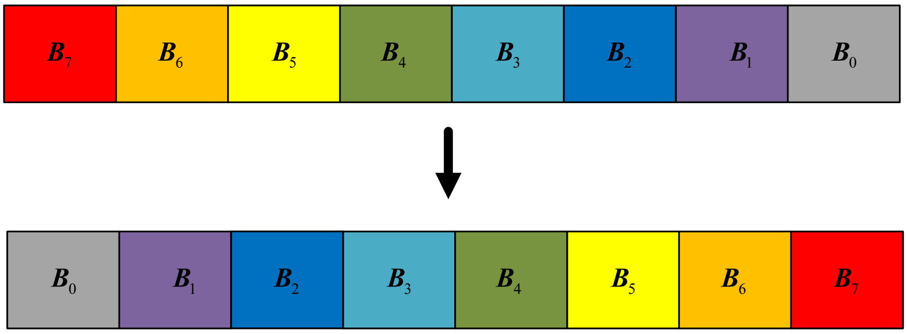


Figure 4. Diagram of bit-planes reconstruction

3.4 Image Encryption and Decryption Algorithm

3.4.1 Encryption Algorithm

In the NRCD-based image encryption algorithm proposed in this paper, the image is processed by bit-plane reconstruction and split into two subgraphs. The two subgraphs are scrambled by the NRCD, respectively, and then the encrypted image are synthesized by simple addition. Through such process, the scramble process and the diffusion process independently of each other in the traditional image encryption algorithm achieve a high degree of coupling.

Given an image to be encrypted with the size of M × N, the flowchart of the encryption algorithm is shown in Figure 5.

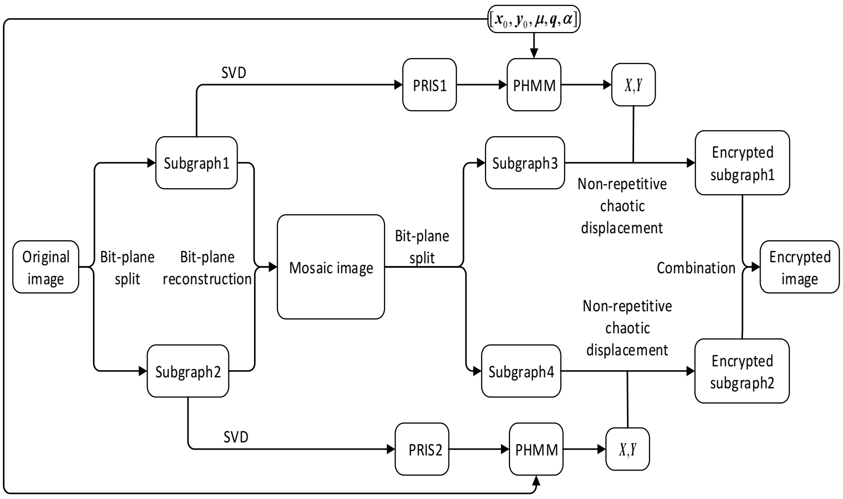


Figure 5. Flowchart of encryption

The detailed steps are given as follows:

Step 1. Set $[x_0, y_0, q, \mu, \alpha]$;

Step 2. Split the original image (consisting of $B_0, B_1, ..., B_7$) into two subgraphs, Subgraph1 and Subgraph2. Where $Subgraph\,1 = B_7 \times 2^7 + B_6 \times 2^6 + B_5 \times 2^5 + B_4 \times 2^4$ and $Subgraph\,2 = B_3 \times 2^3 + B_2 \times 2^2 + B_1 \times 2^1 + B_0 \times 2^0$;

Step 3. Perform singular value decomposition (SVD) on Subgraph1, and then solve the variance contribution rate of each singular value. All variance contribution rates are reserved in four decimal places and arranged in descending order, then constitute vector $PRIS\,1 = [pris\,1_0, pris\,1_1, ..., pris\,1_{\min(M,N)}]$;

Step 4. Perform singular value decomposition (SVD) on Subgraph2, and then solve the variance contribution rate of each singular value. All variance contribution rates are reserved in four decimal places and arranged in descending order, then constitute vector $PRIS\,2 = [pris\,2_0, pris\,2_1, ..., pris\,2_{\min(M,N)}]$;

Step 5. Reconstruct to B_0 to B_7 generate Mosaic image;

Step 6. Split the Mosaic image into two subgraphs, Subgraph3 and Subgraph4. Where $Subgraph\,3 = B_0 \times 2^7 + B_1 \times 2^6 + B_2 \times 2^5 + B_3 \times 2^4$ and $Subgraph\,4 = B_4 \times 2^3 + B_5 \times 2^2 + B_6 \times 2^1 + B_7 \times 2^0$;

Step 7. Driven by $[x_0, y_0, q, \mu, \alpha]$ and $PRIS\,1$, PHMM generates two chaotic sequences. These two chaotic sequences are combined with Subgraph3 as inputs to t Algorithm 1, and then an encrypted subgraph is generated;

Step 8. Driven by $[x_0, y_0, q, \mu, \alpha]$ and $PRIS\,2$, PHMM generates two chaotic sequences. These two chaotic sequences are combined with Subgraph4 as inputs to Algorithm 1, and then another encrypted subgraph is generated;

Step 9. Adding the encrypted subgraph generated in step 7 to the encrypted subgraph generated in step 8 derectly, and then the encrypted image is generated.

Algorithm 1. Non-repetitive chaotic displacement algorithm

```
Input: Original image matrix $O \in R^{M \times N}$, and chaotic sequences $X$ and $Y$;
Output: Encrypted image matrix $E \in R^{M \times N}$;
Get the index of each element (from $x_0$ to $x_M$) in $X$ when this sequence is sorted in descending order and compose these indexes into a vector
    $LABEL\,1 = \{label\,1_0, label\,1_1, ..., label\,1_M\}$;
Get the index of each element (from $y_0$ to $y_N$) in $Y$ when this sequence is sorted in descending order and compose these indexes into a vector
    $LABEL\,2 = \{label\,2_0, label\,2_1, ..., label\,2_N\}$;
for $i$ = 1 to $M$ do
    if $i = label\,1_i$
        Exchange positions of $label\,1_i$ and $label\,1_{i+1}$;
    end if
    $E(label\,1_i, :) = O(i, :)$;
end for
for $i$ = 1 to $N$ do
    if $i = label\,2_i$
        Exchange positions of $label\,2_i$ and $label\,2_{i+1}$;
    end if
    $E(:, label\,2_i) = O(:, i)$;
end for
```

3.4.2 Decryption Algorithm

The secret image can be correctly decrypted only when the decrypter obtains $[x_0, y_0, q, \mu, \alpha]$ and vector $PRISx\,(x = 1, 2)$ for the encrypted image. The flowchart of the decryption is shown in Figure 6. The detailed decryption process is given as follows:

Step 1. The encrypted image (consisting of $E_0, E_1, \ldots, E_7$) is split into two subgraphs: Encrypted subgraph1 and Encrypted subgraph2. Where $Encrypted\ subgraph\,1 = E_7 \times 2^7 + E_6 \times 2^6 + E_5 \times 2^5 + E_4 \times 2^4$ and $Encrypted\ subgraph\,2 = E_3 \times 2^3 + E_2 \times 2^2 + E_1 \times 2^1 + E_0 \times 2^0$;

Step 2. Driven by $[x_0, y_0, q, \mu, \alpha]$; and $PRIS1$, PHMM generates two chaotic sequences. These two chaotic sequences are combined with *Encrypted subgraph*1 as inputs to Algorithm 2, and then Subgraph3 is generated;

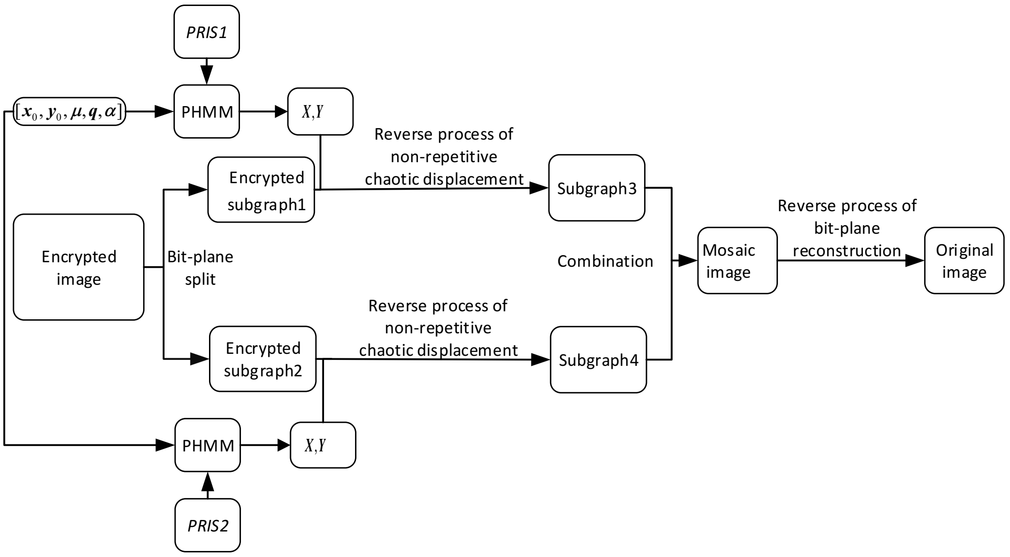

PRIS1
$[x_0, y_0, \mu, q, \alpha]$
PHMM
X,Y
Reverse process of non-repetitive chaotic displacement
Encrypted subgraph1
Subgraph3
Encrypted image
Bit-plane split
Combination
Mosaic image
Reverse process of bit-plane reconstruction
Original image
Reverse process of non-repetitive chaotic displacement
Encrypted subgraph2
Subgraph4
PHMM
X,Y
PRIS2

Figure 6. Flowchart of decryption

Algorithm 2. Inverse process of non-repetitive chaotic displacement algorithm

```
Input: Encrypted image matrix $E \in R^{M \times N}$, and chaotic sequences $X$ and $Y$;
Output: Original image matrix $O \in R^{M \times N}$;
Get the index of each element (from $x_0$ to $x_M$) in $X$ when this sequence is sorted in descending order and
    compose these indexes into a vector
    $LABEL1 = \{label1_0, label1_1, ..., label1_M\}$;
Get the index of each element (from $y_0$ to $y_N$) in $Y$ when this sequence is sorted in descending order and
    compose these indexes into a vector
    $LABEL2 = \{label2_0, label2_1, ..., label2_N\}$;
for $i = 1$ to $M$ do
    if $i = label1_i$
        Exchange positions of $label1_i$ and $label1_{i+1}$;
    end if
    $O(i, :) = E(label1_{i+1}, :)$;
end for
for $i = 1$ to $N$ do
    if $i = label2_i$
        Exchange positions of $label2_i$ and $label2_{i+1}$;
    end if
    $O(:, i) = E(:, label2_i)$;
end for
```

Step 3. Driven by $[x_0, y_0, q, \mu, \alpha]$ and *PRIS*2, PHMM generates two chaotic sequences. These two chaotic sequences are combined with *Encrypted subgraph*2 as inputs to Algorithm 2, and then Subgraph4 is generated;
Step 4. Subgraph3 and Subgraph4 are merged into Mosaic image by simple addition, and then the Mosaic image is converted into the plaintext image by the inverse process of bit-planes reconstruction.

4 Security and Efficiency Analysis

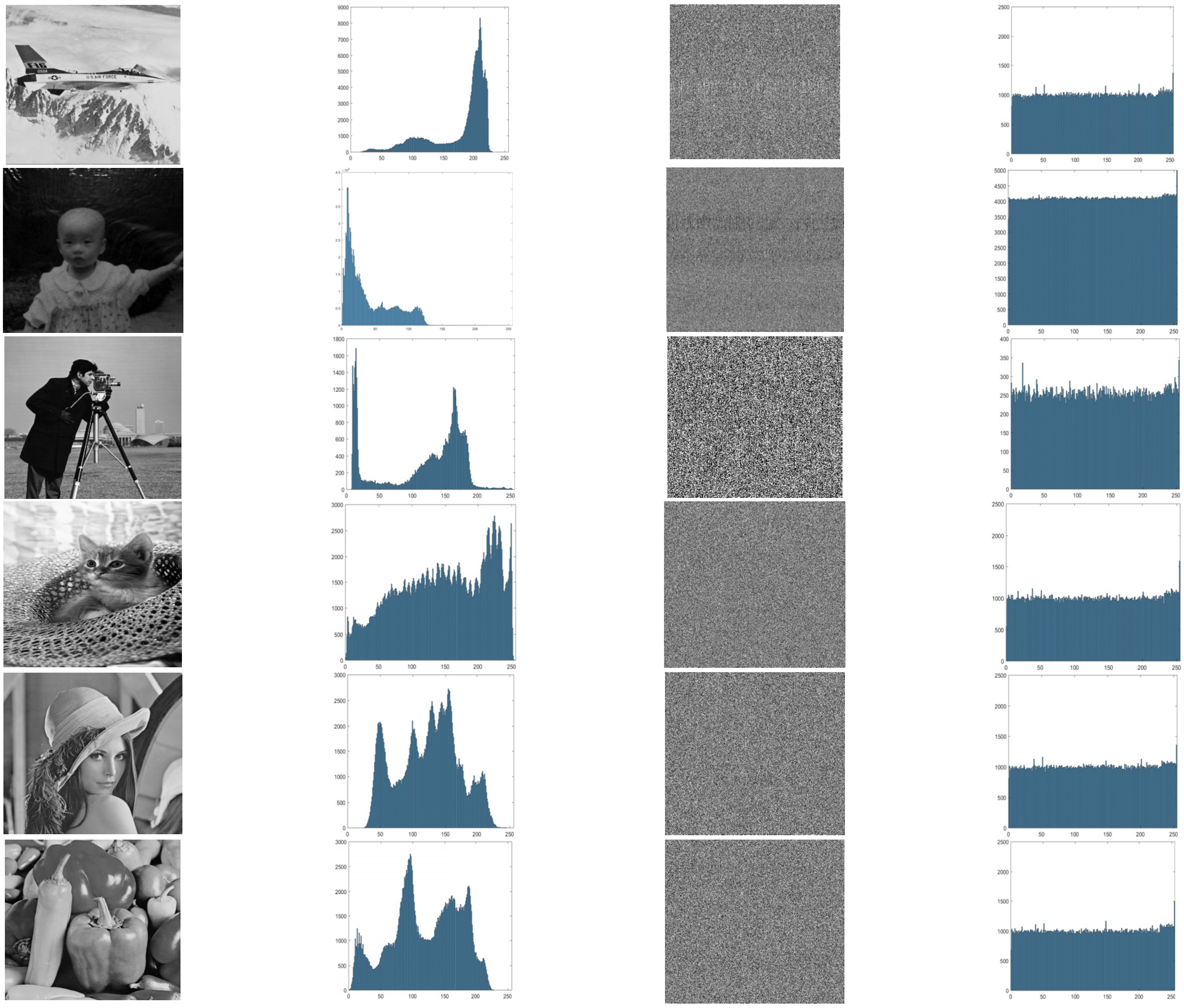
4.1 Statistical Analysis

In order to resist statistical attacks, an encrypted image generated by a good image encryption algorithm should approximate random noise. In this subsection, histogram, correlation coefficient, and information entropy are used to measure how close the encrypted image is to random noise.

4.1.1 Histogram

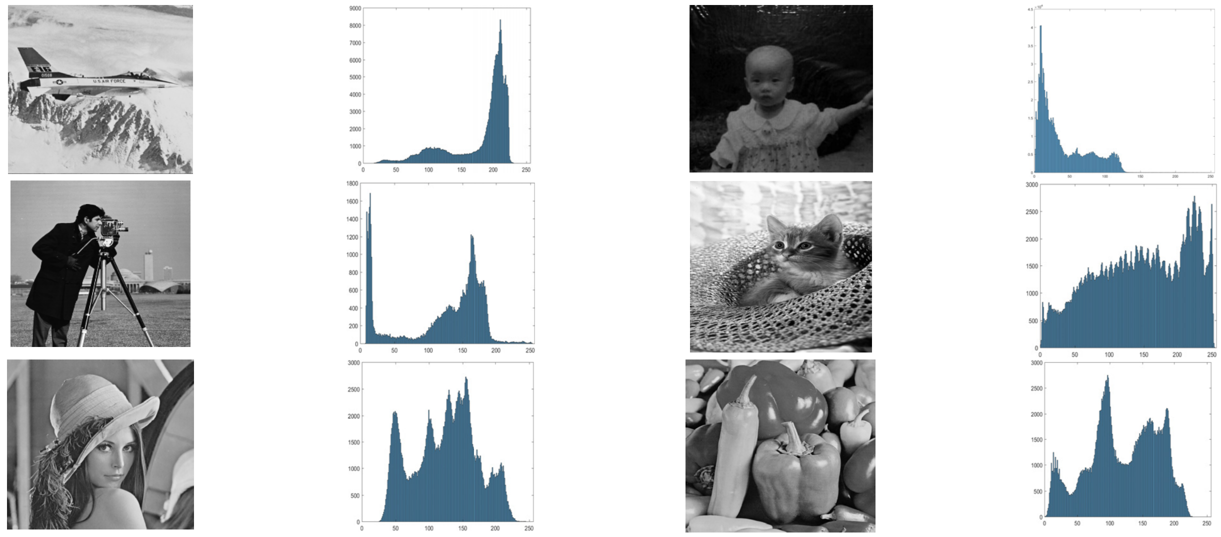
Pixels of random noise image are evenly distributed at [0, 255]. Therefore, an encrypted image generated by a good image encryption algorithm should also have this feature. Given secret key pair $[x_0, y_0, \mu, q, \alpha] =$

[0.2000, 0.3000, 1.8800, 3.8800, 0.0050], Figure 7 shows histograms of several plaintext images and their corresponding ciphertext images. Figure 8 illustrates decrypted images and their corresponding histograms.



(a) Plaintext images (b) Histograms of (a) (c) Encrypted images (d) Histograms of (c)

Figure 7. Encryption results



(a) Decrypted images (b) Histograms of (a) (c) Decrypted images (d) Histograms of (c)

Figure 8. Decryption results

Intuitively, the histogram of the plaintext image has a tortuous outline, while the ciphertext image has a flat histogram.

In order to quantitatively measure the probability that a ciphertext image obeys uniform distribution, chi-squared test is used for quantitative analysis.

Given a set of observed frequency distribution $o_i, i = 0, 1, ..., n$ and, assume that its theoretical frequency distribution is $t_i, i = 0, 1, ..., n$. To verify whether the assumption is true, the Pearson chi-squared test shown in formula (7) is often used.

$$\chi^2 = \sum_{i=0}^{n} \frac{(o_i - t_i)^2}{t_i} \quad \textbf{(7)}$$

For a grayscale image with n gray scales and a size of $M \times N$, assume that the pixels frequency $o_i, i = 0, 1, ..., n$ of i-th grayscale value obeys uniform distribution, that is, $t_i = t = M \times N / n$. Then formula (7) can be rewriten as formula (8).

$$\chi^2 = \sum_{i=0}^{n} \frac{(o_i - t)^2}{t} \quad \textbf{(8)}$$

In order to verify the reliability of the hypothesis, a small significance level α is given. As shown in formula (9), when $\chi^2 < \chi^2_\alpha(n)$, o_i has a great probability to obey the uniform distribution.

$$P\{\chi^2 \geq \chi^2_\alpha(n)\} = \alpha \quad \textbf{(9)}$$

The grayscale value range of the encrypted image obtained by this algorithm is [0, 255], so n=255. Given significance level α=0.01. The chi-squared test was performed on each ciphertext image in Figure 7 and the results are shown in Table 2.

Table 2. Results of chi-squared test

Name	Airplane	Baby	Cameraman	Cat	Lena	Peppers
χ^2	211.0506	272.4293	181.0010	224.3544	230.1001	241.2689
$\frac{\chi^2}{\chi^2_{0.01}(255)}$	67.98%	87.75%	58.30%	72.27%	74.12%	77.71%

When setting the degree of freedom to 255, $\chi^2_{0.01}(255) = 310.46$. The Pearson Chi-squared statistic of each ciphertext image in Figure 7 is significantly smaller than $\chi^2_{0.01}(255)$. This means that the pixel values of all ciphertext images are uniformly distributed, that is, all ciphertext images can be considered as random noise images.

4.1.2 Pearson Correlation Coefficient

There should be no correlation between adjacent pixels in the ciphertext image in order to resist statistical attacks.

In this subsection, the Pearson correlation coefficient is used to quantitatively measure the correlation between adjacent pixels of a ciphertext image. 1500 pairs of adjacent pixels are randomly collected from the image in the horizontal direction, the vertical direction, and the diagonal direction, respectively, and then the correlation coefficients between adjacent pixels in the horizontal, vertical, and diagonal directions of the image are calculated. The mean values of the correlation coefficients of the multiple plaintext images shown in Figure 7 are shown in Table 3. Several typical chaotic image encryption algorithms [19-23] are compared with proposed algorithm. The mean values of the correlation coefficients corresponding to the ciphertext images are shown in Table 4.

Table 3. Mean of correlation coefficients of plaintext images in Figure 7

Direction	Horizontal	Vertical	Diagonal
Mean	0.9023	0.9430	0.9188

Table 4. Comparison of mean of correlation coefficients of encrypted images in Figure 7 (absolute value)

Direction	Horizontal	Vertical	Diagonal	Mean
Liu's [19]	0.0177	0.0089	0.0264	0.0177
Sheela's [20]	0.0102	0.0480	0.0118	0.0233
Hua's [21]	0.0066	0.0227	0.0180	0.0158
Zhou's [22]	0.0145	0.0107	0.0061	0.0104
Ye's [23]	0.0098	**0.0082**	0.0208	0.0129
Ours	**0.0042**	0.0093	**0.0057**	**0.0064**

The closer the absolute value of the correlation coefficient is to 0, the weaker the correlation between adjacent pixels. From Table 3 and Table 4, it's obviously that the correlation between adjacent pixels of plaintext image is strong, while it's weak in ciphertext image. The test results also sugest that the proposed algorithm is generally more resistant to statistical attacks than other algorithms.

4.1.3 Information Entropy

Information entropy reflects the randomness of image information. The larger the information entropy, the less the visual information of the image. Its calculation formula is as follows:

$$H = -\sum_{i=0}^{L} p(i) \log_2^{p(i)} \quad \textbf{(10)}$$

Where L is the number of gray scales of the image, and $p(i)$ is the probability that the i-th gray scale value appears. When L = 255, the theoretical value of H is 8. The information entropy of the plurality of encrypted images in Figure 7 is shown in Table 5. The test results show that compared to other algorithms, the encrypted images generated by proposed algorithm are most similar to random noise.

Table 5. Information entropy of encrypted images

Name	Airplane	Baby	Cameraman	Cat	Lena	Peppers	Mean
Liu et al.'s [19]	7.9923	7.9975	**7.9991**	7.9930	7.9941	7.9918	7.9946
Sheela et al.'s [20]	7.9932	7.9912	7.9940	7.9866	7.9895	7.9934	7.9913
Hua and Zhou's [21]	7.9915	7.9927	7.9930	7.9925	7.9907	7.9950	7.9927
Zhou et al.'s [22]	**7.9942**	7.9937	7.9982	7.9951	7.9943	7.9942	7.9950
Ye's [23]	7.9801	7.9972	7.9944	7.9968	**7.9954**	**7.9961**	7.9933
Ours	7.9918	**7.9986**	7.9983	**7.9995**	7.9925	7.9960	**7.9961**

4.2 Secret Key Analysis

In order to effectively resist brute force attack, image encryption algorithms should have a large enough key space and sensitivity to keys. The key sequence of the algorithm proposed in this paper is 320 bits, so its key space is 2^{320}. At the same time, considering that $PRIS_1$ and $PRIS_2$ extracted from the plaintext image also affect the encryption process. Therefore, according to [25], the algorithm proposed in this paper is capable of resisting brute force attack.

Two indicators, number of pixels change rate (NPCR) and unified average changing intensity (UACI), are used for quantitative analysis of secret key sensitivity. The definition of NPCR and UACI between two images P and P' are as follows:

$$NPCR(P, P') = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} D(P_{i,j} - P'_{i,j}) \times 100\% \quad \textbf{(11)}$$

where $D(x) = \begin{cases} 0, x = 0 \\ 1, x \neq 0 \end{cases}$.

$$UACI = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{|P_{i,j} - P'_{i,j}|}{255} \times 100\% \quad \textbf{(12)}$$

For two completely unrelated digital images, the theoretical expected value of NPCR is 99.6904%, and UACI is 33.4635%. The closer these two indicators between two images are to the ideal value, the greater the difference between the two images.

Change b_{256}, b_{192}, b_{128}, b_{64} and b_0 in the original secret key sequence to $1-b_{256}$, $1-b_{192}$, $1-b_{128}$, $1-b_{64}$ and $1-b_0$, respectively. And then generate five new key pairs: $[x_0', y_0, \mu, q, \alpha]$, $[x_0, y_0', \mu, q, \alpha]$, $[x_0, y_0, \mu', q, \alpha]$, $[x_0, y_0, \mu, q', \alpha]$, and $[x_0, y_0, \mu, q, \alpha']$, respectively.

Take the six original images shown in Figure 7 as test materials, the mean values of NPCR and UACI between the ciphertext images generated based on the five new secret key pairs and the ciphertext image generated based on $[x_0, y_0, \mu, q, \alpha]$, are listed in Table 6.

Table 6. NPCR and UACI between the new ciphertext images and the original ciphertext image

Index Secret key pair	NPCR	UACI
$[x_0', y_0, \mu, q, \alpha]$	99.6190%	33.2817%
$[x_0, y_0', \mu, q, \alpha]$	98.7884%	33.2100%
$[x_0, y_0, \mu', q, \alpha]$	98.7528%	33.5035%
$[x_0, y_0, \mu, q', \alpha]$	99.4333%	33.4836%
$[x_0, y_0, \mu, q, \alpha']$	99.9244%	33.2844%

Obviously, a subtle change in the parameters of the secret key pair can result in a large difference between two ciphertext images corresponding to the same original image. Therefore, this algorithm is extremely sensitive to secret key.

4.3 Plaintext Image Sensitivity Analysis

Aim at resist selected plaintext attack and known plaintext attack, an image encryption system with excellent performance should be highly sensitive to plaintext images. That is, two ciphertext images generated by two plaintext images with slight differences should have significant differences.

In this subsection, the six plaintext images in Figure 7 are used to test the plaintext sensitivity of the proposed algorithm. For each plaintext image, 800 pixels are randomly selected and increment $\Delta = 1$ is added, then a new plaintext image is generated. After the process is repeated 50 times, a total of 50 new plaintext images will be newly generated for each plaintext image. The mean values of NPCR and UACI between the ciphertext images generated by the new 50 plaintext images and the original ciphertext image are listed in Table 7.

Table 7. Mean of NPCR and UACI between new ciphertext images and original ciphertext image

Index Name	NPCR	UACI
Airplane	99.4785%	32.7286%
Baby	99.6032%	33.1020%
Cameraman	98.5254%	33.6732%
Cat	99.6110%	33.6835%
Lena	99.2268%	32.5605%
Peppers	98.4849%	33.6914%

Table 7 shows that there is a significant difference between the newly generated ciphertext images and the original ciphertext image. Therefore, the proposed algorithm has extreme sensitivity to plaintext images. It has excellent resistance to selective plaintext attacks and known plaintext attacks.

4.4 Execution Efficiency

The speed of encryption and decryption is an important indicator to measure the performance of an image encryption algorithm. In this paper, The efficiency of execution is calculated by this rule: assuming that the grayscale image size is $M \times N$, the execution efficiency is $M \times N \times 8$ bit/(encryption time+decryption time).

Taking the six plaintext diagrams shown in Figure 7 as test materials, the average speed of 50 encryption and decryption by this algorithm and some other algorithms [19-20, 21-23] are listed in Table 8.

Table 8. Comparison of efficiency

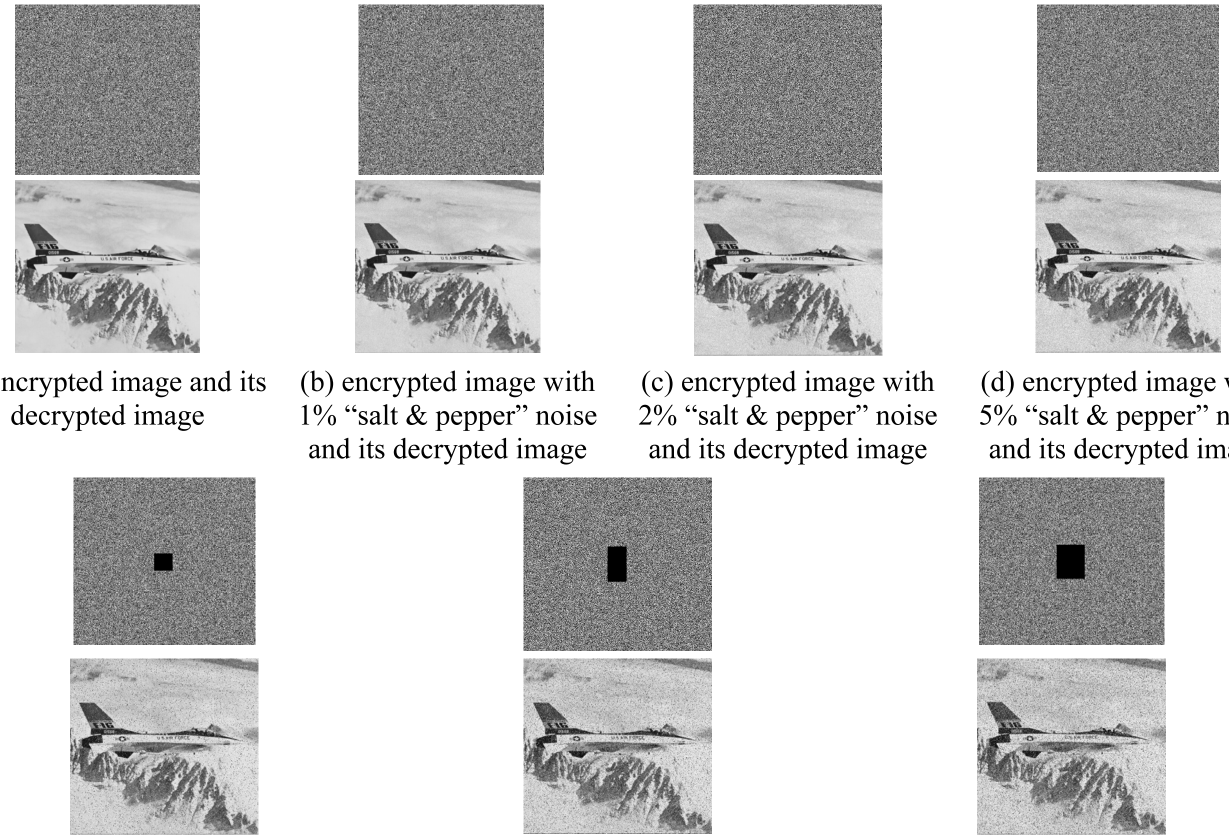
Algorithm Name	Liu et al.'s [19]	Sheela et al.'s [20]	Hua and Zhou's [21]	Zhou et al. 's [22]	Ye's [23]	Ours
Airplane	2.1668Mbps	0.0123Mbps	2.1961Mbps	1.8811Mbps	0.4326Mbps	2.3690Mbps
Baby	2.1630Mbps	0.0190Mbps	2.3144Mbps	1.7220Mbps	0.4103Mbps	2.3842Mbps
Cameraman	2.3816Mbps	0.0188Mbps	2.0025Mbps	2.0031Mbps	0.4185Mbps	2.2379Mbps
Cat	2.3912Mbps	0.0176Mbps	2.3242Mbps	2.0471Mbps	0.4057Mbps	2.1674Mbps
Lena	2.2408Mbps	0.0194Mbps	1.9591Mbps	1.9892Mbps	0.4033Mbps	2.1830Mbps
Peppers	2.3986Mbps	0.0217Mbps	2.1642Mbps	1.8599Mbps	0.4219Mbps	2.4049Mbps
Average	2.2903Mbps	0.0181Mbps	2.1601Mbps	1.9171Mbps	0.4154Mbps	**2.2911**Mbps

It should be pointed out that all test programs are run using Dev-C++ 5.9.2. The main configuration of the computer is: Windows 7 (64-bit), Core-i7 5820K (3.3 GHz) and 24G RAM.

It can be observed from Table 8 that the proposed algorithm has the highest execution efficiency. This is mainly because the algorithm only requires a small number of floating point operations and avoidance of iterative operations.

4.5 Noise Attack and Data Loss Attack Analysis

Encrypted images are often attacked by noise and data loss during transmission. In order to test the anti-noise attack ability of the proposed algorithm, the encrypted image is artificially added with different salt and pepper noise (noise density = 1%, 2%, 5%), the anti-noise attack capability is then obtained by observing the visibility of the decrypted image. Setting some pixels in the encrypted image to 0 (1%, 2% and 3% of the total number of pixels), and then observing the visibility of the decrypted image, the algorithm's anti-data loss attack capability is also tested. The test results are shown in Figure 9.



(a) encrypted image and its decrypted image

(b) encrypted image with 1% "salt & pepper" noise and its decrypted image

(c) encrypted image with 2% "salt & pepper" noise and its decrypted image

(d) encrypted image with 5% "salt & pepper" noise and its decrypted image

(e) encrypted image with 1% data loss and its decrypted image

(f) encrypted image with 2% data loss and its decrypted image

(g) encrypted image with 3% data loss and its decrypted image

Figure 9. Noise attack and data loss attack analysis

It can be seen from Figure 9(b) to Figure 9(d) that the salt and pepper noise causes some data in the encrypted image to be contaminated, which leads to a decrease in the visibility of the decrypted image, but the decryptor can still obtain most of the useful information from it. The same conclusion can also be obtained by Figure 9(e) to Figure 9(g). Therefore, the proposed algorithm has strong anti-noise attack capability and anti-data loss attack capability.

5 Extended Algorithms for Binary Image Encryption and RGB Image Encryption

Considering the widespread presence and frequent transmission of binary and color images on the Internet, it is also necessary to encrypt these two images. For this purpose, two extended versions of the image encryption scheme is shown in this section.

5.1 Extended Algorithm for Binary Image Encryption

Binary images have only one bit per pixel, so it is not feasible to encrypt them directly using the grayscale image encryption algorithm. In the proposed extended algorithm, the binary image is converted into a size-reduced grayscale image and then encrypted by the proposed grayscale image encryption algorithm. Given the original binary image $\boldsymbol{P} \in \boldsymbol{B}^{m \times n}, \boldsymbol{B} = \{\boldsymbol{0}, \boldsymbol{1}\}$, and the grayscale image obtained by the conversion is $\boldsymbol{P}_{gray} \in \boldsymbol{R}^{\left\lceil \frac{m}{2} \right\rceil \times \left\lceil \frac{m}{2} \right\rceil}$. The extended algorithm for binary image encryption is described in detail below.

Encryption:

Step 1: Converting binary image $\boldsymbol{P}$ to grayscale image $\boldsymbol{P}_{gray}$ by equation (13);

$$\begin{aligned} P_{gray}(\left\lceil \frac{i}{2} \right\rceil, \left\lceil \frac{j}{2} \right\rceil) = {} & P(i,j) \times 2^7 + [1 - P(i,j)] \times 2^6 \\ & + P(i,j+1) \times 2^5 + [1 - P(i,j+1)] \times 2^4 \\ & + P(i+1,j) \times 2^3 + [1 - P(i+1,j)] \times 2^2 \\ & + P(i+1,j+1) \times 2 + [1 - P(i+1,j+1)], \\ & (i = 1,3,5,...,M-2,\ j = 1,3,5,...,N-2) \end{aligned} \tag{13}$$

Step 2: Encrypting the $\boldsymbol{P}_{gray}$ according to the flow shown in Figure 5 to obtain the encrypted image $\boldsymbol{T} \in \boldsymbol{R}^{\left\lceil \frac{m}{2} \right\rceil \times \left\lceil \frac{m}{2} \right\rceil}$.

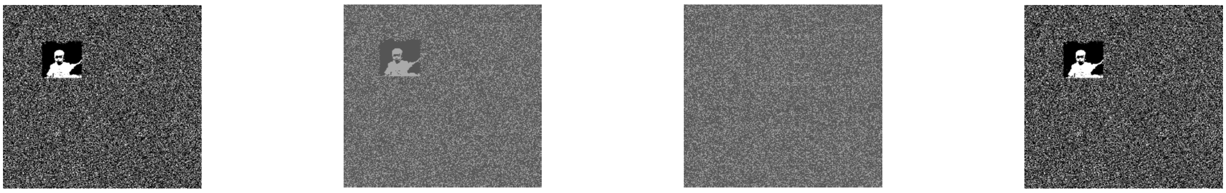
Decryption:

Step 1: Decrypting $\boldsymbol{T}$ to get $\boldsymbol{P}_{gray}$ according to the flow shown in Figure 6;

Step 2: Converting $\boldsymbol{P}_{gray}$ into binary image $\boldsymbol{P}$ by bitwise AND operation according to formula (14).

$$\begin{cases} \boldsymbol{P}(\boldsymbol{i},\boldsymbol{j}) = \boldsymbol{bitand}(\boldsymbol{P}_{gray}(\left\lceil \frac{i}{2} \right\rceil, \left\lceil \frac{j}{2} \right\rceil), \mathbf{10000000}) \\ \boldsymbol{P}(\boldsymbol{i},\boldsymbol{j}+\mathbf{1}) = \boldsymbol{bitand}(\boldsymbol{P}_{gray}(\left\lceil \frac{i}{2} \right\rceil, \left\lceil \frac{j}{2} \right\rceil), \mathbf{00100000}) \\ \boldsymbol{P}(\boldsymbol{i}+\mathbf{1},\boldsymbol{j}) = \boldsymbol{bitand}(\boldsymbol{P}_{gray}(\left\lceil \frac{i}{2} \right\rceil, \left\lceil \frac{j}{2} \right\rceil), \mathbf{00001000}) \\ \boldsymbol{P}(\boldsymbol{i}+\mathbf{1},\boldsymbol{j}+\mathbf{1}) = \boldsymbol{bitand}(\boldsymbol{P}_{gray}(\left\lceil \frac{i}{2} \right\rceil, \left\lceil \frac{j}{2} \right\rceil), \mathbf{00000010}) \end{cases} \tag{14}$$

The binary image (500×500) used for the test, the grayscale image (250×250) obtained by the binary image conversion, the encrypted image (250×250) and the decrypted binary image (500×500) are shown in Figure 10. The histogram corresponding to the grayscale image and the encrypted image is shown in Figure 11. The chi-square statistic, Pearson correlation coefficient, Information entropy, mean of NPCR and UACI (executing algorithm 50 times) of the encrypted image and execution efficiency is shown in Table 9. It can be seen from Table 9 that the algorithm has a high degree of security, and the operating efficiency of the algorithm is also taken into consideration.



(a) Binary image (b) Grayscale image (c) Encrypted image (d) Decrypted binary image

Figure 10. Binary image encryption/Decryption results

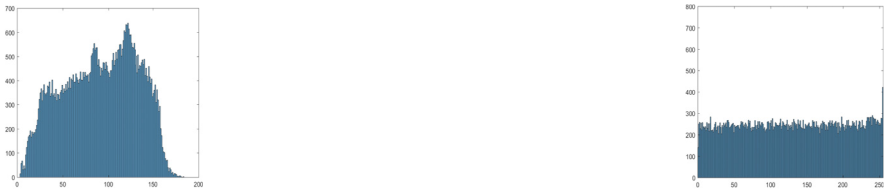


Figure 11. Histogram of grayscale image (left) and histogram of encrypted image (right)

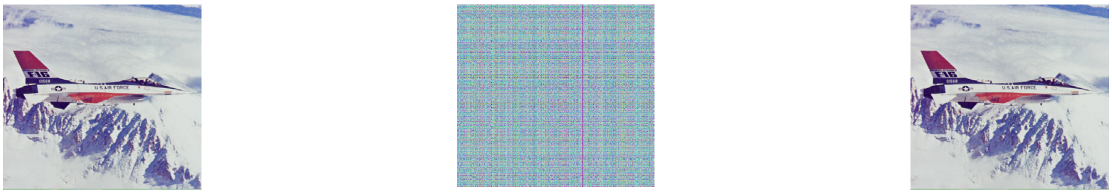
Table 9. Test results of extended algorithm for binary image

Index	Encrypted image
Chi-squared statistic	178.0225
Pearson correlation coefficient (Horizontal)	0.0102
Pearson correlation coefficient (Vertical)	0.0080
Pearson correlation coefficient (Diagonal)	0.0078
Information entropy	7.9948
Execution efficiency	1.7938Mbps
mean of NPCR	99.4210%
mean of UACI	33.6715%

5.2 Extended Algorithm for RGB Image Encryption

The color image contains three pixel matrices of R, G, and B, and each pixel matrix can be regarded as a grayscale image. In the extended color image encryption algorithm, the three pixel matrices of the color image are respectively encrypted using proposed grayscale image encryption algorithm, and then the three encrypted images are combined into one color encrypted image. At the time of decryption, the acquisition of the original color image is completed by separately decrypting the three pixel matrices of the color encrypted image.

The original color image Airplane, the encrypted color image, and the decrypted image are listed in Figure 12. The histograms of the three components of R, G, and B of the original color image and the histograms of the three components of R, G, and B of the encrypted image are shown in Figure 13.

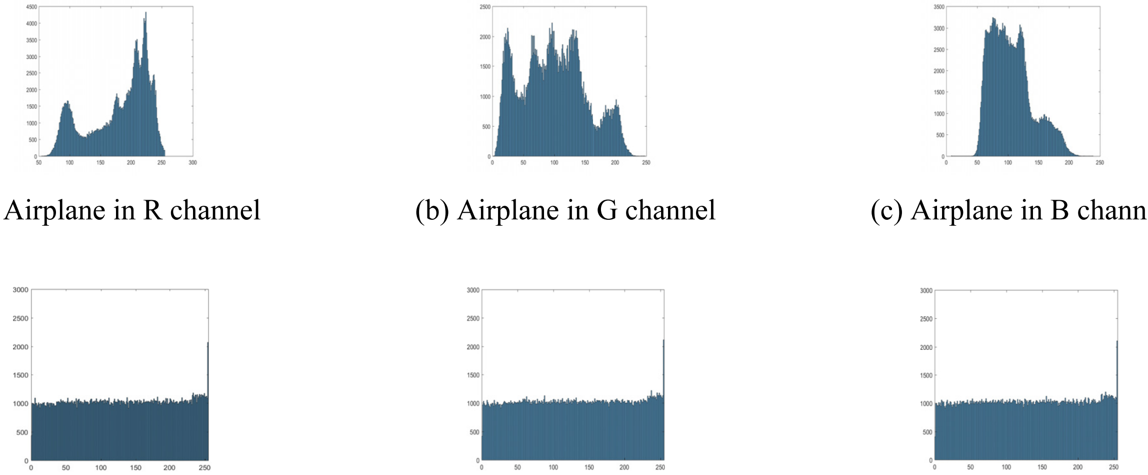


(a) Original image

(b) Encrypted image

(c) Decrypted image

Figure 12. Color image encryption/Decryption results



(a) Airplane in R channel

(b) Airplane in G channel

(c) Airplane in B channel

(d) Encrypted image in R channel

(e) Encrypted image in G channel

(f) Encrypted image in B channel

Figure 13. Histograms of original color image and histograms of encrypted image

The chi-square statistic, Pearson correlation coefficient, Information entropy, mean of NPCR and UACI (executing algorithm 50 times) of the encrypted image and execution efficiency is shown in Table 10. According to Table 10, it can be seen that the algorithm achieves the same effect as grayscale image encryption in encryption of color image.

Table 10. Test results of extended algorithm for color image

Index	Encrypted image		
	R	G	B
Chi-squared statistic	195.1280	175.3419	197.1188
Pearson correlation coefficient (Horizontal)	0.0102	0.0126	0.0143
Pearson correlation coefficient (Vertical)	0.0083	0.0079	0.0085
Pearson correlation coefficient (Diagonal)	0.0081	0.0072	0.0081
Information entropy	7.9964	7.9968	7.9963
Execution efficiency	2.0338Mbps	2.0957Mbps	2.1438Mbps
mean of NPCR	99.4581%	99.3570%	99.3285%
mean of UACI	33.3875%	33.2992%	33.1036%

6 Conclusion

In this paper, driven by 1D-Tent map and 1D-Logistic map, a 2D chaotic map with plaintext correlation property, plaintext-related hybrid modulation map (PHMM) is proposed. Its performance is verified by bifurcation diagram, Lyapunov exponent (LE) and Spearman correlation test. Then, an efficient pixel positions scrambling method, non-repetitive chaotic displacement (NRCD) is proposed based on PHMM. Based on the combined use of NRCD and bit-plane reconstruction, a novel image encryption algorithm is proposed. In this algorithm, the permutation and diffusion processes are completed in the same stage. A series of security analysis experiments demonstrate the resistance of the proposed algorithm to attacks such as brute force attack, known plaintext attack, selective plaintext attack, noise and data loss attack. The algorithm also has excellent performance in efficiency analysis experiment. Further, extended algorithms for Binary image encryption and RGB image encryption also performed well in a series of tests. Therefore, the proposed algorithm has certain application prospects.

Acknowledgements

This work was partly supported by National Natural Science Foundation of China under grant Nos. 6U19A2086, 1802033, 61662016, and Innovation Project of GUET Graduate, No. 2017YJCX49.

References

[1] J. Fridrich, Symmetric Ciphers Based on Two-Dimensional Chaotic Maps, *International Journal of Bifurcation and Chaos*, Vol. 8, No. 6, pp. 1259-1284, November, 1998.

[2] X.-Y. Wang, L. Yang, R. Liu, A. Kadir, A Chaotic Image Encryption Algorithm Based on Perceptron Model, *Nonlinear Dynamics*, Vol. 62, No. 3, pp. 615-621, November, 2010.

[3] A. Kanso, M. Ghebleh, A Novel Image Encryption Algorithm Based on a 3D Chaotic Map, *Communications in Nonlinear Science and Numerical Simulation*, Vol. 17, No. 7, pp. 2943-2959, July, 2012.

[4] G.-R. Chen, Y. Mao, C.-K. Chui, A Symmetric Image Encryption Scheme Based on 3D Chaotic Cat Maps, *Chaos, Solitons and Fractals*, Vol. 21, No. 3, pp. 749-761, July, 2004.

[5] L. Liu, Q. Zhang, X. Wei, C. Zhou, Image Encryption Algorithm Based on Chaotic Modulation of Arnold Dual Scrambling and DNA Computing, *Journal of Computational & Theoretical Nanoscience*, Vol. 4, No. 11, pp. 3537-3542, November, 2011.

[6] O. Lafe, Data Compression and Encryption Using Cellular Automata Transforms, *Engineering Applications of Artificial Intelligence*, Vol. 10, No. 6, pp. 581-591, December, 1997.

[7] R.-J. Chen, J.-L. Lai, Image Security System Using Recursive Cellular Automata Substitution, *Pattern Recognition*, Vol. 40, No. 5, pp. 1621-1631, May, 2007.

[8] H. B. Kekre, T. Sarode, P. N. Halarnkar, Partial Image Scrambling Using Walsh Sequency in Sinusoidal Wavelet Transform Domain, *Intelligent Systems Technologies and Applications*, Springer International Publishing, 2016.

[9] Y. Zhou, S. Agaian, V. M. Joyner, K. Panetta, Two Fibonacci P-code Based Image Scrambling Algorithms, *Image Processing: Algorithms and Systems VI*, San Jose, CA, USA, 2008, pp. 1-12.

[10] T. Podoba, J. Giesl, K. Vlcek, Image Encryption in Wavelet Domain Based on Chaotic Maps, *2nd International Congress on Image & Signal Processing*, Tianjin, China, 2009, pp.1-5.

[11] R. Lan, J. He, S. Wang, Y. Liu, X. Luo, A Parameter-selection-based Chaotic System, *IEEE Transactions on Circuits and Systems II: Express Briefs*, Vol. 66, No. 3, pp. 492-496, March, 2019.

[12] R. Lan, J. He, S. Wang, T. Gu, X. Luo, Integrated Chaotic Systems for Image Encryption, *Signal Processing*, Vol. 147, pp. 133-145, June, 2018.

[13] Z. Hua, F. Jin, B. Xu, H. Huang, 2D Logistic-sine-coupling Map for Image Encryption, *Signal Processing*, Vol. 149, pp. 148-161, August, 2018.

[14] C.-Y. Song, Y.-L. Qiao, A Novel Image Encryption Algorithm Based on DNA Encoding and Spatiotemporal Chaos, *Entropy*, Vol. 17, No. 10, pp. 6954-6968, October, 2015.

[15] N. Nandy, D. Banerjee, C. Pradhan, Color Image Encryption Using DNA Based Cryptography, *International Journal of Information Technology*, pp. 1-8, February, 2018.

[16] Y. Zhou, L. Bao, C.-L. Chen, A New 1D Chaotic System for Image Encryption, *Signal Processing*, Vol. 97, pp. 172-182, April, 2014.

[17] Y.-Q. Zhang, X.-Y. Wang, A Symmetric Image Encryption Algorithm Based on Mixed Linear-nonlinear Coupled Map Lattice, *Information Sciences*, Vol. 273, pp. 329-351, July, 2014.

[18] X. Wu, H. Hu, B. Zhang, Parameter Estimation only from the Symbolic Sequences Generated by Chaos System, *Chaos, Solitons & Fractals*, Vol. 22, No. 2, pp. 359-366, October, 2004.

[19] W. Liu, K. Sun, C. Zhu, A Fast Image Encryption Algorithm Based on Chaotic Map, *Optics & Lasers in Engineering*, Vol. 84, pp. 26-36, September, 2016.

[20] S. J. Sheela, K. V. Suresh, D. Tandur, Image Encryption Based on Modified Henon Map Using Hybrid Chaotic Shift Transform, *Multimedia Tools and Applications*, Vol. 77, No. 19, pp. 25223-25251, October, 2018.

[21] Z. Hua, Y. Zhou, Image Encryption Using 2D Logistic-Adjusted-sine Map, *Information Sciences*, Vol. 339, pp. 237-253, April, 2016.

[22] Y. Zhou, L. Bao, C. L. P. Chen, Image Encryption Using a New Parametric Switching Chaotic System, *Signal Processing*, Vol. 93, No. 11, pp. 3039-3052, November, 2013.

[23] G. Ye, Image scrambling Encryption Algorithm of Pixel Bit Based on Chaos Map, *Pattern Recognition Letters*, Vol. 31, No. 5, pp. 347-354, April, 2010.

[24] Y. Zhang, The Image Encryption Algorithm with Plaintext-Related Shuffling, *IETE Technical Review*, Vol. 33, No. 3, pp. 1-13, October, 2015.

[25] G. Alvarez, S.-J. Li, Some Basic Cryptographic Requirements for Chaos-Based Cryptosystems, *International Journal of Bifurcation and Chaos*, Vol. 16, No. 8, pp. 2129-2151, August, 2006.

[26] Q. Zhang, L. Guo, X. Wei, Image Encryption Using DNA Addition Combining with Chaotic Maps, *Mathematical and Computer Modelling*, Vol. 52, No. 11-12, pp. 2028-2035, December, 2010.

[27] H. Cheng, Y. Song, C. Huang, Q. Ding, Self-Adaptive Chaotic Logistic Map: An Efficient Image Encryption Method, *Journal of Internet Technology*, Vol. 17, No. 4, pp. 743-752, July, 2016.

[28] T. Wu, X. Fan, K. Wang, J. Pan, C. Chen, Security Analysis and Improvement on an Image Encryption Algorithm Using Chebyshev Generator, *Journal of Internet Technology*, Vol. 20, No. 1, pp. 13-23, January, 2019.

[29] X. Li, Z. Xia, A Distribution Outsourcing Scheme Based on Partial Image Encryption, *Journal of Internet Technology*, Vol. 19, No. 3, pp. 807-814, May, 2018.

[30] T. Wei, P. Lin, Y. Wang, L. Wang, Stability of Stochastic Impulsive Reaction-diffusion Neural Networks with S-type Distributed Delays and Its Application to Image Encryption, *Neural Networks*, Vol. 116, pp. 35-45, August, 2019.

[31] X. Wang, Z. Li, A Color Image Encryption Algorithm Based on Hopfield Chaotic Neural Network, *Optics and Lasers in Engineering*, Vol. 115, pp. 107-118, April, 2019.

[32] P. Mani, R. Rajan, L. Shanmugam, Y. Joo, Adaptive Control for Fractional Order Induced Chaotic Fuzzy Cellular Neural Networks and Its Application to Image Encryption, *Information Sciences*, Vol. 491, pp. 74-89, July, 2019.

[33] J. Deng, S. Zhao, Y. Wang, L. Wang, H. Wang, H. Sha, Image Compression-encryption Scheme Combining 2D Compressive Sensing with Discrete Fractional Random Transform, *Multimedia Tools and Applications*, Vol. 76, No. 7, pp. 10097-10117, April, 2017.

[34] R. Huang, K. H. Rhee, S. Uchida, A Parallel Image Encryption Method Based on Compressive Sensing, *Multimedia Tools and Applications*, Vol. 72, No. 1, pp. 71-93, September, 2014.

[35] M. Hasler, Y. L. Maistrenko, An Introduction to the Synchronization of Chaotic Systems: Coupled Skew Tent Maps, *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, Vol. 44, No. 10, pp. 856-866, October, 1997.

[36] K. B. Athreya, J. Dai, Random Logistic Maps. I, *Journal of Theoretical Probability*, Vol. 13, No. 2, pp. 595-608, April, 2000.

[37] H.-H. Zhao, Y.-N. Wang, X.-J. Peng, Z.-J. Qiao, Gradient-based Compressive Sensing for Noise Image and Video Reconstruction, *IET Communications*, Vol. 9, No. 7, pp. 940-946, May, 2015.

Biographies

Mingzhe Liu received his B.Sc in Computer Application from Chengdu University of Technology, China, in 1994; Ph.D. in Computer Science from Massey University, New Zealand, in 2010. He is a Professor of School of Network Security, Chengdu University of Technology, China. His research interests include intelligent information processing, information security.

Feixiang Zhao received his B.Sc in Measurement, Control Technology and Instrumentation from Chengdu University of Technology in 2018. He is studying for his master’s degree in Chengdu University of Technology. His research interests include digital image processing and machine learning.

Xin Jiang received his B.Sc in Information and Computing Science from Chengdu University of Technology, China, in 2012. He is currently studying for a doctoral degree in Nuclear technology and Application, Chengdu University of Technology, China. His research interests include medical imaging, deep learning, and cyberspace security, etc.

Xianghe Liu received the B.S. degree in nuclear engineering and technology from the Engineering and Technical College of Chengdu University of Technology, Sichuan, China, in 2018. He is currently a M.S. candidate at Chengdu University of Technology, Sichuan, China. His research interest covers processing of nuclear data and simulation of nuclear medical imaging.

Yining Liu is a professor in Guilin University of Electronic Technology, Guilin, China. He received the B.Sc in Applied Mathematics from Information Engineering University, Zhengzhou, China, and the Ph.D. degree in Mathematics from Hubei University, Wuhan, China, in 2007. His research interests include information security and big data.