Mustansiriyah Journal of Pure and Applied Sciences

www.MJPAS.com

An image encryption system based on random matrix interpolation followed by iterative generation of sequences

Zheng Bojun[1], Ma Tongtong[2], Liu Boxuan[3]

[1,2] Department of Computer Science, School of Information Engineering, Dalian University: 118863339453@163.com[1] , 2836814966@qq.com[2]

[3] Shandong Ganglianhua Pipeline Oil Transportation Co., Ltd : 3mu2159837056@163.com[3]

A B S T R A C T

To solve the problem of image security during data transmission, researchers in the field of information security have proposed several different image encryption techniques. In recent years, chaotic systems have been widely applied to image encryption because of their extreme initial value sensitivity and pseudo-random properties. To further improve the security of the image encryption algorithm, this paper proposes a new image encryption scheme. The scheme consists of three stages, firstly, a novel two-dimensional nonlinear chaotic system is constructed using a random matrix and a bilinear interpolation function, and then the chaotic sequence generated by the iteration of the system is used to perform an XOR operation with the image, and finally, a disambiguation operation is carried out to obtain the ciphertext image. By analyzing the histogram, information entropy, adjacent pixel correlation, key sensitivity, keyspace, and differential attack. The experimental test results show that the algorithm has high security, fast encryption speed, and strong anti-attack ability.

Keywords*: image encryption; random matrix; bilinear interpolation; chaos; iteration*

1. Introduction

Rapid developments have been observed in network and multimedia technologies with the rise of the information technology era. Digital images have become a significant medium in the transmission of information, and numerous images contain sensitive information such as national security and military secrets. Therefore, how to ensure the security of information in the process of information transmission is an important topic in modern information science research[1].

Modern cryptography includes cryptography and cryptanalysis. Cryptography transforms information based on the principles of integrity, confidentiality, and consistency

*Corresponding author : Zheng Bojun[1]
E-mail address: 118863339453@163.com

to ensure that information is not stolen and used by unauthorized persons during transmission[3]. Cryptanalysis is used to analyze and decipher passwords. The unencrypted message is called plaintext, the encrypted message is called ciphertext, the sequence used in the encryption process is called a cipher, and the key information used to generate the cipher is called a key. Modern cryptography field is divided into two main types, one is symmetric key cryptography, such as DES[4], AES[5], and the other is asymmetric key cryptography, such as RSA[6], ECC[7]. Chaotic cryptography belongs to the category of symmetric key cryptography, which has become an important modern cryptography due to its good pseudo-randomness as well as initial sensitivity.

As an important branch of nonlinear science, chaos theory was first proposed by the meteorologist Lorenz and gave the Lorenz chaotic system[8]. Chaotic systems are widely used in image encryption technology due to their high initial value sensitivity, unpredictability, pseudo-randomness, and other characteristics. Mathematician Matthews published the first encryption algorithm based on chaos theory, which was the first proposal of chaotic cryptography[9]. Since then more and more experts and scholars have proposed image encryption algorithms based on chaos theory. Researchers have proposed an assortment of encryption methods based on chaotic systems of different dimensions[10]. Talhaoui et al.[10] proposed an encryption algorithm based on one-dimensional cosine polynomial (1-DCP) chaotic mapping. Luo et al. [12] proposed an image encryption algorithm based on a dual chaotic system. Using two-dimensional Baker chaos mapping to control the system parameters and state variables of the logistic chaos mapping, the algorithm has better security performance. peng et al. [17] proposed a new four-dimensional chaotic system capable of generating multi-wing chaotic attractors, and this four-dimensional multi-wing hyperchaotic system can be used for hybrid image encryption with physical chaotic encryption and high-level encryption standard encryption cascade. In recent years, researchers have combined chaotic systems with Arnold transformations, quantum communication, compression awareness, DNA coding operations alternative boxes, block structures, deep learning networks, and complex mathematical models to build many high-performance image encryption algorithms[18].

Many scholars have combined chaotic systems with transformations of matrices to design additional encryption schemes[24]. Zhang et al.[24] achieved good security performance by directly disambiguating the image using a two-dimensional rectangular transform with correlated substitutions for each pixel based on the image pixels. Liang et al. [25] proposed a hybrid encryption scheme for images based on Lagrange interpolation, generalized Henon mapping, and nonlinear operations on matrices. Due to the combination of multiple nonlinear methods and random factors, the scheme can withstand many types of attacks. Based on the above research, a new encryption method is proposed in this paper. Firstly, two random matrices are bilinearly interpolated to construct a two-dimensional nonlinear chaotic system, and then the chaotic sequence generated by this chaotic system is utilized for image encryption. We analyze the chaotic properties of this system in Section 2, give the encryption method analyze the encryption effect in Section 3, and finally conclude in Section 4.

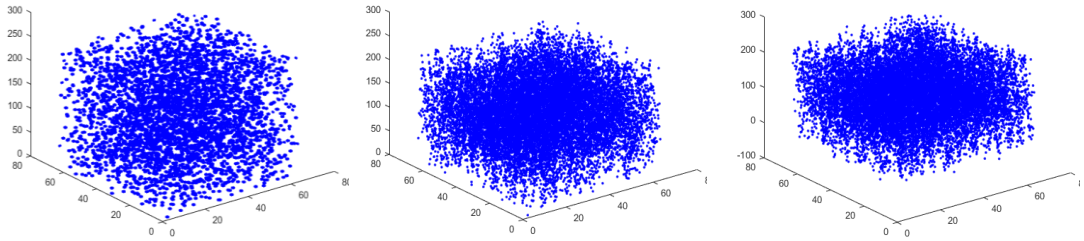
2. Chaotic characteristic analysis

2.1 Advantages of bilinear interpolation

Bilinear interpolation is an extension of linear interpolation, which is essentially linear interpolation in two directions, but nonlinear overall, for two functions of independent variables $z = f(x, y)$ on a two-dimensional linear grid. For example, suppose we know the values of the function $z = f(x, y)$ at four points $A_{11} = (x_1, y_1)$, $A_{12} = (x_1, y_2)$, $A_{21} = (x_2, y_1)$, $A_{22} = (x_2, y_2)$, then the bilinear interpolation is calculated as shown in equation (1).

$$f(x,y) \approx \frac{f(A_{11})}{(x_2-x_1)(y_2-y_1)}(x_2-x)(y_2-y)+\frac{f(A_{21})}{(x_2-x_1)(y_2-y_1)}(x-x_1)(y_2-y) + \frac{f(A_{12})}{(x_2-x_1)(y_2-y_1)}(x_2-x)(y-y_1)+\frac{f(A_{22})}{(x_2-x_1)(y_2-y_1)}(x-x_1)(y-y_1) \tag{1}$$

Equation (1) is an illustration of the calculation process of the bilinear interpolation method (in the experiment, we can directly call the bilinear interpolation function for calculation). The bilinear interpolation algorithm has the advantages of high accuracy, simple calculation, and high efficiency, and the calculation takes into account the influence of the values of the four neighboring points around the target point. We performed nearest neighbor interpolation, bilinear interpolation, and cubic convolution interpolation for a 64×64 random matrix, respectively, and plotted the images of the interpolation results, as in Figure 1(a), Figure 1 (b), Figure 1 (c). We found that bilinear interpolation overcomes the numerical discontinuity compared to the nearest neighbor interpolation method, and compared to cubic convolutional interpolation, there will be no negative numbers thus affecting the subsequent iterative operations, so bilinear interpolation is chosen for this experiment.



(a) **(b)** **(c)**

Figure 1. Point set diagram of different interpolation methods

2.2 Construction of chaotic systems

First generate two 256×256 random number matrices F and G, with matrix values in the range of 0 to 255. Using the bilinear interpolation function, a two-dimensional nonlinear chaotic system is constructed by doing bilinear interpolation on the matrices F, and G, respectively, and the construction method is shown in the system of equations (2):

$$\begin{cases} z_1 = F(x,y) \\ z_2 = G(x,y) \end{cases} \tag{2}$$

$F(x, y)$ *and* $G(x, y)$are functions of the bilinear interpolation of F and G, respectively.

2.3 Phase diagram of the system

The phase diagram of the system can reflect the distribution of the sequences produced by the system in space. equations (2) Given the initial values x=25,y=125 for iteration, Figure 2(a) represents the phase diagram of x,y versus $z1$ and Figure 2(b) represents the phase diagram of x,y versus $z2$. The phase diagram shows that the sequences generated after the iteration of this system have a strong randomness.

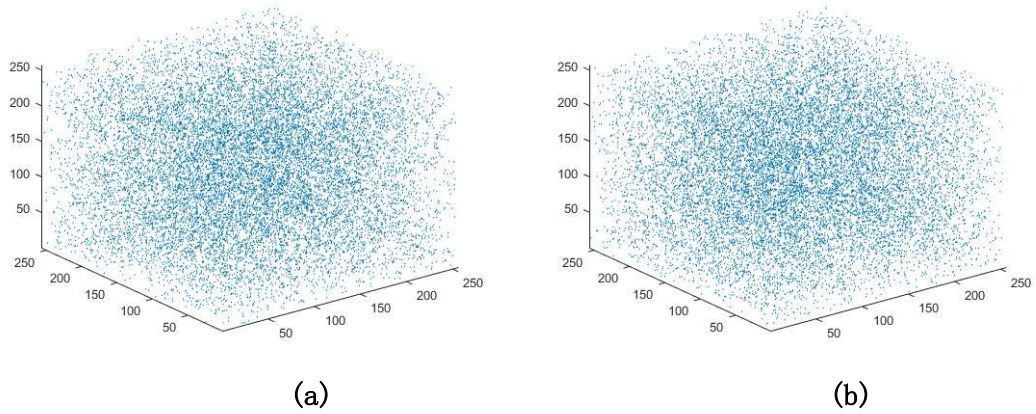


Figure 2. Phase diagram

2.4 NIST testing

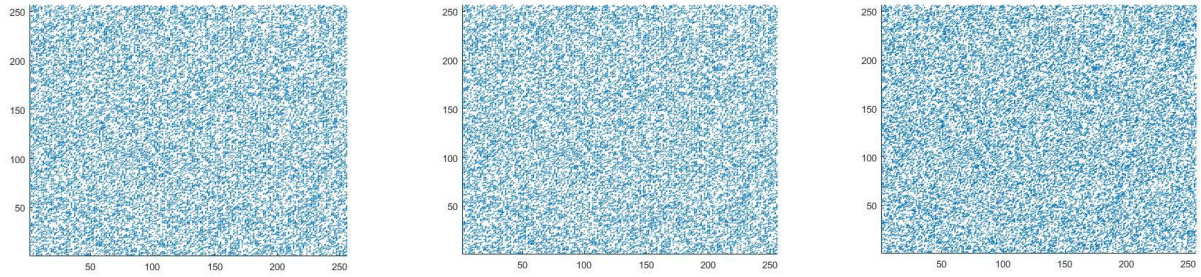
To test the randomness of the chaotic sequences generated by the chaotic system, NIST SP800 is used to carry out the test, the experiment contains a total of 15 test data components, corresponding to the experimental results of the values shown in Table 1, through the results of the experiment, it is concluded that the sequences generated by the system have a strong randomness, and it applies to the image encryption.

Table 1 NIST test results

Test items	P-value	Test results
Frequency	0.6277	Pass
Block Frequency	0.1503	Pass
Cumulative Sums	0.9972	Pass
Runs	0.7985	Pass
Longest Run of Ones	0.3021	Pass
Rank	0.1514	Pass
Discrete Fourier Transform	0.1358	Pass
Non-overlappingTemplate Matching	0.1256	Pass
OverlappingTemplate Matching	0.9625	Pass
Universal Statistical	0.1814	Pass
Approximate Entropy	0.8485	Pass
Random Excursions	0.1429	Pass
Random Excursions Variant	0.2865	Pass
Serial	0.3566	Pass
Linear Complexity Test	0.3589	Pass

2.5 System bifurcation diagram

Bifurcation is one of the basic features of chaotic mapping, and a bifurcation map is used to observe how the state of the system changes subsequently when one of the variables in the system changes. We take three elements *f12*, *f22*, and *f32* of the matrix F as parameters, Figure 3(a)-(c) represent bifurcation plots of z1 varying with *f12*, *f22*, and *f32*, respectively. From the figure, it can be seen that the encryption scheme in this paper has a large chaos interval, and the system has good chaos characteristics no matter which parameter is chosen.



(a) (b) (c)

Figure 3. Bifurcation diagram

2.6 Initial sensitivity analysis

For example, we set the initial value to x=2.1,y=10.5. 200 iterations are performed according to equation (2) (assign the results of z_1 and z_2 each time to x and y, respectively, and continue the calculation according to equation (2)), and store the results after each iteration in array A. Change the initial value x=2.1000001, repeat the above operation and store the result in array B to get two sequences. The values of the two sequences are subtracted in turn and then plotted as a line graph, through which it can be seen that this system is very sensitive to the initial state. As shown in Figure 4. It is obvious from the line graph that when a small change in the initial value of this system occurs, the sequence generated by the corresponding iteration changes significantly, which indicates that this system is very sensitive to the initial state.

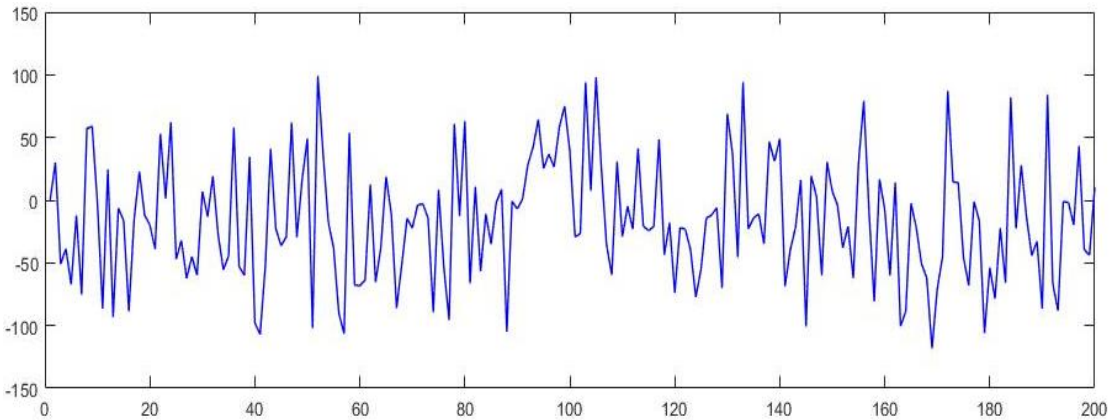


Figure 4. Sequationuence difference line chart

2.7 Experimental analysis of its periodicity

First, we generate a random set of initial data *x,y* and record the results *Xi*, *Yi* after the first interpolation. It was subjected to (512×512) iterations (in the interpolation case) according to the method of equation (2) above. Since there are fractions in the results of the computer's interpolation calculations, we cannot find the exact period, so we can only find the period within a certain error range, which we call "period-like". Δx_1 and Δx_2 are the "period-like" error ranges, which are obtained using equation 3.z_1 and z_2 show the results after each iteration of the matrix (in the case of interpolation), respectively. When $\Delta x_1$and Δx_2 are less than or equational to 1, 0.5, 0.3, 0.1, 0.05, and 0.01, find the number of "class cycles" of the experimental data in the range of 512×512 times. To increase the accuracy of the experiment, we used 150 sets of random initial data for validation. Calculate the number of "cycle-like" occurrences for each set of experimental data in the range of 512×512 iterations. The maximum, minimum, mean, standard deviation, and variance of the number of "class cycles" in the 150 sets of data were recorded. As shown in Table 2.

$$\begin{cases} \Delta x_1 = | z_1 - Xi | \\ \Delta x_2 = | z_2 - Yi | \end{cases} \tag{3}$$

Table 2 shows that when Δx_1 and Δx_2 are in the range $[0.05,1]$, there are 512 ×512 iterations and some of the 150 sets of experimental data will be " class cycles ".However, as the values of Δx_1 and Δx_2 decrease, the number of experimental data with "class cycles" decreases. Accordingly, the number of "class cycles" of the experimental data with "class cycles" decreases. The number of "class cycles" for the "class cycle" experimental data also decreases. When Δx_1 and Δx_2 are less than or equational to 0.01, the 150 sets of experimental data, with 512×512 iterations, respectively, do not have overlapping points, i.e., in this sense, the iteration period of all 150 sets of initial values is greater than 512×512.

Table 2. Table of class cycle statistics

(Δx_1&Δx_2)≤	NOI	NDL	MAX	MIN	AVG	SD	ANON
1	512*512	150	79	0	32.24	16.84	283.68
0.5	512*512	150	21	0	8.441	4.79	22.98
0.3	512*512	150	10	0	2.826	2.33	5.44
0.1	512*512	150	3	0	0.353	0.58	0.33
0.05	512*512	150	1	0	0.086	0.28	0.07
0.01	512*512	150	0	0	0	0	0

NOI: Number of iterations NDL: Number of experimental data MAX: Maximum number of class cycles
MIN: Minimum number of class cycles AVG: Average number of class cycles
SD: Standard deviation of the number of class cycles ANOV: Class cycle count variance

3. Image encryption experiment

3.1 Encryption and decryption algorithm process

As can be seen by Fig 1, the points generated iteratively after matrix interpolation are predominantly between the center ranges, and the closer to the middle position the more points are generated, indicating that the interpolation of the matrix is mostly done at the center of the matrix during iteration, which also indicates that the closer the elements of matrix F and matrix G are to the center when interpolation iteration is performed, the more sensitive they are. Based on the above analysis, we select nine elements from matrix F to form a 3×3 matrix as the key parameter matrix denoted N. These nine elements are *F*(127,127), *F*(127,128), *F*(127,129), *F*(128,127), *F*(128,128), *F*(128,128), *F*(128,129), *F*(129,127), *F*(127,128), *F*(129,129). Similarly, we select nine elements from the matrix G to form a 3×3 matrix as the key parameter matrix denoted as K, these nine elements are *G*(127,127), *G*(127,128), *G*(127,129), *G*(128,127), *G*(128,128), *G*(128,128), *G*(128,129), *G*(129,127), *G*(127,128), *G*(129,129).

Step 1: Read the plaintext image (Lena as an example) and convert it into a 512×512 two-dimensional pixel matrix I.

Step 2: Given the initial values *x,y* and the key parameter matrices N and K (the parameter matrices can be used as initial matrices or you can adjust the parameters yourself).512×512 iterations are performed according to equation (2), and the values of z_1 and z_2 after each iteration are stored in the arrays Z_1 and Z_2, respectively, after which the sequences are processed according to equation (4) and the resulting sequences are stored in the 512×512 matrices D1 and D2 in sequences by row.

$$\begin{cases} D1(m,n) = \mathrm{mod}(Z_1(i) \times abs(sum(N) - sum(K)),256) \\ D2(m,n) = \mathrm{ceil}(\mathrm{mod}(Z_2(i) \times abs(sum(N) - sum(K)),256)) \end{cases} \tag{4}$$

The key parameter matrices N and K are summed separately, followed by the difference. The matrix D2 is rounded upwards to keep the range of element values in the matrix between 1 and 256, which is convenient for the subsequent scrambling operation. Since the pixel value range of the plaintext image is 0-255 and the pixel value is an integer type, the element value in matrix D1 will be double type by default, so we first convert the elements in matrix D1 to uint8 type, Then the bit-by-bit xor operation is performed on D1 and I to obtain the cipher text image E1.

Step 3: Scrambling the elements in E1. First, scramble the elements of each row in E1, Generate a 512×512 all-0 matrix L1, traversing the elements of each row of E1 from left to right, For example: when traversing to the element *E*1(*i,j*), find the value $m = D2(i,j)$ of the element in the same position in D2, exchange the position of the element *E*1(*i,j*) with *E*1(*i,m*), As shown in equation (5). At the end of the scrambling, copy E1 to generate the intermediate quantity E2. Scrambles the elements of each column of E2, Generate a 512×512 all-0 matrix L2, traversing the elements of each column of E2 from top to bottom, For example: when

traversing to the element $E2(i,j)$, find the value $n = D2(i,j)$ of the element in the same position in D2, exchange the position of the element $E2(i,j)$ with $E2(n,j)$, As shown in equation (6), The final encrypted image E2 is obtained.

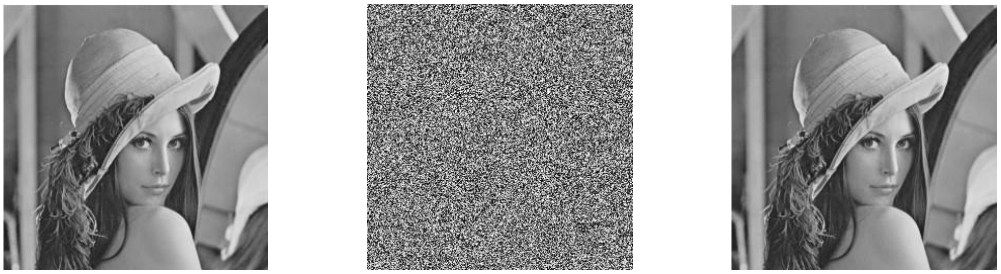
$$\begin{cases} m = D2(i,\ j) \\ L1(i,\ j) = E1(i,\ j) \\ E1(i,\ j) = E1(i, m) \\ E1(i, m) = L1(i,\ j) \end{cases} \tag{3}$$

$$\begin{cases} n = D2(i,\ j) \\ L2(i,\ j) = E2(i,\ j) \\ E2(i,\ j) = E2(n,\ j) \\ E2(n,\ j) = L2(i,\ j) \end{cases} \tag{4}$$

The decryption method is the reverse operation of the encryption method. First, Reverse scrambling the elements of each column in E2, the method of reverse scrambling per-column elements is similar to scrambling per-column elements, but in this case, the traversal order is from bottom to top. At the end of the scrambling, copy E2 to generate the intermediate quantity E3. Reverse scrambling the elements of each row in E3, the method of reverse scrambling per-row elements is similar to scrambling per-row elements, but in this case, the traversal order is right-to-left. Then a bit-by-bit xor operation is performed on E3 and D1 to obtain the plaintext image E.

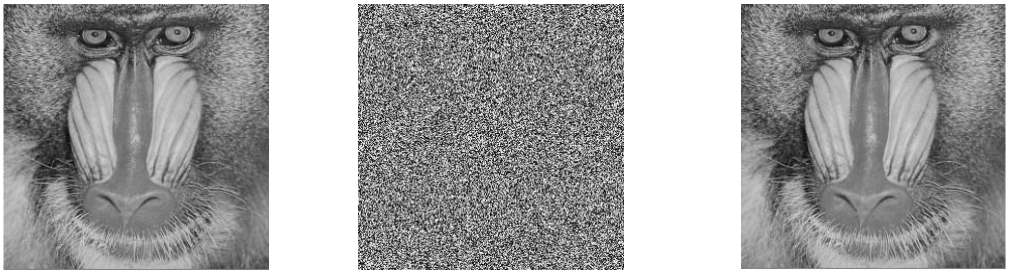
3.2 Encryption and decryption test results graph

In this experiment, we use three sets of images as experimental subjects. The grayscale image Lena, grayscale image Baboon, and grayscale image Peppers are shown in Figure 5 (a), Figure 6(a), and Figure 7 (a), respectively. Simulation experiments are performed using the Matlab platform. Figure 5(a), and Figure 6(a) are 512×512 plaintext images, and Figure 7(a) shows a 256×256 plaintext image. Figure 5(a)-Figure 5(c), Figure 6(a)-Figure 6(c), and Figure 7(a)-Figure 7(c) show the plaintext image, ciphertext image, and decrypted image, respectively. The encryption algorithm works for both 256×256 and 512×512 images.



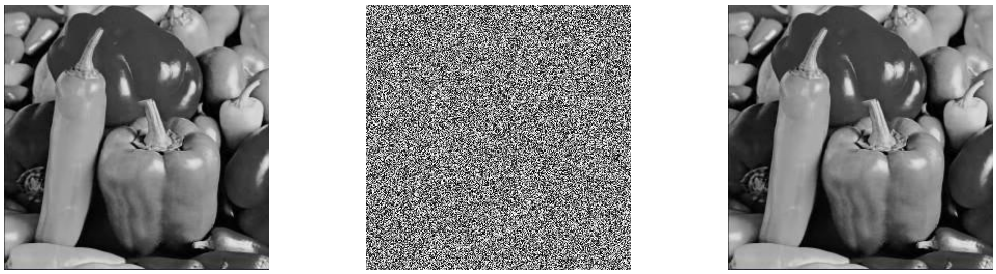
(a) (b) (c)

Figure 5. Lena encrypted and decrypted images



(a) (b) (c)

Figure 6. Baboon encrypted and decrypted images

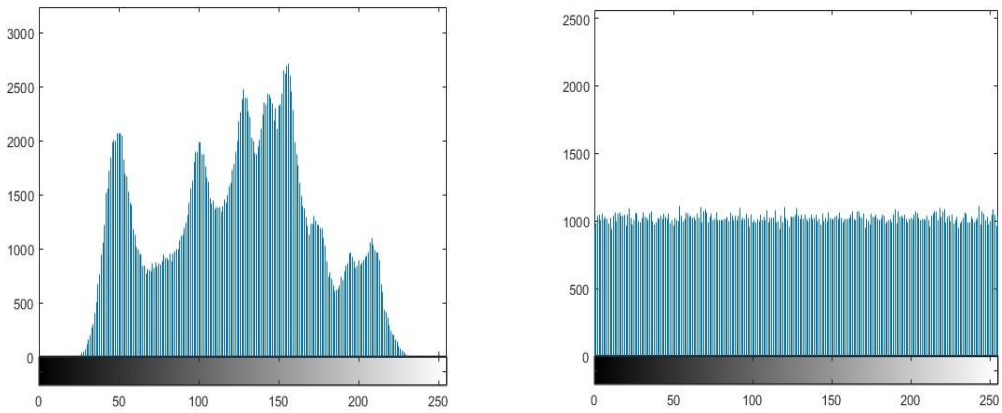


(a) (b) (c)

Figure 7. Peppers encrypted and decrypted images

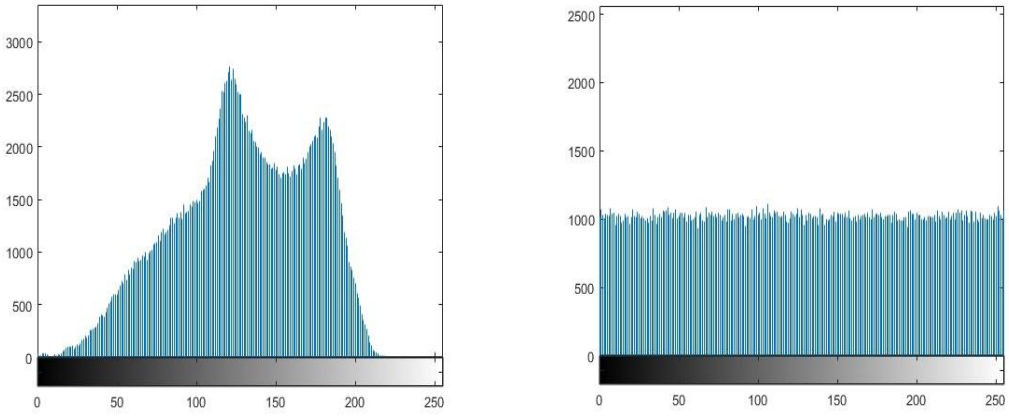
3.3 Histogram performance analysis

The histogram can accurately quantify the pixel distribution in an image. We can judge whether the encryption scheme is secure or not by observing the histogram of the cipher image. The more balanced the histogram is, the less statistical information the image shows, and the more secure the image encryption scheme is.The histograms of the plaintext images Figure 5(a), Figure 6(a), Figure 7(a) and ciphertext images Figure 5(b), Figure 6(b), Figure 7(b) in this experiment are shown in Figure 8(a), Figure 9(a), Figure 10(a) and Figure 8(b), Figure 9(b), Figure 10(b), respectively. We can see that the encrypted histogram is more balanced, with a more even distribution of pixel points, which is difficult to crack and highly secure.



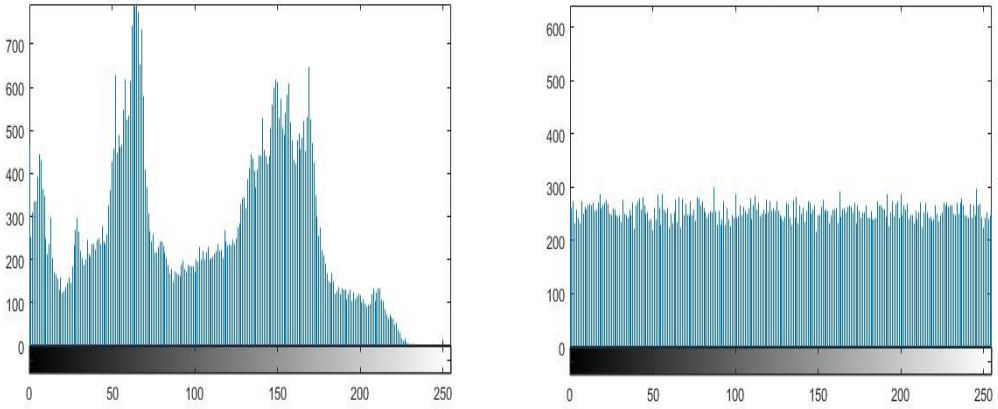
(a) (b)

Figure 8. Lena's plaintext ciphertext histogram



(a) (b)

Figure 9. Baboon plaintext ciphertext histogram



(a) (b)

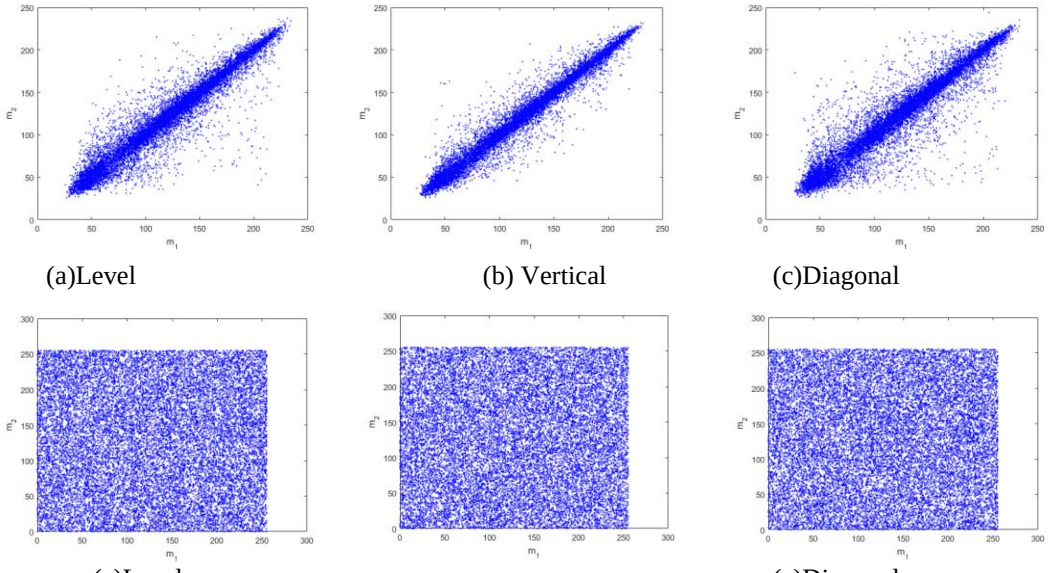
Figure 10. Peppers plaintext ciphertext histogram

3.4 Adjacent pixel correlation performance analysis

The image has a high correlation between adjacent pixels, a property that allows an attacker to roughly recover the plaintext image using very few pixel points, so this correlation should be broken during the encryption process. The maximum correlation coefficient is 1 and the minimum correlation coefficient is 0. The adjacent correlation coefficient of the encrypted image should be close to 0 to make the attacker unpredictable and unbreakable for the adjacent pixels and achieve the effect of secure encryption. The equation for the adjacent pixel correlation coefficients in horizontal, vertical, and diagonal directions is shown in equation (7).

$$\begin{cases} R_{to} = \dfrac{\operatorname{cov}(t,o)}{\sqrt{D(t)}} \\ E(t) = \dfrac{1}{n}\sum_{i=1}^{n} t_i \\ D(t) = \dfrac{1}{n}\sum_{i=1}^{n} (t_i - E(t))^2 \\ \operatorname{cov}(t,o) = \dfrac{1}{n}\sum_{i=1}^{n} (t_i - E(t))(o_i - E(o)) \end{cases} \tag{5}$$

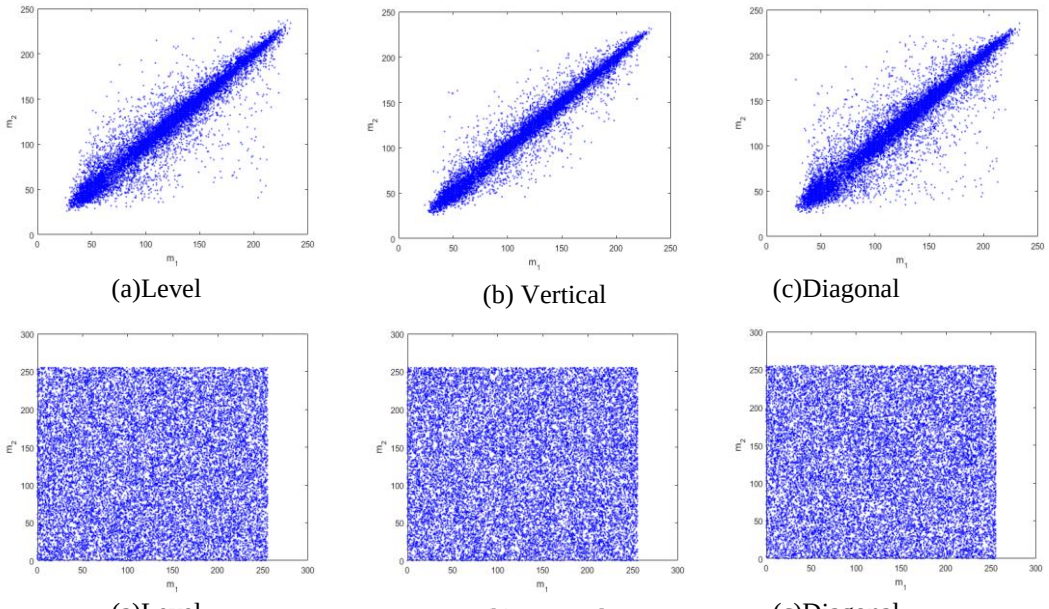
In equation (7), o is the adjacent pixels of t, n is the total number of pixel points in the 256×256 (512×512) image, R_{to} is the correlation of two adjacent pixels,$cov(t,o)$ is the covariance at the two-pixel points t and o. Figure 11(a)- Figure 11(c), Figure 12(a)- Figure 12(c), Figure 13(a)- Figure 13(c), represent the plots of adjacent pixel correlations in horizontal, vertical, and diagonal directions for three plaintext images. Figure 11(d)- Figure 11(f), Figure 12(d)- Figure 12(f), Figure 13(d)- Figure 13(f), represent the plots of adjacent pixel correlations in horizontal, vertical, and diagonal directions of the three ciphertext images. Table 3 shows the statistics of the correlation coefficient values between plaintext and ciphertext images and compares them with other literature data.



(a)Level (b) Vertical (c)Diagonal

(a)Level (b) Vertical (c)Diagonal

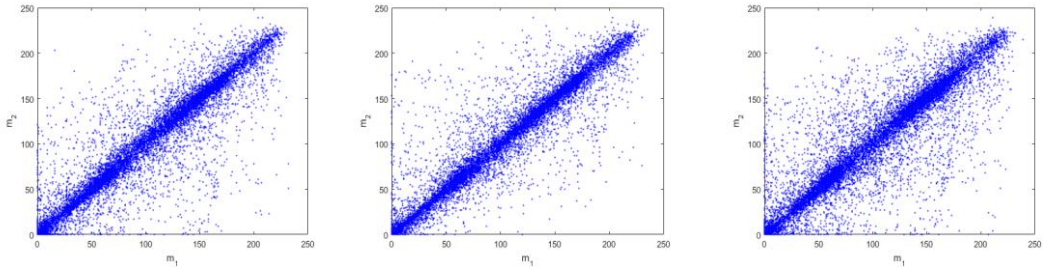
Figure 11.Correlation of lena plaintext ciphertext pixels



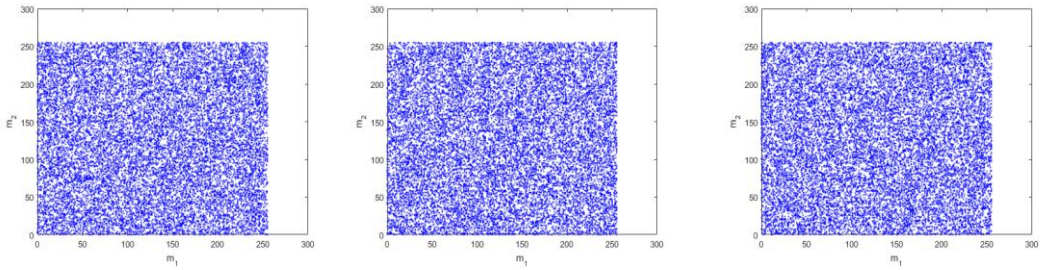
(a)Level (b) Vertical (c)Diagonal

(a)Level (b) Vertical (c)Diagonal

Figure 12.Correlation of baboon plaintext ciphertext pixels



(a)Level (b) Vertical (c)Diagonal



(a)Level (b) Vertical (c)Diagonal

Figure 13.Correlation of peppers plaintext ciphertext pixels

Table 3. Table of correlation values in the horizontal, vertical, and diagonal directions

Image	Level	Vertical	Diagonal
Lena(Original)	0.9671	0.9828	0.9556
Lena(Encrypten)	0.0031	0.0003	0.0030
Baboon(Original)	0.8654	0.7425	0.7124
Baboon(Encryption)	0.0046	0.0015	0.0013
Peppers(Original)	0.9470	0.9576	0.9180
Peppers(Encryption)	0.0012	0.0038	0.0047
Peppers[19]	0.0020	0.0080	0.0008
Baboon[19]	0.0009	0.0047	0.0050
Peppers[20]	0.0054	0.0060	0.0094
Lena[20]	-0.0021	-0.0032	0.0037
Lena[21]	0.0016	-0.0020	0.0047
Baboon[21]	0.0002	0.0017	0.0017
Lena[23]	0.0006	0.0007	0.0036

Combining the data of Figure 11, Figure 12, Figure 13, and Table 3, we can see that the encrypted image breaks the characteristic that the adjacent pixels of the original image have a high correlation compared with the original image, and the correlation coefficient of the adjacent pixels of the encrypted image is close to 0, indicating that the adjacent pixels are not

correlated. By comparing with experimental data from other literature, it was shown that this scheme has a lower adjacent pixel correlation[19].

3.5 Information entropy performance analysis

Information entropy is a quantitative measure of the degree of randomness of a signal source, which can also be described as the degree of information clutter. It is calculated as shown in equation (8):

$$H(s) = -\sum_{i=0}^{2^n-1} P(s_i)\log_2 P(s_i) \tag{8}$$

Where is the number of bits used to represent the symbols and $P(s_i)$ is the probability of occurrence of the signal s_i. The grayscale image used in this experiment is of order 256, and the value of information entropy can be calculated to be the maximum value of 8, which is also the most desirable value for the encryption effect. So, the closer the ciphertext image information entropy is to 8, the higher the security of this image encryption method. Through Table 4 we can see that the ciphertext image information entropy of the encryption algorithm in this paper is very close to the theoretical value 8, and by comparing the experimental results with other literature, we can see that the encryption method in this paper provides better information entropy values[12].

Table 4. Results of information entropy analysis

Image	Lena	Baboon	Peppers	Lena[12]	Lena[19]	Peppers [19]	Lena[29]
Entropy	7.9919	7.9918	7.9915	7.9894	7.8693	7.8734	7.8232

3.6 Differential analysis

A secure image encryption scheme should be resistant to differential attacks, which require a strong sensitivity to the plaintext image, where small changes in the plaintext image can greatly affect the ciphertext image, and such changes we can measure by the values of NPCR (Number of Pixels Changed Rate) and UACI (Unified Average Changing Intensity).NPCR represents the ratio of different gray values of different ciphertext images at the same position, and UACI denotes the average variation density between different ciphertext images. In general, the ideal values of NPCR and UACI are NPCR=99.6094% and UACI=33.4635%, respectively. The calculation formulas are shown in the following equation (9) and equation (10).

$$NPCR = \frac{\sum_{i,j} D1(i,j)}{M \times N} \times 100\% \quad (9)$$

$$UACL = \frac{1}{M \times N} \frac{\sum C1(i,j) - C2(i,j)}{255} \times 100\% \quad (10)$$

The definition of $D1(i,j)$ in equation (9) is shown in equation (11).

$$D1(i,j) = \begin{cases} 1, C1(i,j) \neq C2(i,j) \\ 0, C1(i,j) = C2(i,j) \end{cases} \quad (11)$$

M, N are the width and height of the image, $C1$ is the cipher image of the original image, $C2$ is the cipher image after changing one value of the original image pixels; $C1(i,j)$ and $C2(i,j)$ represent the grayscale values of the two cipher images at point (i,j), respectively. $D(i,j)=0$ if $C1(i,j)=C2(i,j)$, $D(i,j)=1$ if $C1(i,j) \neq C2(i,j)$.The metrics calculated for the encryption method in this paper are shown in Table 5, it can be seen that it is very close to the ideal value and compared with the values in other literature, which indicates that the encryption method in this paper has good resistance to differential attacks.

Table 5. NPCR and UACI indicators

Image	Lena	Baboon	Peppers	Lena[10]	Lena[12]	Baboon[18]	Lena[23]
NPCR	0.9961	0.9960	0.9959	0.9964	0.9966	0.9959	0.9959
UACL	0.3348	0.3346	0.3344	0.3343	0.3342	0.3020	0.3345

3.7 Speed analysis

The program was tested for speed and compared to other programs, which is the Matlab 2017 programming on a personal computer with 2.50GHz Intel(R)_Core(TM)_i5-7300 CPU and 16GB memory running on Microsoft Windows 10. We encrypted images of different sizes 150 times and calculated the average encryption speed and the average time is listed in Table 6. It can be seen that this algorithm encrypts faster and has real-time performance.

Table 6 Encryption speed test results

Image	256×256(Ours)	512×512(Ours)	256×256[21]	512×512[24]	256×256[25]	512×512[25]
Time(s)	1.1256	3.9558	7.1783	1.8514	1.9054	1.9054

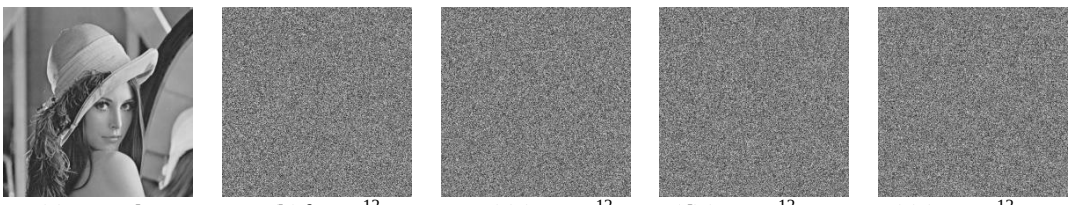
3.8 Keyspace and sensitivity analysis

The key space of the algorithm needs to be greater than 2^{128} to resist any kind of brute-force attack. The system parameters in this paper include two 3×3 parameter matrices and two iterative initial values, for a total of 20 parameters. If one parameter is calculated by 10^{12}, the key space can reach about 2^{240}. The key space of the encryption algorithm in this paper is large enough to effectively resist external attacks.

Key sensitivity refers to the fact that a small change in the initial key parameters produces a large difference in the encryption and decryption results. In this paper, we choose a value *f*, *g* from the key parameter matrices N and K, respectively, and we make small changes in the values of *f*, *g*, and the iterative initial values *x*, *y*. The changes are shown in equation (12).

$$\begin{cases} f = f + 10^{-12} \\ g = g + 10^{-12} \\ x = x + 10^{-12} \\ y = y + 10^{-12} \end{cases} \tag{12}$$

Decryption with the changed parameters yields the result shown in Figure 14(Lena as an example). Figure 14(a) is the correct decryption graph, and Figure 14(b)-Figure 14(e) represent the graphs of decryption results after the change of *f*, *g*, *x*, and *y*, respectively. The experimental results prove that the encryption method in this paper has high sensitivity.



(a) Raw data (b)$f+10^{-12}$ (c) $)g+10^{-12}$ (d) $)x+10^{-12}$ (e) $)y+10^{-12}$

Figure 14. Images decrypted with different parameters

4. Conclusion

In this paper, a two-dimensional nonlinear chaotic system is constructed by bilinear interpolation of a random matrix. The initial sensitivity and periodicity of the iterative sequences are analyzed experimentally to demonstrate that the system has certain chaotic properties. The two sequences generated by this system are used to encrypt images. By analyzing the histogram of the encrypted image, adjacent pixel correlation, information

entropy, differential, keyspace, and key sensitivity, it is proved that the encryption algorithm in this paper has a good encryption effect, high security, and anti-attack.

References

[1] Y. Zhang, L. Y. Zhang, J. Zhou, L. Liu, F. Chen and X. He, "A Review of Compressive Sensing in Information Security Field," in IEEE Access, vol. 4, pp. 2507-2519, 2016.https://doi.org/10.1109/ACCESS.2016.2569421

[2] Shukla, P.K.; Khare, A.; Rizvi, M.A.; Stalin, S.; Kumar, S. Applied Cryptography Using Chaos Function for Fast Digital Logic-Based Systems in Ubiquitous Computing. Entropy 2015, 17, 1387-1410. https://doi.org/10.3390/e17031387

[3] Fang, P., Liu, H., Wu, C. et al. A survey of image encryption algorithms based on chaotic system. Vis Comput (2022). https://doi.org/10.1007/s00371-022-02459-5

[4] Wu, Y.H., Dai, X.Q.: Encryption of accounting data using DES algorithm in a computing environment. J. Intell. Fuzzy Syst 39(4), 5085–5095 (2020). https://doi.org/10.3233/JIFS-179994

[5] Yang, C.H., Chien, Y.S.: FPGA implementation and design of a hybrid chaos-AES color image encryption algorithm. SymmetryBasel (2020). https://doi.org/10.3390/sym12020189

[6] Wardak, O., Sinha, D. & Sethi, A. Encryption and decryption of signed graph matrices through RSA algorithm. Indian J Pure Appl Math (2023). https://doi.org/10.1007/s13226-023-00452-9

[7] Kalaichelvi, V. et al. 'Design of Digital Image Encryption Based on Elliptic Curve Cryptography (ECC) Algorithm and Radix-64 Conversion'. 1 Jan. 2022 : 6697 – 6708.https://doi.org/10.3233/JIFS-220767

[8] Lorenz E.N.Deterministic non-periodic follow[J].Journal of the Atmospheric Sciences,1963,20:130-141.

[9] R.Matthews.On the derivation of a"chaotic"encryption algorithm[J].Cryptologia,1989,13(1):29-42.

[10] Talhaoui, M.Z., Wang, X. & Midoun, M.A. A new one-dimensional cosine polynomial chaotic map and its use in image encryption. Vis Comput 37, 541–551 (2021). https://doi.org/10.1007/s00371-020-01822-8

[11] Li, C., Xie, T., Liu, Q. et al. Cryptanalyzing image encryption using chaotic logistic map. Nonlinear Dyn 78, 1545–1551 (2014). https://doi.org/10.1007/s11071-014-1533-8

[12] Luo, Y., Yu, J., Lai, W. et al. A novel chaotic image encryption algorithm based on improved baker map and logistic map. Multimed Tools Appl 78, 22023–22043 (2019). https://doi.org/10.1007/s11042-019-7453-3

[13] T. Li, B. Du and X. Liang, "Image Encryption Algorithm Based on Logistic and Two-Dimensional Lorenz," in IEEE Access, vol. 8, pp. 13792-13805, 2020,doi:https://doi.org/10.1109/ACCESS.2020.2966264.

[14] Al-Hazaimeh, O.M., Al-Jamal, M.F., Alhindawi, N. et al. Image encryption algorithm based on Lorenz chaotic map with dynamic secret keys. Neural Comput & Applic 31, 2395–2405 (2019). https://doi.org/10.1007/s00521-017-3195-1

[15] Bouteghrine, B., Tanougast, C. & Sadoudi, S. Novel image encryption algorithm based on new 3-d chaos map. Multimed Tools Appl 80, 25583–25605 (2021). https://doi.org/10.1007/s11042-021-10773-8

[16] Wang, X., Xu, M. & Li, Y. Fast encryption scheme for 3D models based on chaos system. Multimed Tools Appl 78, 33865–33884 (2019). https://doi.org/10.1007/s11042-019-08171-2

[17] Peng Zai-Ping, Wang Chun-Hua, Lin Yuan, Luo Xiao-Wen. A novel four-dimensional multiwing hy-per-chaotic attractor and its application in image encryption. Acta Phys. Sin., 2014, 63(24): 240506. doi:.http://dx.doi.org/10.7498/aps.63.240506 .

[18] Çavuşoğlu, Ü., Kaçar, S., Zengin, A. et al. A novel hybrid encryption algorithm based on chaos and S-AES algorithm. Nonlinear Dyn 92, 1745–1759 (2018). https://doi.org/10.1007/s11071-018-4159-4

[19] Arab, A., Rostami, M.J. & Ghavami, B. An image encryption method based on chaos system and

AES algorithm. J Supercomput 75, 6663–6682 (2019). https://doi.org/10.1007/s11227-019-02878-7

[20] Song, C.; Qiao, Y. A Novel Image Encryption Algorithm Based on DNA Encoding and Spatiotemporal Chaos. Entropy 2015, 17, 6954-6968. https://doi.org/10.3390/e17106954

[21] Lone, P.N., singh, D. & Mir, U.H. Image encryption using DNA coding and three-dimensional chaotic systems. Multimed Tools Appl 81, 5669–5693 (2022). https://doi.org/10.1007/s11042-021-11802-2

[22] Zhang, D., Liao, X., Yang, B. et al. A fast and efficient approach to color-image encryption based on compressive sensing and fractional Fourier transform. Multimed Tools Appl 77, 2191–2208 (2018). https://doi.org/10.1007/s11042-017-4370-1

[23] Fei Yu, Li Liu, Shuai Qian, Lixiang Li, Yuanyuan Huang, Changqiong Shi, Shuo Cai, Xianming Wu,Sichun Du, Qiuzhen Wan, "Chaos-Based Application of a Novel Multistable 5D Memristive Hyperchaotic System with Coexisting Multiple Attractors", Complexity, vol. 2020, Article ID 8034196, 19 pages, 2020. https://doi.org/10.1155/2020/8034196

[24] Zhang, X., Fan, X., Wang, J. et al. A chaos-based image encryption scheme using 2D rectangular transform and dependent substitution. Multimed Tools Appl 75, 1745–1763 (2016). https://doi.org/10.1007/s11042-014-2372-9

[25] Liang, X., Tan, X. & Tao, L. Plaintext related image hybrid encryption scheme using algebraic interpolation and generalized chaotic map. Multimed Tools Appl 79, 2719–2743 (2020). https://doi.org/10.1007/s11042-019-08295-5

[26] Es-Sabry, M., El Akkad, N., Merras, M. et al. A new image encryption algorithm using random numbers generation of two matrices and bit-shift operators. Soft Comput 24, 3829–3848 (2020). https://doi.org/10.1007/s00500-019-04151-8

[27] Parveiz Nazir Lone, Deep Singh & Umar Hussain Mir (2021) A novel image encryption using random matrix affine cipher and the chaotic maps, Journal of Modern Optics, 68:10, 507-521, doi:https://doi.org/10.1080/09500340.2021.1924885.

[28] Sabir, S., Guleria, V. Multi-layer color image encryption using random matrix affine cipher, RP2DFrHT and 2D Arnold map. Multimed Tools Appl 80, 27829–27853 (2021). https://doi.org/10.1007/s11042-021-11003-x

[29] Y. Li, C. Wang, and H. Chen, “A hyper-chaos-based image encryption algorithm using pixel-level permutation and bitlevel permutation,” Optics and Lasers in Engineering, vol. 90, pp. 238–246, 2017.https://doi.org/10.1016/j.optlaseng.2016.10.020