A Novel Hyper-Chaos-Based Image Encryption Algorithm Using Bit-Level Permutation and Pixel-Level Diffusion

Yajuan Li[a], Ruisong Ye[1,a], Yucheng Chen[a]
[a]*Department of Mathematics, Shantou University, Shantou, Guangdong, 515063, P. R. China*

Abstract

In this paper, a 4D Lorenz map is proposed using in cryptography. Performance evaluations show that it has hyper-chaotic behavior, wide chaotic range and large complexity. Based on this map, a novel image encryption algorithm is designed by employing bit-level permutation and pixel-level diffusion. The bit-level permutation is performed by chaotic sequence, and the bit-level diffusion is carried out by Arithmetic plus. Besides, to achieve the better ability of resisting chosen-plaintext or known-plaintext attack, the substitution key stream generated using SHA-256 in our method is dependent on the plain image. Consequently, different plain images produce the distinct key stream for substitution. The simulation results and performance analysis show that the proposed image encryption algorithm is both secure and reliable for image encryption.

Keywords - *4-D hyper-chaotic maps, Bit-level, Image encryption, SHA-256*

I. INTRODUCTION

With the rapid development in internet technology and multimedia, multimedia communication has become more and more important, and images sharing online becomes increasingly popular and common. Therefore, information security issue is urgent and faces a great challenge.However, due to bulky data capacity, highredundancy and strong correlations among adjacent pixels, traditionalencryption algorithms, such as DES and AES, are poorly suited toimage encryption [1-2]. Therefore, many researchers began to investigate efficient and secure image encryption schemes. In 1989, Matthews first proposed encryption scheme based on chaotic system [3]. In 1997, Fridrich applied chaotic mapping to image encryption system [4]. The permutation-substitution which was first proposed by Fridrich is the most widely used architecture for image encryption [5]. Under this structure, the pixel positions are firstly shuffled in the permutation process for the sake of decreasing the strong correlation between pixels adjacent to each other. After that, values of the pixels are changed one by one in the substitution process to achieve the avalanche effect [6].

To improve the security of image security, researchers have presented many effective imagecryptographicsuch as DNA-coding [7-8], mixed image encryption [9], cellular automata [10], image filtering [11] and so on. Among these techniques, chaos-based ones are the most widely used [1,10-14], because chaotic maps have the properties of initial state sensitivity, unpredictability and ergodicity, and these properties can be found similar counterparts in image cipher [4-5]. Some examples of image encryption using chaotic maps are as follows. In [12], a colorimage encryption using combination of the 1D chaotic map has been proposed by Pak et al. The encryptionscheme has defined the new chaotic system structure, combined two Sine maps in the permutation stage, and utilized the key-streams generated by the Sine-Sine-map to confuse the permuted image. However, it was Cracked by Hui Wanga [13]. In [6], The article constructs a cryptosystem using discrete 2D Henon map, and in [14], a new 2D Logistic ICMIC cascade map (2D-LICM) is proposed based on cascade modulation couple (CMC) model. Compared with 1D chaotic maps, they generally contain one variable and few parameters and their orbits are simple, so their parameters and initial values may be easily estimated [12].When those maps are used to design image encryption schemes, the cryptosystems are unsafe [15]. On the other hand,high dimensional chaotic maps have more variables and parameters and usually shows good hyper-chaos natures, which is more suitable to encryption.

Compared with pixel-level permutation, the bit-level permutation not only changes pixel positions, but also alters pixel values [14]. So it has better encryption effect. For the sake of achieving the better ability of resisting chosen-plaintext orknown-plaintext attack, the substitution keystream generated in the proposed method is dependent on the plain image. As aresult, different plain images produce the distinct keystream for substitution. It is difficult for an attacker to obtain any useful information about the keystream by a number of possible plain-image or cipher-image pairs [6, 8].

The rest of this paper is organized as follows. Section 2 presents the basic theoryabout 4D hyper-Lorenz map. Section 3 describes the proposed cryptosystem. Section 4 shows the computer

simulations, security and performance analyses, and conclusions are given in Section 5.

II. CHAOTIC SYSTEM

This section presents the 4D Lorenz modulation map, which is defined by

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

Where a, b, c, r arereal constant parameters of system (1). The initial system variables $x_0 \in (-40, 40)$, $y_0 \in (-40, 40)$, $z_0 \in (1, 81)$, $w_0 \in (-250, 250)$ are used as the cipher key. When$a = 10$, $b = 8/3$, $c = 28$, $-1.52 \leqslant r \leqslant -0.06$,system (1) is in a chaotic state，andwhen$r = -1$, it has four Lyapunov exponents, $\lambda_1 = 0.3381$, $\lambda_2 = 0.1586$, $\lambda_3 = 0$, $\lambda_4 = -15.1752$ .As the system has two positive Lyapunov exponents, it is obvious that system (1) exhibits hyper-chaotic behavior. The projection phase diagrams of the attractor are shown in Fig. 1.

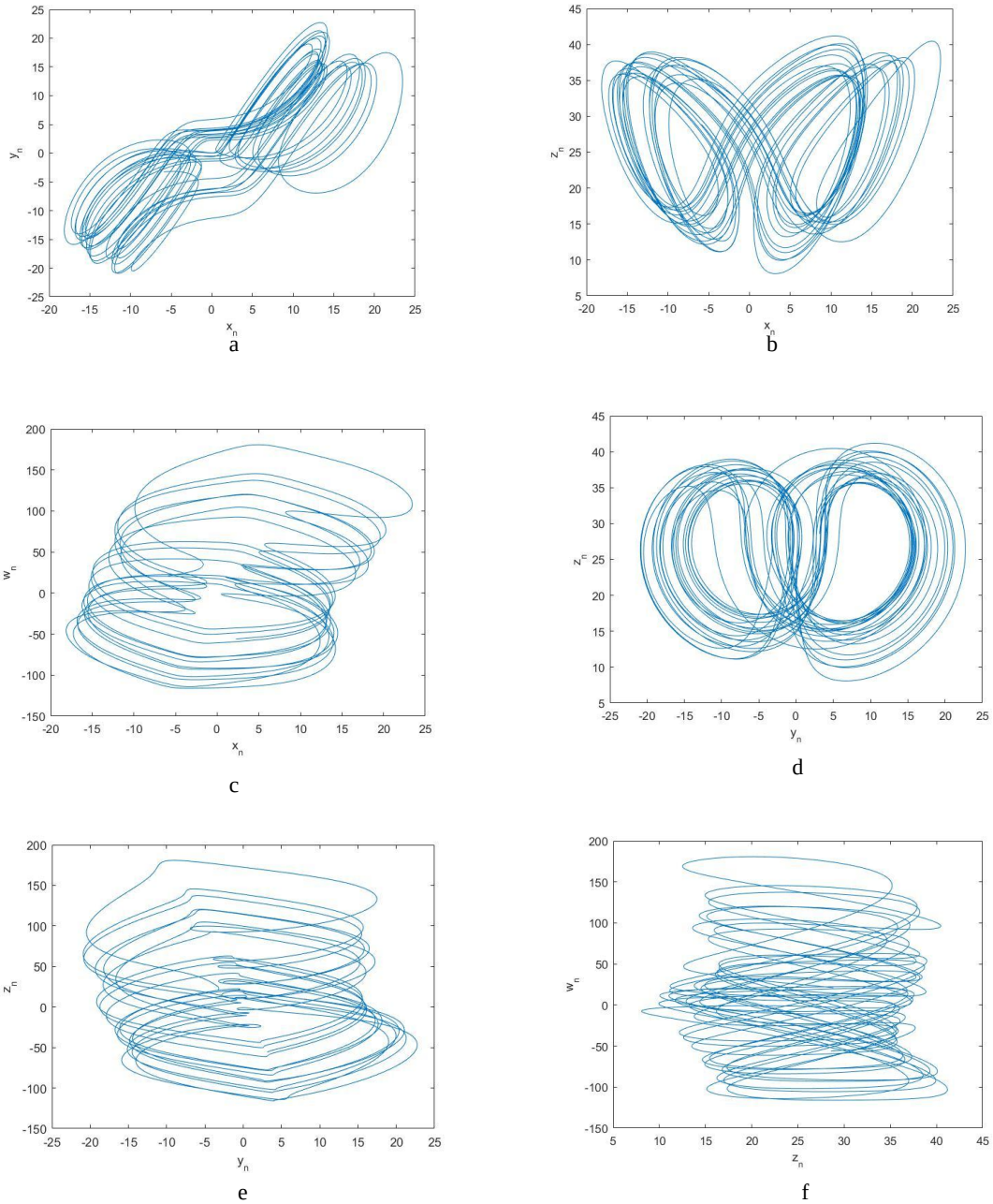


Fig. 1 Projections of the chaotic attractor of system (1), a: x–y, b: x–z, c: x–w, d: y–z, e: y–w, f: z–w.

III.PROPOSED SYSTEM

A. Key Stream Generation Algorithm

For satisfactory plaintext attack resistance, plaintext-related key stream generation has been adopted in the recently proposed image cryptosystems [6,8,17-18], The most fashion way is to extract an intrinsic feature of the plaintext and then disturb the original secret key, so that the produced chaotic variables and further the generated key stream are not only related to the input secret key but also to the plain image. In other words, different plaintexts will bring about distinct key streams, with which the adversary cannot correctly recover other ciphertexts. The common extraction approaches include hashing, summing the pixels, calculating Hamming distance, specifying certain pixel, and so on [8]. In this paper, the generations of seeds for chaotic maps are dependent on common key and SHA-256 of the plain image, similar existing algorithms described in [7, 19-21]. Let the SHA-256 hash value of the plain image be d, according to the order of bits, we segment d into four groups, each consists of sixteen hexadecimal numbers, i.e., d_1, d_2, d_3, d_4. Each group is converted to a floating point decimal number $h_i \in (0,1),\ i = 1,...,4$, by Eq. (2). The method to generate these keys is expressed clearly in Fig. 2.

$$h_i = hex2dec(d_i)/2^{64}, where\ i = 1,...,4. \quad (2)$$

Now, input the initial value x_0, y_0, z_0, w_0 from user as secret key, add all these secret keys to get one common key as follows,

$$u = x_0 + y_0 + z_0 + w_0 \ \ mod\ 1. \quad (3)$$

The new initial value x_0, y_0, z_0, w_0 are calculated as follows that will be used to iterate for Eq. (1).

$$\begin{aligned} x_0' &= h_1 + x_0 + u \ \ mod\ 40; \\ y_0' &= h_2 + y_0 + u \ \ mod\ 40; \\ z_0' &= h_3 + z_0 + u \ \ mod\ 81; \\ w_0' &= h_4 + w_0 + u \ \ mod\ 250; \end{aligned} \quad (4)$$

The detailed description of key stream generation for encryption is shown in algorithm 1.

B. Encryption Algorithm

The encryption process in the paper is shown in Fig. 4. First, thechaotic sequence generated by chaotic system is relevant to characteristicsof plain-image. Then, a gray plain image is decomposed into 8 bit-planes by means of binary bit-plane decomposition (BBD) [22], a bit-level permutation is employed toshuffle the plain-image. Next, a pixel-level image is obtained by binary bit-plane composition (BBC), a pixel- level diffusion is utilized tostrengthen the security of the cryptosystem. Finally, we can get the cipher-image. The detailed encryption process is illustrated in algorithm 2.

C. Decryption Algorithm

Since the proposed encryption algorithm is a kind of the symmetric cryptosystem, it is easy to implement the decryption process in a way that is the reverse procedure of image encryption illustrated above.

Algorithm 1: Key stream generation

Input: initial values for hyper-chaotic Lorenz map x_0, y_0, z_0, w_0 and parameters rr which used to avoid transient effects and the plain image P

Output: $x_{(1,M)}, y_{(1,N)}, z_{(1,8)}, w_{M\times N}$ will used in the whole system

1. set parameters for hyper-chaotic Lorenz mapwith $a = 10, b = 8/3, c = 28, r = -1$;
2. update the initial conditions x_0, y_0, z_0, w_0 according to Section 3.1;
3. iterate the hyper-chaotic Lorenz map with the updated initial conditions x_0, y_0, z_0, w_0, and get four sequence x_1, y_1, z_1, w_1, update them using $x_1 = x_1 - \lfloor x_1 \rfloor, y_1 = y_1 - \lfloor y_1 \rfloor$, $z_1 = z_1 - \lfloor z_1 \rfloor, w_1 = w_1 - \lfloor w_1 \rfloor$, $\lfloor\ \rfloor$ means round function.
4. select x, y, z, w using $x = x_1(rr+1:rr+M)$ $y = y_1(rr+1:rr+N)$ $z = z_1(rr+1:rr+8)$, $w = w_1(rr+1:rr+M\times N)$;
5. change w using $w = \lceil w \times 10^{14} \rceil \ \ mod\ 256$, and make the size of w is $M \times N$, $\lceil\ \rceil$ means floor function of w, M, N means the size of plain image;

D. Simulation Results

Setting secret key K. The plain, encrypted and decrypted images of Lena, Clock, Elaine whose sizes are 256×256, 512×512, 256×256 respectively are shown in Fig. 5 (a)~ (c), (g)~(i), (m)~(o). Obviously, the encrypted images are all noise-like images and the deciphered images are the same as the plain images. It indicates this algorithm is effective.

$K = [x_0 = 1.452416, y_0 = 1.78256, z_0 = 11.28941,$
$w_0 = 1.98672, rr = 2000];$

IV.SECURITY ANALYSIS

A. Statistical Analysis

The ability to resist statistical attack of an algorithm is assessed by histogram, correlation between adjacent pixels and information entropy [14].

1. Histogram

Histogram represents the distribution of pixel intensity values about the image. Fig. 5 (d)~(f), (j)~(l) showthe histograms of cipher images and plain images. It is obvious by visual inspection that values

of cipher images distribute uniformly among the interval of [0, 255] which is totally different with plain images. That means this algorithm is strong enough to resist the powerful attack of statistical analysis.

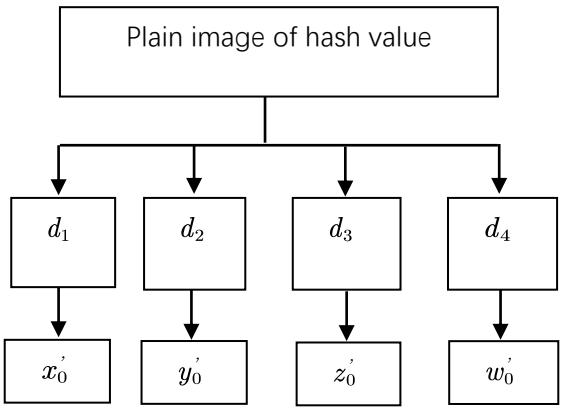


Fig. 2 The method to generate keys

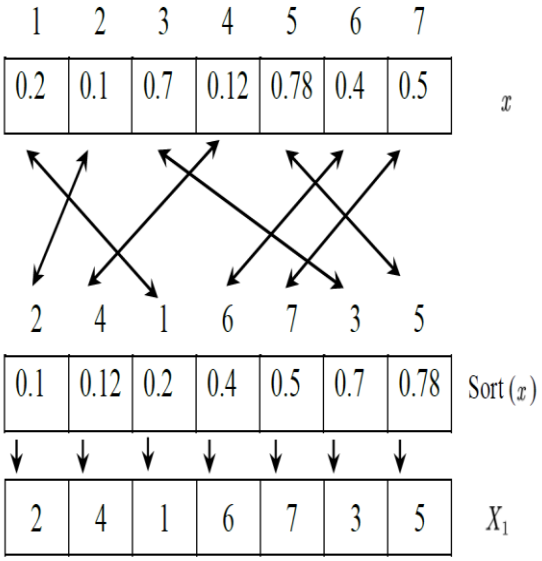


Fig. 3 The diagram of (a) the generating of permutation position matrix

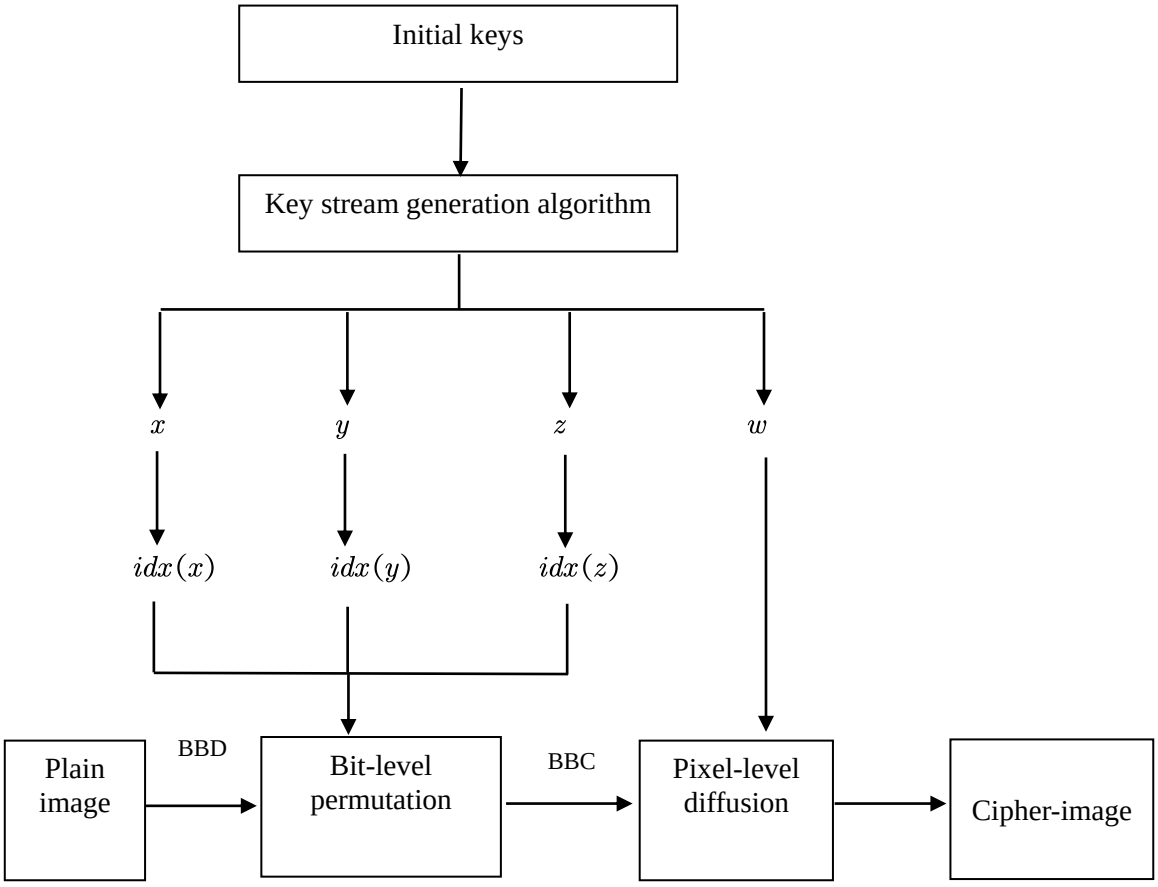


Fig. 4 The encryption process

Algorithm 2: Image encryption

Input: the plain image $P_{M\times N}$ and parameters $r_1 = 77$

Output: Cipher image $C_{M\times N}$

1. obtain chaotic sequence x,y,z,w from the key stream generation algorithm;
2. initialization:
   $A = zeros(M,N,8), D = zeros(M,N)$
3. P is decomposed into 8 bit-planes record as $PBit$, and x,y,z is sorted in ascending order. According to the value position in the initial sequence, we can obtain sequence $X_{1_{(1,M)}}, Y_{1_{(1,N)}}, Z_{1_{(1,8)}}$. The detailed describe is shown in Fig. 3;
4. for $i = 1:M$
5. for $j = 1:N$
6. for $u = 1:8$
7. Permute the image bit-level position;
8. $A(i,j,u) = PBit(X_1(i), Y_1(j), Z_1(u))$;
9. end
10. end
11. end
12. B is obtained by binary bit-plane composition (BBC) using A;
13. a pixel-level diffusion using Arithmetic plus and mod;
14. for $i = 1:M$
15. for $j = 1:N$
16. if $i == 1$ and $j == 1$ then
17. $D(i,j) = (B(i,j) + w(i,j) + r1) \ mod \ 256$;
18. elseif $j == 1$
19. $D(i,j) = (B(i,j) + sum(D(i-1,:)) + w(i,j)) \ mod \ 256$;
20. else
21. $D(i,j) = (B(i,j) + D(i,j-1) + w(i,j)) \ mod \ 256$;
22. end
23. end

2. *Correlation Between Adjacent Pixels*

Generally, the images of the plaintext have strong correlation between the adjacent pixels in the horizontal, vertical, positive directions, and there should be no correlation between the adjacent pixels in the cipher-text image. Therefore, a good encryption algorithm should have the ability to break correlations between adjacent pixels. Mathematically, adjacent pixels correlation coefficients are defined by Eq. (5), where u, v are data sequences.

$$r_{xy} = \frac{cov(u,v)}{\sqrt{D(u)}\ \sqrt{D(v)}}$$

$$cov(u,v) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(u))\,(y_i - E(v))$$

$$D(u) = \frac{1}{N}\sum_{i=1}^{N}(u_i - E(u))^2 \qquad (5)$$

$$E(u) = \frac{1}{N}\sum_{i=1}^{N}u_i$$

We select N adjacent pixels from inspecting image arbitrarily, recording their value as (u_i, v_j), $i = 1...N$. Calculate their correlation by Eq. (5), shown in Fig. 6, larger values of r_{xy} means higher correlation. Apparently, most points are close to the diagonal line of axis for the plain image. However, points distribute randomly on the whole space for the cipher image. Table.1 shows quantitative results from different images. Clearly, the r_{xy} value of the proposed

Table. 1 Correlation coefficients of the plain and cipher images.

Image		horizontal	vertical	Diagonal
Lena	Plain	0.9724	0.9377	0.9160
	Cipher	0.0022	-0.0062	0.0026
Elaine	Plain	0.9732	0.9754	0.9689
	Cipher	-0.0113	0.0037	-0.0039
Man	Plain	0.9703	0.9562	0.9453
	Cipher	0.0045	-0.0263	0.0033

Table. 2 Information entropy for encrypted images.

	Lena	Man	Elaine
Plain Image	7.5683	7.3574	7.5060
Cipher Image	7.9973	7.9993	7.9994

scheme is close to 0. It means that the proposed algorithm breaks adjacent pixels correlation thoroughly.

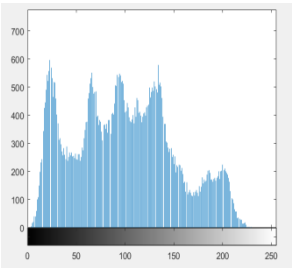
Table. 3 The NPCR and UACI for decrypted image using different keys (Man)

	NPCR	UACI
Key1	99.5998	33.4401
Key2	99.5936	33.4719
Key3	99.6071	33.5278
Key4	99.5998	33.4343
Key5	99.6155	33.4485

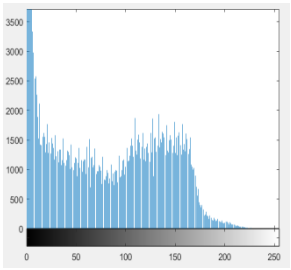
(a)

(b)

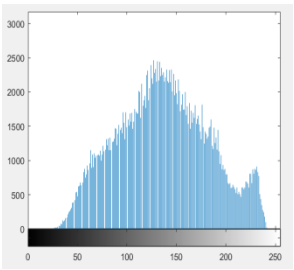
(c)



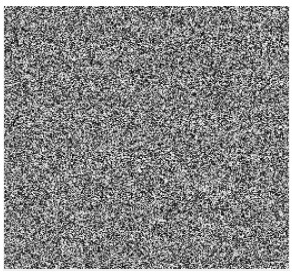
(d)



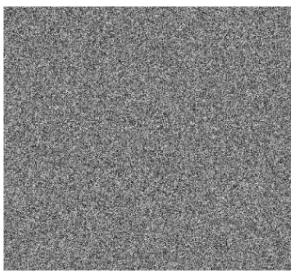
(e)



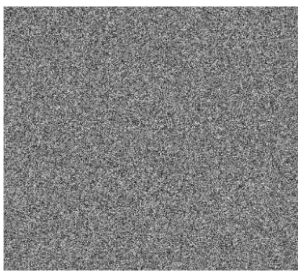
(f)



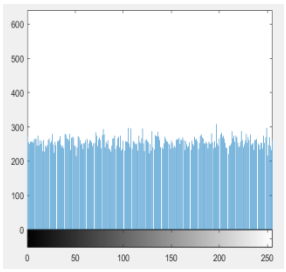
(g)



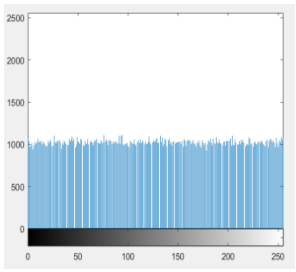
(h)



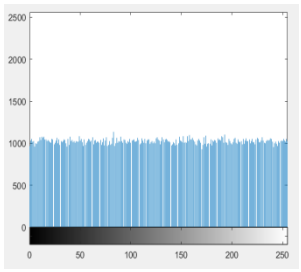
(i)



(j)



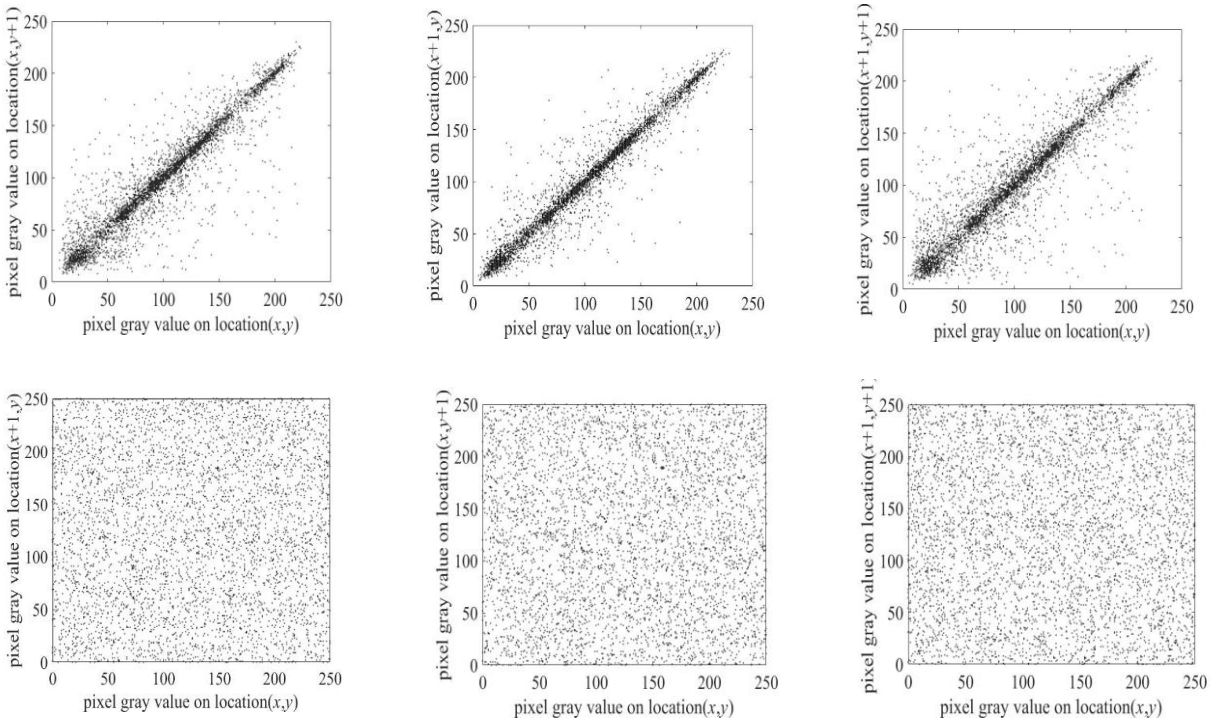
(k)



(l)

(m) (n) (o)

Fig. 5. (a)~(c) Plain images Lena, Man, Elaine; (d)~(f) the histogram of (a)~(c); (g)~(i) the encrypted images of (a)~(c); (j)~(l) the histogram of (g)~(i); (m)~(o) the decryption of (g)~(i).



(a) (b) (c)

Fig. 6 Distribution of adjacent pixels. The first row shows the plain image, the second row shows the cipher image. (a) Horizontal direction. (b) Vertical direction. (c) Diagonal direction.

3. *Information Entropy*

If a pair of images has L gray values $m_i(i=1...L-1)$, and the probability of each gray value is $p(m_i)$, $i=1...L-1$, respectively. According to the Shannon theorem, the amount of information of the image is as follows

$$H(m)=\sum_{i=0}^{L-1}p(m_i)\log_2^{p(m_i)} \qquad (6)$$

$$\sum_{i=0}^{L-1}p(m_i)=1$$

H is called Information entropy of image. When the probability of each gray value appears in the image is equal, the information entropy of the image is the largest. Information entropy is one of criteria to measure the image randomness. For a 256 gray level image, the ideal information entropy value is 8. The results are listed in Table. 2. Obviously, the information entropy values of the cipher images are extremely close to 8, which means that the cipher images are random from global perspective.

B. Key Space

The key space is the set of all the legal keys, and the key space of the image cryptography system should be large enough, to combat brute-force attacks effectively. Especially encryption and decryption of very fast password system, its password length

should be at least $128b$. In the propose cryptosystem, the cipher keys consist of x_0, y_0, z_0, w_0, rr .The computational precision of double-precision number is taken as10^{-16}.The valid choices of initial values x_0, y_0, z_0, w_0 are all 10^{16}, and that of rr is 10^3, therefore the total key space is at least $L = \log_2^{(10^{16})^4} = \log_2^{1.0000e+80} \approx 213b$. This value is far greater than $128b$ .It is clear that the encryption algorithm has a sufficiently large key space to resist all types of brute-force attacks.

C. *Key Sensitivity*

To guarantee the security of the cryptosystem, a good cryptosystem should be sensitive to the key. The incorrect plain image will be produced when different keys areused to decrypt the cipher-image. We use the original key to encrypt theLena image and the slightly modified key to decrypt the cipher-image. Figure. 7 shows the encryption sensitivity test, using key and key1~key5 to decrypt the Lena graph encrypted by key, respectively. Table. 3 present the NPCR and UACI for decrypted image using kye1 to key5 compared with encrypted image using key.

key= $[x_0 = 1.452416, y_0 = 1.78256, z_0 = 11.28941, w_0 = 1.98672, rr = 2000]$;

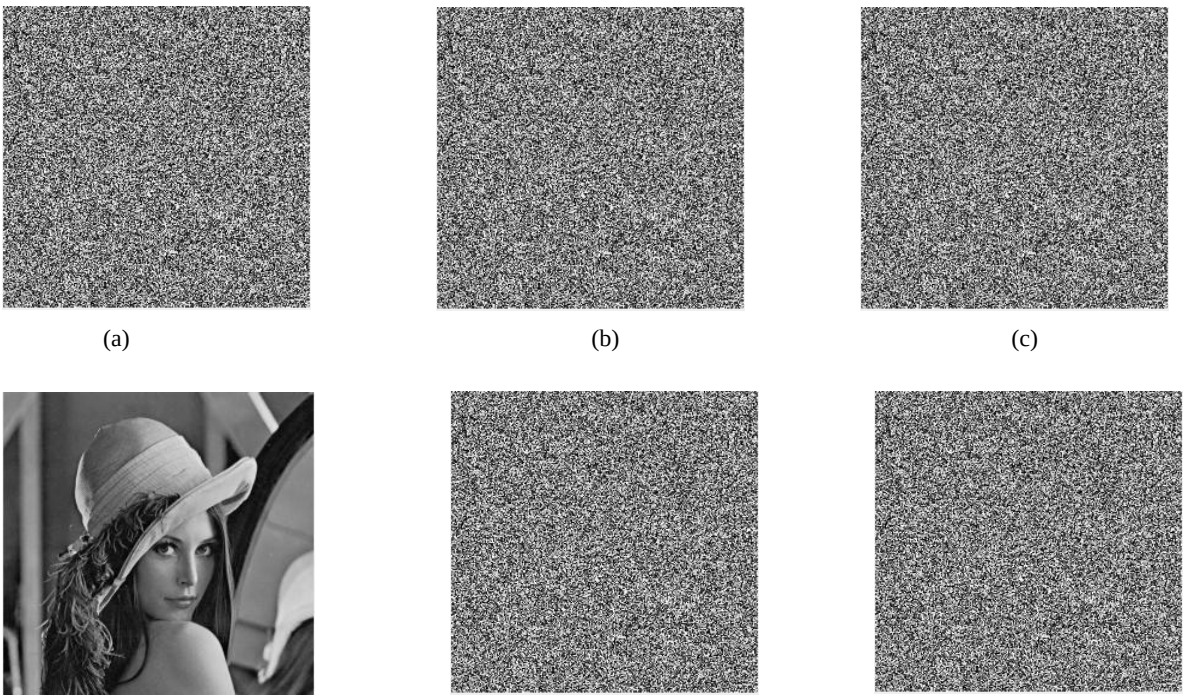
key1=$[x_0 = 1.45241600000001, y_0 = 1.78256, z_0 = 11.28941, w_0 = 1.98672, rr = 2000]$;

key2=$[x_0 = 1.452416, y_0 = 1.78256000000001, z_0 = 11.28941, w_0 = 1.98672, rr = 2000]$;

key3=$[x_0 = 1.452416, y_0 = 1.78256, z_0 = 11.28941000000001, w_0 = 1.98672, rr = 2000]$;

key4=$[x_0 = 1.452416, y_0 = 1.78256, z_0 = 11.28941, w_0 = 1.98672000000001, rr = 2000]$;

key5=$[x_0 = 1.452416, y_0 = 1.78256, z_0 = 11.28941, w_0 = 1.98672, rr = 2001]$;



(a) (b) (c)

(d) (e) (f)

Fig. 7 Secret key sensibility in the decryption process. (a)=Dec(C,key), (b)=Dec(C,key1), (c)=Dec(C,key2), (d)=Dec(C,key3), (e)=Dec(C,key4), (f)=Dec(C,key5)

D. *Differential Analysis*

The differential attack mechanism aims to introduce a tiny modification in the original image, then discover the

difference between two encrypted images. Through comparing the difference, the attacker is able to discover the

relation between original image and encrypted image. Therefore, two well-known measurements named Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) are

employed for evaluating the

performance of resisting against differential attack [23]. They are calculated by

$$NPCR = \frac{1}{W \times H}\sum_{i=0}^{H}\sum_{j=0}^{W} D(i,j) \times 100\%$$

$$UACI = \frac{1}{W \times H}\sum_{i=0}^{H}\sum_{j=0}^{W} \frac{|c_1(i,j) - c_2(i,j)|}{255}$$

$$D(i,j) = \begin{cases} 0 & \text{if } c_1(i,j) = c_2(i,j) \\ 1 & \text{if } c_1(i,j) = c_2(i,j) \end{cases}$$

(7)

where W and H stand for the image size. $c_1(i,j)$ and $c_2(i,j)$ are gray-values of the pixels at position (i,j) of two

encrypted images before and after one pixel is changed in the plain image.Tests are carried out using distinct image such as Lena, Man. For each test image, 1000 pixelsare selected randomly and changed to generate new plain images. Then, we use the same secret key to encryptboth plain images varied in only one pixel [6]. Table 4,5 gives the minimum, maximum, and mean values of UACI and NPCR for different images. As observed, the NPCR and UACIare obtained as shown in Fig. 8.

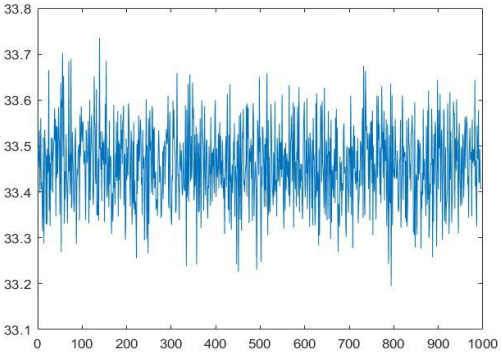
Table. 4 The UACI results for different image

Cipher	minimum	maximum	mean
Lena	33.2063	33.7858	33.4502
Man	33.3012	**33.5734**	33.4479

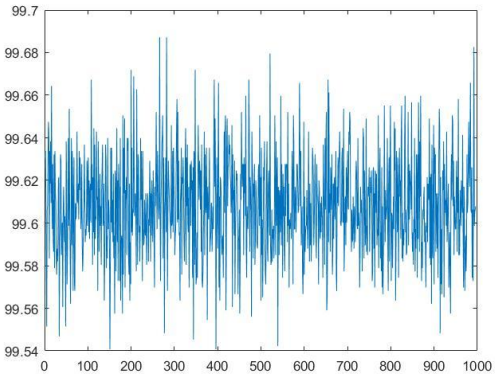
Table. 5 The NPCR results for different image

Cipher	minimum	maximum	mean
Lena	99.5316	99.6887	99.6098
Man	99.5712	99.6456	99.6099

These test results prove that value of NPCR and UACI produced byour proposed algorithm is extremely close to the expected values [2,24-25] NPCR Expected = 99.6094% and UACI Expected = 33.4635%. Besides, the minimum and maximum NPCR and UACIobtained from our algorithm are more close to the mean value. These indicate that the proposed method achieves anexcellent diffusion property that it is very sensitive to the plain image. So it is able to resistance the differential attack.



(a) UACI



(b) NPCR

Fig. 8 The NPCR and UACI of 1000 images that only change one pixel. (a) NPCR (b)UACI.

V. CONCLUSIONS

In this paper, we propose a hyper-chaos Lorenz map based image encryptionalgorithm using bit-level permutation and pixel-level permutation.Itcan overcome the common weaknesses of the algorithm based on low dimensionalchaotic map [12-13,26] as it is based on a hyper-chaotic system.And to achieve the better ability ofresisting chosen-plaintext or known-plaintext attack, the substitution keystream generated in our method is dependenton the plain image. Then, bit-level permutation and pixel-level diffusion are employedto strengthen security of the cryptosystem. What's more, we carry out manyexperiments, including histogram analysis, key sensitivity analysis,key space analysis, correlation analysis and differential analysis toshow that the proposed algorithm is secure and reliable for imageencryption.

REFERENCE

[1] Li, S., Chen, G., Cheung, A., Bhargava, B., & Lo, K. T. (2007). On the design of perceptual MPEG-video encryption algorithms. IEEE Transactions on Circuits and Systems for Video Technology, 17(2), 214-223.

[2] Li, Y., Wang, C., & Chen, H. (2017). A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation. Optics & Lasers in Engineering, 90, 238-246.

[3] Matthews, R. (1989) On the Derivation of a "Chaotic" Encryption Algorithm[J]. Cryptologia, 8(8), 29-42.

[4] Fridrich, J. (1997). Image encryption based on chaotic maps. IEEE International Conference on Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation (Vol.2, pp.1105-1110 vol.2). IEEE.

[5] Fridrich, J. (1998). Symmetric ciphers based on two-dimensional chaotic maps. International Journal of Bifurcation and chaos, 8(06), 1259-1284.

[6] Ping, P., Xu, F., Mao, Y., & Wang, Z. (2018). Designing permutation–substitution image encryption networks with Henon map. Neurocomputing, 283, 53-63.

[7] Liao, X., Hahsmi, M. A., & Haider, R. (2018). An efficient mixed inter-intra pixels substitution at 2bits-level for image encryption technique using DNA and chaos. Optik-International Journal for Light and Electron Optics, 153, 117-134.

[8] Chen, J., Zhu, Z. L., Zhang, L. B., Zhang, Y., & Yang, B. Q. (2018). Exploiting self-adaptive permutation–diffusion and dna random encoding for secure and efficient image encryption. Signal Processing, 142, 340-353.
[9] Zhang, X., & Wang, X. (2017). Multiple-image encryption algorithm based on mixed image element and chaos ☆. Computers & Electrical Engineering, 92, 6-16.
[10] Yang, Y. G., Tian, J., Lei, H., Zhou, Y. H., & Shi, W. M. (2016). Novel quantum image encryption using one-dimensional quantum cellular automata. Information Sciences, 345, 257-270.
[11] Hua, Z., & Zhou, Y. (2017). Design of image cipher using block-based scrambling and image filtering. Information Sciences, 396, 97-113.
[12] Pak, C., & Huang, L. (2017). A new color image encryption using combination of the 1D chaotic map. Signal Processing, 138, 129-137.
[13] Wang, H., Xiao, D., Chen, X., & Huang, H. (2018). Cryptanalysis and enhancements of image encryption using combination of the 1D chaotic map. Signal Processing, 144, 444-452.
[14] Cao, C., Sun, K., & Liu, W. (2018). A novel bit-level image encryption algorithm based on 2D-LICM hyperchaotic map. Signal Processing, 143, 122-133.
[15] Li, C., Liu, Y., Zhang, L. Y., & Chen, M. Z. (2013). Breaking a chaotic image encryption algorithm based on modulo addition and XOR operation. International Journal of Bifurcation and Chaos, 23(04), 1350075.
[16] Zhang, Y., Xiao, D., Shu, Y., & Li, J. (2013). A novel image encryption scheme based on a linear hyperbolic chaotic system of partial differential equations. Signal Processing: Image Communication, 28(3), 292-300.
[17] Wang, X. Y., Zhang, Y. Q., & Bao, X. M. (2015). A novel chaotic image encryption scheme using DNA sequence operations. Optics and Lasers in Engineering, 73, 53-61.
[18] Li, X., Wang, L., Yan, Y., & Liu, P. (2016). An improvement color image encryption algorithm based on DNA operations and real and complex chaotic systems. Optik-International Journal for Light and Electron Optics, 127(5), 2558-2565.
[19] Kulsoom, A., Xiao, D., & Abbas, S. A. (2016). An efficient and noise resistive selective image encryption scheme for gray images based on chaotic maps and DNA complementary rules. Multimedia Tools and Applications, 75(1), 1-23.
[20] Liao, X., Kulsoom, A., & Ullah, S. (2016). A modified (Dual) fusion technique for image encryption using SHA-256 hash and multiple chaotic maps. Multimedia Tools and Applications, 75(18), 11241-11266.
[21] Liu, H., & Wang, X. (2012). Image encryption using DNA complementary rule and chaotic maps. Applied Soft Computing, 12(5), 1457-1466.
[22] Zhou, Y., Cao, W., & Chen, C. P. (2014). Image encryption using binary bitplane. Signal Processing, 100, 197-207.
[23] Wu, Y., Zhou, Y., Noonan, J. P., &Agaian, S. (2014). Design of image cipher using latin squares. Information Sciences, 264, 317-339.
[24] Patidar, V., Pareek, N. K., Purohit, G., & Sud, K. K. (2011). A robust and secure chaotic standard map based pseudorandom permutation-substitution scheme for image encryption. Optics Communications, 284(19), 4331-4339.
[25] Wu, Y., Noonan, J. P., &Agaian, S. (2011). 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), 31-38.
[26] Wang, X., Luan, D., & Bao, X. (2014). Cryptanalysis of an image encryption algorithm using Chebyshev generator. Digital Signal Processing, 25, 244-247.