Hindawi
Security and Communication Networks
Volume 2020, Article ID 8841302, 13 pages
https://doi.org/10.1155/2020/8841302

Research Article

3-Image Bit-Level Encryption Algorithm Based on 3D Nonequilateral Arnold Transformation and Hyperchaotic System

Huiqing Huang[1] **and Dongsheng Cheng**[2]

[1] *School of Mathematics, Jiaying University, Meizhou 514015, Guangdong, China*
[2] *School of Software Engineering, Shenzhen Institute of Information Technology, Shenzhen 518172, Guangdong, China*

Correspondence should be addressed to Dongsheng Cheng; chengds@sziit.edu.cn

Received 2 September 2020; Revised 28 October 2020; Accepted 16 November 2020; Published 26 November 2020

Academic Editor: Sen Bai

In this paper, we propose a novel 3-image bit-level encryption algorithm based on 3D nonequilateral Arnold transformation and hyperchaotic system. Firstly, the three plain images with $N \times M$ are decomposed into 8-bit planes and then they overlap into a 3D bit matrix with size $N \times M \times 24$. Then, the 3D bit matrix is scrambled by 3D nonequilateral Arnold transformation and the scrambled 3D bit matrix is integrated and transformed into three 2D pixel-level images. Finally, the hyperchaotic system is used to diffuse the three 2D pixel-level images; then three diffused images are rearranged to be one color image, resulting in the encrypted image. Numerical simulations and analyses of the proposed encryption scheme are given to validate the feasibility and safety of the method. The statistical analyses like histogram, correlation, and entropy confirm that the proposed method can effectively resist statistical attacks and security key analysis shows that the key space is large enough to render the brute-force attack ineffective in proposed method. The differential analysis confirms that the proposed method is effective against differential attacks and the results of the experiment confirmed that the method can resist occlusion attack.

1. Introduction and Related Works

Image is an important carrier for human visual system to obtain rich information and is one of the important ways of information transmission nowadays. As the most important carrier of information, hundreds of millions of images are generated, stored, and transmitted over the network every day. How to ensure the security of image has become a hot topic in recent years. Image encryption is one of the important means to ensure image security. As early as 1989, Matthews proposed to introduce chaos theory into the image encryption system in [1] and found that the image encryption scheme based on chaos had high security. Thereafter, image encryption based on chaos has been widely used and developed in many image encryption algorithms [2–15]. However, with the development of computer technology and password cracking technology, more and more shortcomings of chaos-based image encryption schemes are found [16–21]. In [16], Li and Zheng point out that the encryption algorithm proposed in [2] cannot resist chosen/known-plaintext attack. In [17], it is pointed out that the encryption scheme proposed in [3] can be broken by chosen-plaintext attack and differential known-plaintext attack, respectively. Zhu et al. [18] cryptanalyzed the image encryption scheme proposed in [4] using chosen-plaintext combined with chosen-ciphertext attack. Liu and Liu [19] found that some image encryption schemes based on hybrid chaotic system and cyclic elliptic curve are not only insecure against known-plaintext attack but also insecure against chosen-plaintext attack. In [21], Chen et al. analyzed a medical image encryption algorithm that integrates high-speed scrambling and adaptive pixel diffusion [21], and they completely recovered the original image from the corresponding encrypted image under bad randomness using the chosen plain image attack method.

The cryptanalysis of image encryption algorithm motivates the cipher designers to design more secure schemes. We know that the common encryption algorithm is to turn an image into a white noise image through the encryption algorithm, so it is vulnerable to attack by existing attack methods. From the above, we find that many image encryption algorithms are broken because the encryption scheme cannot resist the chosen-plaintext attack. So, in this paper, we have proposed a novel method for 3-image encryption based on 3D nonequilateral Arnold transformation and hyperchaotic system, which can effectively confuse the attacker and resist the chosen-plaintext attack. The proposed scheme includes three steps: the pixel value of the image being converted to a binary value, permutation using 3D nonequilateral Arnold transformation, and diffusion using the chaotic sequence generated by hyperchaotic system. In the permutation stage, the positions of elements in the superimposed 3D bit matrix are rearranged, so the change of each original image pixel will affect the entire rearranged 3D bit matrix. In the stage of diffusion, according to equation (7), the change of one-pixel value will affect the change of other pixel values, resulting in the coupling of pixels at different positions. This encryption scheme can encrypt three grayscale images into one color image; this can confuse the attacker to some extent. Instead of simply using a 1-image method to encrypt three images, respectively, and then compose the result into an RGB-encrypted image, in our approach, the change in the pixel value of each original image affects the result of the entire encryption. The proposed encryption scheme is applied to three publicly accessible test images, and the results indicate that our algorithm not only can effectively resist statistical attacks but also is effective against chosen-plaintext attack.

The rest of this paper is structured as follows. Section 2 introduces the hyperchaotic system and 3D nonequilateral Arnold transformation. The 3-image bit-level scheme based on 3D nonequilateral Arnold transformation and hyperchaotic system is described in detail in Section 3. Section 4 discusses and analyzes the numerical simulation of the proposed scheme. Finally, the conclusion is drawn in Section 5.

2. Preliminary Work

2.1. Hyperchaotic System. In 1999, Chen et al. found a chaotic system similar to Lorenz chaotic system but topologically not equivalent and named this chaotic system as Chen's chaotic system [22]. In 2006, Gao et al. promoted four-dimensional (4D) Chen's chaotic system; thus, a 4D hyperchaotic system [23] is obtained, which is shown as

$$\begin{cases} \dot{x} = 36(y - x), \\ \dot{y} = -16x - xz + 28y - h, \\ \dot{z} = xy - 3z, \\ \dot{h} = x + k. \end{cases} \tag{1}$$

where k is the control parameter of the hyperchaotic system. When $-0.7 \leq k \leq 0.7$, the corresponding Lyapunov exponents of system (1) are $\lambda_1, \lambda_2 > 0, \lambda_3 = 0, \lambda_4 < 0$, and system (1) is hyperchaotic [24]. According to [25], when $k = 0.2$, the Lyapunov exponents of the hyperchaotic system are $\lambda_1 = 1.552, \lambda_2 = 0.023, \lambda_3 = 0, \lambda_4 = -12.573$, and the hyperchaotic attractors are shown in Figure 1. Set the initial value $x_0 = 1, y_0 = 0.1, z_0 = 1.4, h_0 = 4$ and, through iterative system (1), the temporal evolutions of system (1) are obtained, as is shown in Figure 1. For more detailed analysis of the complex dynamics of system (1), please see [23, 25].

2.2. 3D Nonequilateral Arnold Transformation. Based on the researches of Shao et al. [26] and Li et al. [27], Wu and Tian [28] presented the 3D nonequilateral Arnold transformation (2) and corresponding inverse transformation:

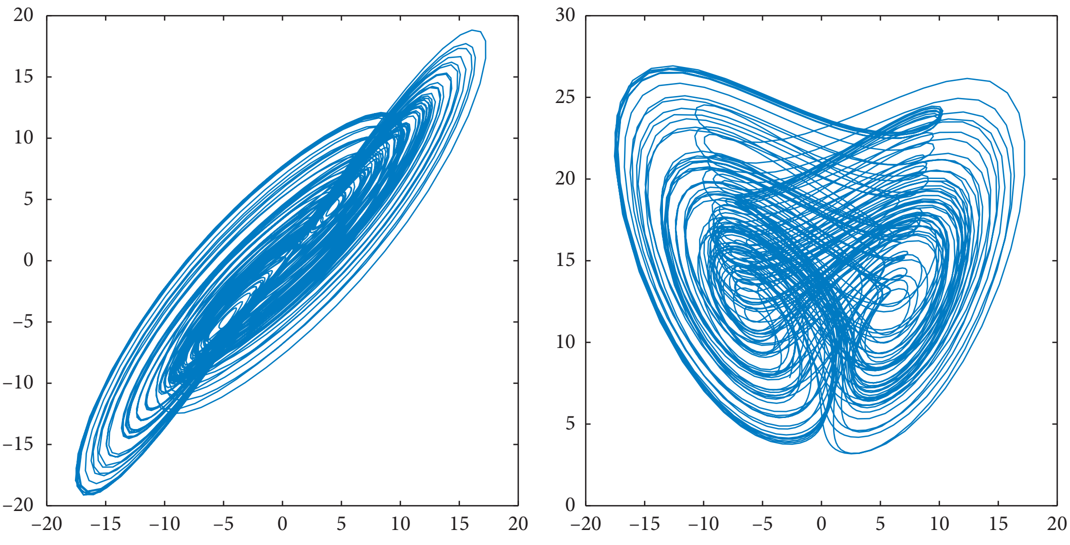
$$\begin{cases} \begin{pmatrix} x_z \\ y_z \\ z_z \end{pmatrix} = \begin{pmatrix} 1 & b_z & 0 \\ c_z & 1 + b_z c_z & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_n \\ y_n \\ z_n \end{pmatrix} \bmod \begin{pmatrix} N \\ M \\ K \end{pmatrix}, \\ \begin{pmatrix} x_x \\ y_x \\ z_x \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & b_x \\ 0 & c_x & 1 + b_x c_x \end{pmatrix} \begin{pmatrix} x_z \\ y_z \\ z_z \end{pmatrix} \bmod \begin{pmatrix} N \\ M \\ K \end{pmatrix}, \\ \begin{pmatrix} x_{n+1} \\ y_{n+1} \\ z_{n+1} \end{pmatrix} = \begin{pmatrix} 1 + b_y c_y & 0 & c_y \\ 0 & 1 & 0 \\ b_y & 0 & 1 \end{pmatrix} \begin{pmatrix} x_x \\ y_x \\ z_x \end{pmatrix} \bmod \begin{pmatrix} N \\ M \\ K \end{pmatrix}. \end{cases} \tag{2}$$

In equation (2), $c_z = r_z (M/\gcd(N, M))$, $c_x = r_x (K/\gcd(M, K)), c_y = r_y (N/\gcd(K, N))$, and $\gcd(\cdot)$ represents the greatest common divisor; $b_x, b_y, b_z, r_x, r_y, r_z$ are any positive integer. Its inverse transformation can be achieved by the three following expressions:

$$\begin{cases} y_x = y_{n+1}, \\ x_x = \left(x_{n+1} - c_y z_{n+1}\right) \bmod N, \\ z_x = \left(z_{n+1} - b_y x_x\right) \bmod K, \end{cases}$$
$$\begin{cases} x_z = x_x, \\ z_z = (z_x - c_x y_x) \bmod K, \\ y_z = (y_x - b_x z_z) \bmod M, \end{cases} \tag{3}$$
$$\begin{cases} z_n = z_z, \\ y_n = (y_z - c_z x_z) \bmod M, \\ x_n = (x_z - b_z y_n) \bmod N. \end{cases}$$

3. 3-Image Bit-Level Encryption Algorithm

The flowchart of the proposed encryption scheme is shown in Figure 2. We consider three original grayscale images of size $N \times M$ represented by three matrixes I_1, I_2, and I_3, respectively. By iterating hyperchaotic system in equation (1), generate four chaotic sequences, $H_1 = \{h_{11}, h_{12}, \ldots, h_{1l}\}$, $H_2 = \{h_{21}, h_{22}, \ldots, h_{2l}\}$, $H_3 = \{h_{31}, h_{32}, \ldots, h_{3l}\}$, and $H_4 = \{h_{41}, h_{42}, \ldots, h_{4l}\}$, and their size of $L = N \times M$. The procedure of the proposed algorithm is described in detail as follows:



(a)

(b)

Figure 1: Hyperchaos attractors of system (1). (a) x-y plane; (b) x-z plane.

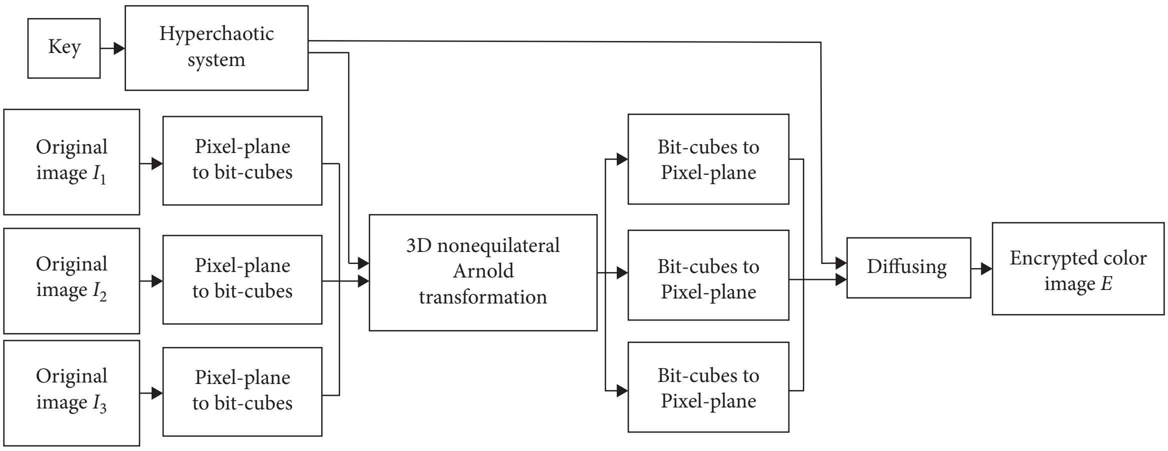


Figure 2: The flowchart of the proposed encryption.

Step 1 Preprocessing chaotic sequences H_1, H_2, H_3, and H_4, we get four integer sequences U_1, U_2, U_3, and U_4, and their elements are between 0 and 255. The mathematical representation of preprocessing is

$$\begin{cases} U_1 = \{u_{11}, u_{12}, \ldots, u_{1L}\}, \\ U_2 = \{u_{21}, u_{22}, \ldots, u_{2L}\}, \\ U_3 = \{u_{31}, u_{32}, \ldots, u_{3L}\}, \\ U_4 = \{u_{41}, u_{42}, \ldots, u_{4L}\}. \end{cases} \tag{4}$$

We have

$$\begin{cases} u_{1i} = \text{floor}\left(10^k h_{1i}\right) \text{mod} 256, \\ u_{2i} = \text{floor}\left(10^k h_{2i}\right) \text{mod} 256, \\ u_{3i} = \text{floor}\left(10^k h_{3i}\right) \text{mod} 256, \\ u_{4i} = \text{floor}\left(10^k h_{4i}\right) \text{mod} 256. \end{cases} \tag{5}$$

Here, $i = 1, 2, \cdots, L$, floor(x) returns the nearest integer less than or equal to x, and mod returns the remainder after division.

Step 2 Transform the three original grayscale images into three bit-level cubes with a size of $N \times M \times 8$, and by superimposing them together, we can get a bit-level cubic J of size $N \times M \times 24$.

Step 3 Take six elements from the sequence U_4 as Arnold parameters b_x, b_y, b_z, r_x, r_y, and r_z; then the bit-level cubic J is scrambled by using the 3D nonequilateral Arnold transformation, and obtain scrambled bit-level cubic A.

Step 4 Divide bit-level cubic A into three bit-level cubes B_1, B_2, and B_3 with a size of $N \times M \times 8$; then perform transformation from these bit-level cubes to three 2D pixel-level images C_1, C_2, and C_3.

Step 5 According to the column scanning, the three 2D pixel-level images are rearranged into three sequences $K_1 = \{k_{11}, k_{12}, \ldots, k_{1L}\}$, $K_2 = \{k_{21}, k_{22}, \ldots, k_{2L}\}$, and $K_3 = \{k_{31}, k_{32}, \ldots, k_{3L}\}$, respectively.

Step 6 The values of the sequences K_1, K_2, and K_3 are diffused with equation (7)fd7; then we can obtain three diffusion sequences D_1, D_2, and D_3.

$$\begin{cases} D_1 = \{d_{11}, d_{12}, \ldots, d_{1L}\}, \\ D_2 = \{d_{21}, d_{22}, \ldots, d_{2L}\}, \\ D_3 = \{d_{31}, d_{32}, \ldots, d_{3L}\}, \end{cases} \tag{6}$$

$$\begin{cases} d_{1i} = \left(\left(d_{1(i-1)} + d_{2(i-1)} + k_{1i} + k_{1(i-1)} + k_{2(i-1)}\right) \bmod 256\right) \oplus k_{3(i-1)} \oplus d_{3(i-1)} \oplus u_{1i}, \\ d_{2i} = \left(\left(d_{2(i-1)} + d_{3(i-1)} + k_{2i} + k_{2(i-1)} + k_{3(i-1)}\right) \bmod 256\right) \oplus k_{1(i-1)} \oplus d_{1(i-1)} \oplus u_{2i}, \\ d_{3i} = \left(\left(d_{3(i-1)} + d_{1(i-1)} + k_{3i} + k_{3(i-1)} + k_{1(i-1)}\right) \bmod 256\right) \oplus k_{2(i-1)} \oplus d_{2(i-1)} \oplus u_{3i}. \end{cases} \tag{7}$$

We have $i = 1, 2, \ldots, L$ and the initial values k_{10}, k_{20}, k_{30}, d_{10}, d_{20}, and d_{30} can be used as encryption and decryption keys.

Step 7 Rearrange the three sequences D_1, D_2, and D_3 into three 2D pixel-level images R, G, and B with a size of $N \times M$, respectively. Then R, G, and B are converted into a color encrypted image E.

Since each step of the above encryption algorithm is reversible, the decryption is the reverse process of the encryption process.

4. Simulation Results and Analysis

The simulation experiments were carried out with MATLAB on a laptop. The initial values and parameters of the hyperchaotic system (1) are $x_0 = 1$, $y_0 = 0.1$, $z_0 = 1.3$, $h_0 = 4$, and $k = 0.2$. The initial values of step 6 are $k_{10} = 34$, $k_{20} = 234$, $k_{30} = 89$, $d_{10} = 60$, $d_{20} = 74$, and $d_{30} = 234$. There are three groups of test images: the first group contains three images ("Lena," "Liftingbody," and "Barbara"), the second group contains three images ("Cameraman," "Rice," and "Text"), and the third group contains three images ("Onion," "Peppers," and "Toyobjects"), which are shown in Figure 3. The results of the three groups of images after 3D non-equilateral transformation are shown in Figure 4. The encryption and decryption results are shown in Figure 5.

4.1. Statistical Analysis. To evaluate the performance of the proposed encryption scheme, some typical statistical analysis methods are adopted in the experiments, such as histogram analysis, correlation analysis, and information entropy analysis.

4.1.1. Histogram Analysis. It is well known that histograms of meaningful image usually show irregular shape, and the histogram reflects the distribution rule of image pixel value. Therefore, a good encryption scheme should change the irregular shape of the original image histogram to make it as evenly distributed as possible, so as to form a completely random-like cryptographic image. The histograms of the original images and their corresponding encrypted color images are shown in Figure 6.

To measure the degree of deviation of pixel distribution from absolute uniformity, we introduce Chi-square test, which can measure the degree of deviation from absolute uniform distribution of image pixel distribution. The Chi-square test can be defined as

$$\chi^2 = \sum_{i=0}^{255} \frac{\left(k_i - \overline{k}\right)^2}{\overline{k}}, \tag{8}$$

where k_i represents the number of occurrences of the pixel value i and $\overline{k}$ is the expected frequency. So, the smaller χ^2 is, the more evenly distributed the image is.

The χ^2 values of the original images and their corresponding encrypted color images are shown in Table 1. From Table 1, we can find that the χ^2 values of the original grayscale images with size of 256×256 are very large. In contrast, the χ^2 values of each component of the encrypted color image are much smaller. This indicates that the proposed encryption scheme can resist histogram attacks.

4.1.2. Correlation Analysis. Meaningful images usually show high correlation; that is, neighboring pixels in each direction (horizontal, vertical, and diagonal directions) have very close pixel values. A good image encryption algorithm can effectively break the correlation between adjacent pixels of the original image. In order to test the correlations of adjacent pixels, the correlation coefficient of adjacent pixels (CCAP) C_{xy} [7] is introduced, which can measure the degree of correlation in a specific direction:

$$C_{xy} = \frac{\sum_{i=1}^{N} \left(x_i - \overline{x}\right)\left(y_i - \overline{y}\right)}{\sqrt{\left(\sum_{i=1}^{N} \left(x_i - \overline{x}\right)^2\right)\left(\sum_{i=1}^{N} \left(y_i - \overline{y}\right)^2\right)}}, \tag{9}$$

where $\overline{x} = (1/N) \sum_{i=1}^{N} x_i$ and $\overline{y} = (1/N) \sum_{i=1}^{N} y_i$. The CCAP of original and encrypted images are listed in Table 2. From Table 2, we can easily see that the correlation coefficient of

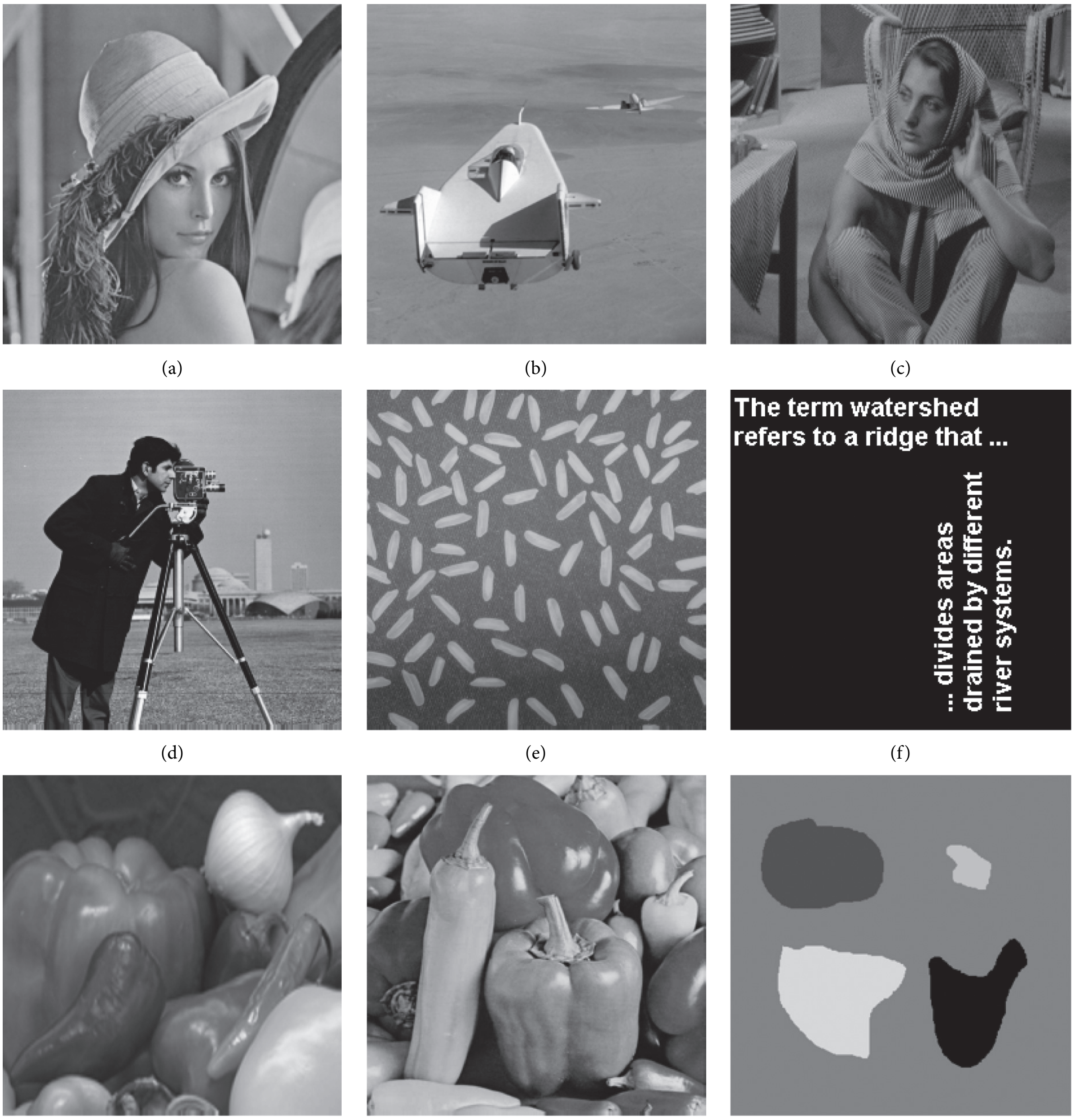

(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 3: Test images: (a) "Lena," (b) "Liftingbody," (c) "Barbara," (d) "Cameraman," (e) "Rice," (f) "Text," (g) "Onion," (h) "Peppers," and (i) "Toyobjects."

the three original images is close to 1 in each direction, while the correlation coefficient of the components of color encrypted image is close to 0 in each direction. In addition, according to the data in Table 2, compared with the method in [7], our algorithm can more effectively eliminate the correlation between adjacent pixels of the original image.

Besides, Figure 7 shows the correlation distribution among vertical adjacent pixels in original images and their encrypted images. It is clear that three strongly correlated original images can be developed into the corresponding color encrypted images with an almost random relationship among adjacent pixels. This further proves that the proposed 3-image encryption algorithm could resist the correlation analysis attack.

4.1.3. Information Entropy. Information entropy $H(m)$ is used to describe the uncertainty of information source. For images, $H(m)$ is proportional to the uniformity of gray value distribution. As for a message source m, the mathematical representation of the corresponding information entropy $H(m)$ can been expressed as follows:

$$H(m) = -\sum_{i=0}^{L-1} p(m_i)\log p(m_i), \tag{10}$$

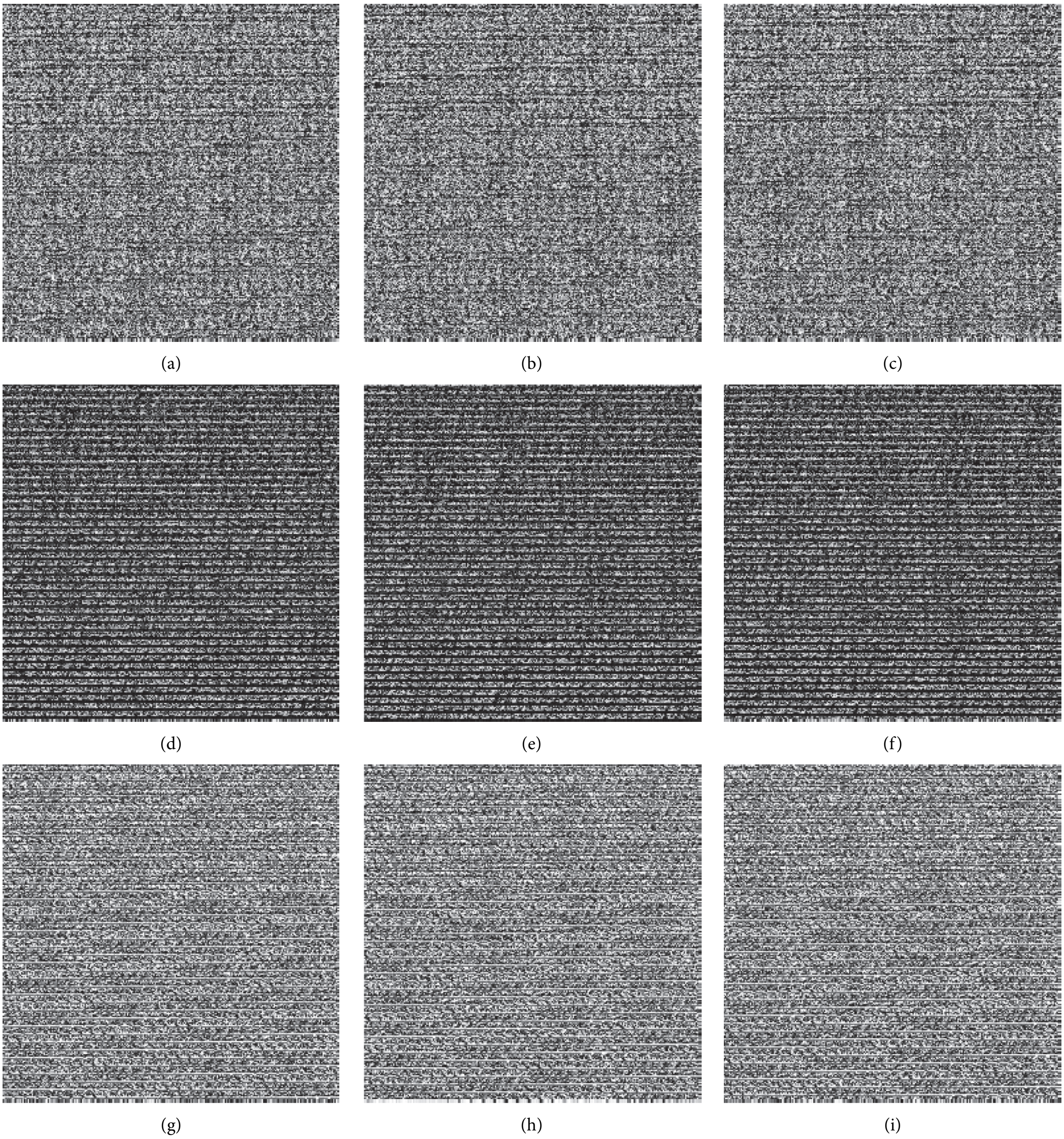


Figure 4: The results of the three groups of images after 3D nonequilateral transformation: (a) scrambling of "Lena," (b) scrambling of "Liftingbody," (c) scrambling of "Barbara," (d) scrambling of "Cameraman," (e) scrambling of "Rice," (f) scrambling of "Text," (g) scrambling of "Onion," (h) scrambling of "Peppers," and (i) scrambling of "Toyobjects."

where L and $p(m_i)$ represent the total number and the probability of symbol m_i, and log represents the base 2 logarithm so that the entropy is expressed in bits. For 8-bit grayscale image, the probability of each value within $[0, 255]$ is $(1/256)$, so the ideal entropy value of a well-encrypted image is 8 bits.

The information entropies of our scheme, [15], and [29] are listed in Table 3. It is indicated that the information entropies of the components of color encrypted images are close to the ideal value of 8; this means that the proposed 3-image encryption algorithm can counteract the entropy attack. As seen from Table 3, the information entropy obtained by our algorithm is larger than those of [15] and [29]; it means that our algorithm is more effective than the proposed algorithms in [15] and [29].

4.2. Security Key Analysis. A secure encryption scheme should be extremely sensitive to the key and have enough large key space. To evaluate the sensitivity of the proposed

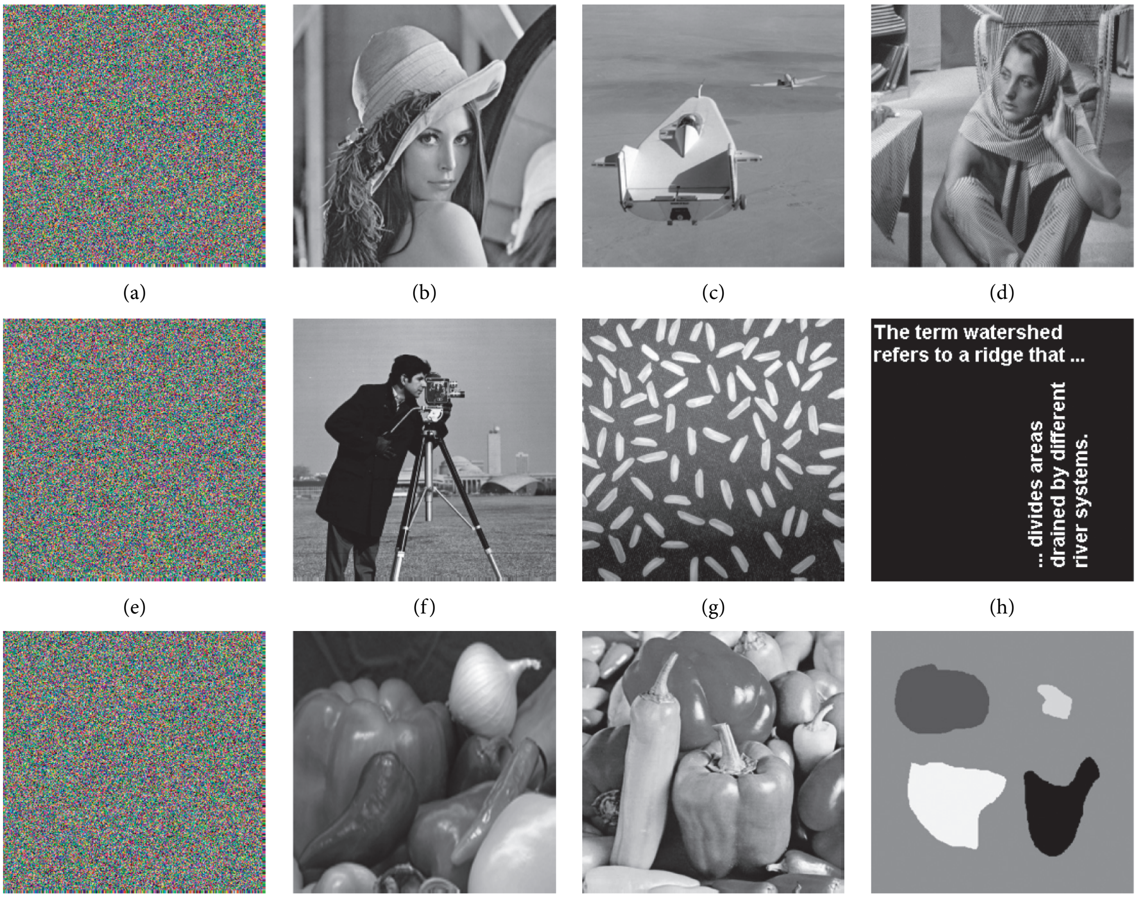

(a)
(b)
(c)
(d)
(e)
(f)
(g)
The term watershed refers to a ridge that ...
... divides areas drained by different river systems.
(h)
(i)
(j)
(k)
(l)

Figure 5: The encryption and decryption results: (a) encrypted color image for the first group of test images, (b) decryption of “Lena,” (c) decryption of “Liftingbody,” and (d) decryption of “Barbara”; (e) encrypted color image for the second group of test images, (f) decryption of “Cameraman,” (g) decryption of “Rice,” and (h) decryption of “Text”; and (i) encrypted color image for the third group of test images, (j) decryption of “Onion,” (k) decryption of “Peppers,” and (l) decryption of “Toyobjects.”

encryption scheme to the key, one of the correct keys can be changed slightly, while the others remain unchanged. To analyze the sensitivity of the proposed algorithm to the key, two sets of keys are used to decrypt the encryption image. Figures 8(a)–8(c) show the decrypted images with an incorrect key $x0$, while the other keys are unchanged. Figures 8(d)–8(f) illustrate the decrypted images with a wrong key k, while the other keys are all correct. It is obvious that even the tiny change of 10^{-15} with the correct keys results is completely unrecognizable. So, the key spaces for x_0, y_0, z_0, h_0, and k are $S_{x_0} = S_{y_0} = S_{z_0} = S_{h_0} = S_k \approx 10^{15}$.

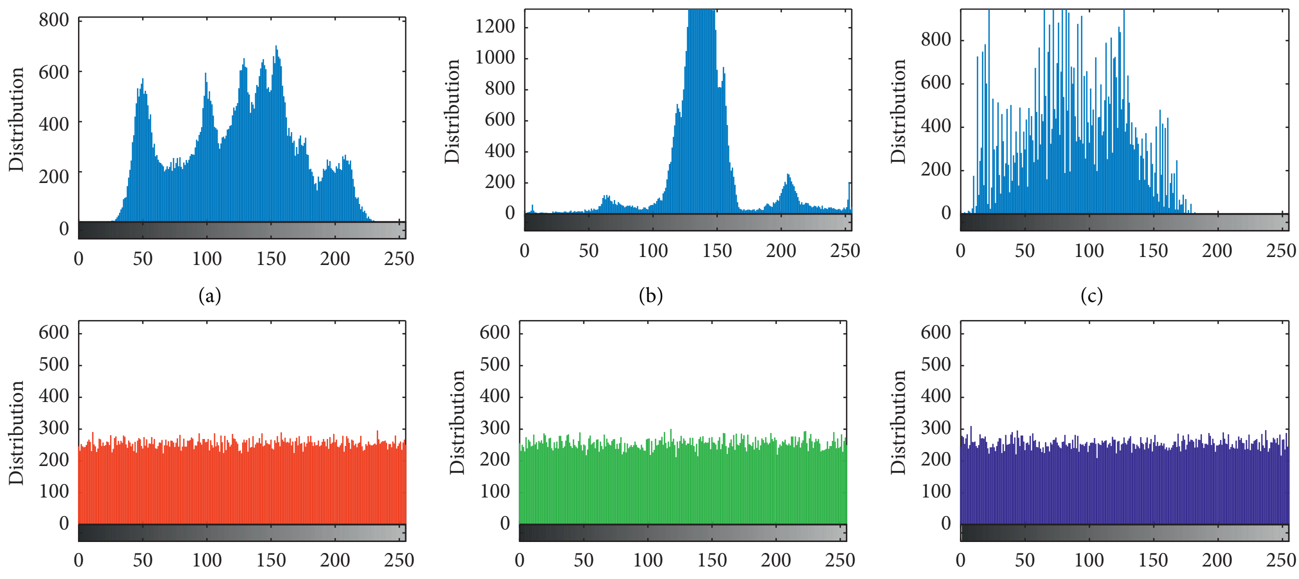
The initial values k_{10}, k_{20}, k_{30}, d_{10}, d_{20}, and d_{30} are 8-bit integers, so $S_{k_{10}} = S_{k_{20}} = S_{k_{30}} = S_{d_{10}} = S_{d_{20}} = S_{d_{30}} = 2^8$. Therefore, the total key space $S = S_{x_0}S_{y_0}S_{z_0}S_{h_0}S_kS_{k_{10}}S_{k_{20}}S_{k_{30}}S_{d_{10}}S_{d_{20}}S_{d_{30}} \approx 2.815 \times 10^{89}$, which is so large that it can resist the brute-force attack. The comparison with other algorithms in key space is shown in Table 4, which shows that our proposed algorithm has larger key space compared to other algorithms.

As for image, the degree of difference between two images can be measured by the mean square error (MSE):

$$\mathrm{MSE} = \frac{1}{N \times M} \sum_{i=1}^{N} \sum_{i=1}^{M} \left(I(i, j) - H(i, j) \right)^2, \tag{11}$$

where $I(i, j)$ and $H(i, j)$ stand for the gray values at point (i, j) of original color image and decrypted image, respectively. The MSE curves of test images for x_0 are computed and shown in Figure 9. Obviously, the MSE is very large with a little deviation to the correct keys and it is close to 0 only when the main keys are correct. It validates that the 3-image encryption algorithm is extremely sensitive to the security keys.

4.3. Differential Analysis. According to the cryptanalyst’s mastery of plaintext, ciphertext, and other data resources, the cryptanalysis attacks against encryption systems can be divided into the following four types: ciphertext-only attack, plaintext-known attack, chosen-plaintext attack, and chosen-ciphertext attack. Among them, the chosen-plaintext attack is the most powerful attack on cryptosystem. So, a good encryption system should be very sensitive to the plain images. In other words, if a little change in a plain image can



(a) (b) (c)

(d) (e) (f)

Figure 6: Histograms of the original images and their corresponding encrypted color images: (a) original image "Lena," (b) original image "Liftingbody," (c) original image "Barbara," (d) R component of Figure 5(a), (e) G component of Figure 5(a), and (f) B component of Figure 5(a).

Table 1: χ^2 values for plain and encrypted images.

	Gray images (256×256)			Figure 5(a) (256×256)		
	Lena	Liftingbody	Barbara	R	G	B
χ^2	41145	213250	77465	200.8125	281.9141	272.5625

Table 2: CCAP in original and encrypted images.

	Original images			Our algorithm			Ref. [7]		
	Lena	Liftingbody	Barbara	R	G	B	R	G	B
Horizontal	0.9718	0.9634	0.9278	−−0.0021	0.0014	−0.0016	0.0027	0.0034	0.0046
Vertical	0.9442	0.9479	0.8869	−0.0011	−0.0002	0.0005	−0.0013	−0.0034	0.0038
Diagonal	0.9181	0.9194	0.8697	−0.0017	0.0015	0.0004	−0.0039	−0.0021	0.0013

lead to a completely different cipher image, then this cryptosystem can effectively resist chosen-plaintext attack and differential attacks [29, 31].

The number of pixels changing rate (NPCR) and the unified averaged changed intensity (UACI) are two widely used performance indexes for differential attack analysis. For two encrypted images $C_{\mathrm{R,G,B}}$, $C'_{\mathrm{R,G,B}}$ the corresponding before and after one pixel of the plain image is changed, respectively. Mathematically, NPCR and UACI can be defined in the following equations [32]:

$$\mathrm{NPCR} = \frac{\sum_{i,j} D_{\mathrm{R,G,B}}(i,j)}{N \times M} \times 100\%,$$
$$\mathrm{UACI} = \frac{\sum_{i,j} \left| C_{\mathrm{R,G,B}}(i,j) - C'_{\mathrm{R,G,B}}(i,j) \right|}{N \times M \times H} \times 100\%, \tag{12}$$

respectively; here, H represents the largest allowed pixel value in the image, and

$$D_{\mathrm{R,G,B}}(i,j) = \begin{cases} 0, & \text{if } C_{\mathrm{R,G,B}}(i,j) = C_{\mathrm{R,G,B}}{}'(i,j), \\ 1, & \text{if } C_{\mathrm{R,G,B}}(i,j) \neq C_{\mathrm{R,G,B}}{}'(i,j). \end{cases} \tag{13}$$

For two random images, the theoretical values of NPCR and UACI for an 8-bit gray image are NPCR = 99.6094% and UACI = 33.4635% [33].

The NPCR and UACI values of plain images and changing a pixel value in one of the plain images are shown in Table 5. It can be seen from Table 5 that only one-pixel value of one of the three images has changed the NPCR values no less than 0.9696 and UACI values no less than 0.3336. That is, any tiny changes in one of the plain images can result in a totally different cipher image, so the proposed scheme can resist differential attacks and chosen-plaintext attack.

4.4. Performance in Lossy and Noisy Communication Channels.

In this section, in order to analyze the performance of the proposed encryption algorithm in noisy and

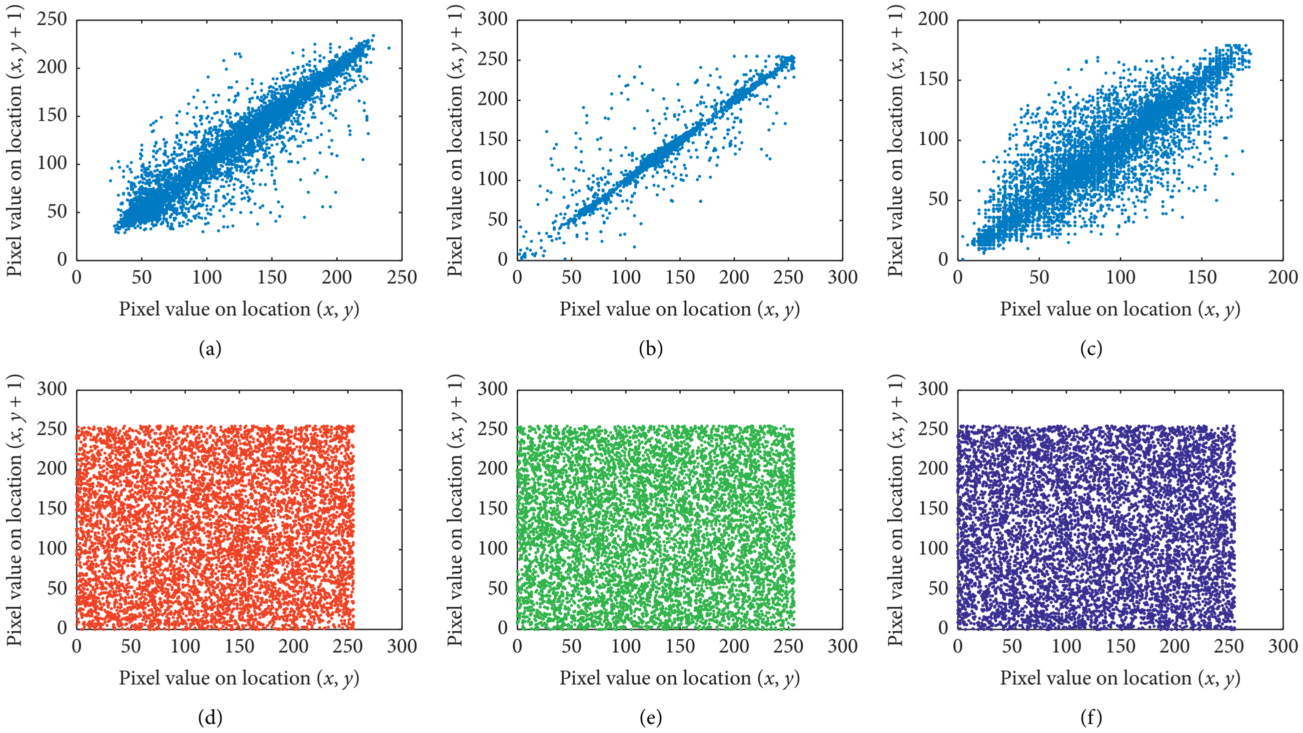


Figure 7: Correlation distribution of two vertical adjacent pixels in (a) original image “Lena,” (b) original image “Liftingbody,” (c) original image “Barbara,” (d) R component of Figure 5(a), (e) G component of Figure 5(a), and (f) B component of Figure 5(a).

Table 3: Comparison of information entropies for encrypted images by different encryption algorithms.

	Our algorithm			Ref. [15]			Ref. [29]		
	R	G	B	R	G	B	R	G	B
Information entropy	7.9978	7.9969	7.9970	7.9898	7.9898	7.9908	7.9896	7.9893	7.9896

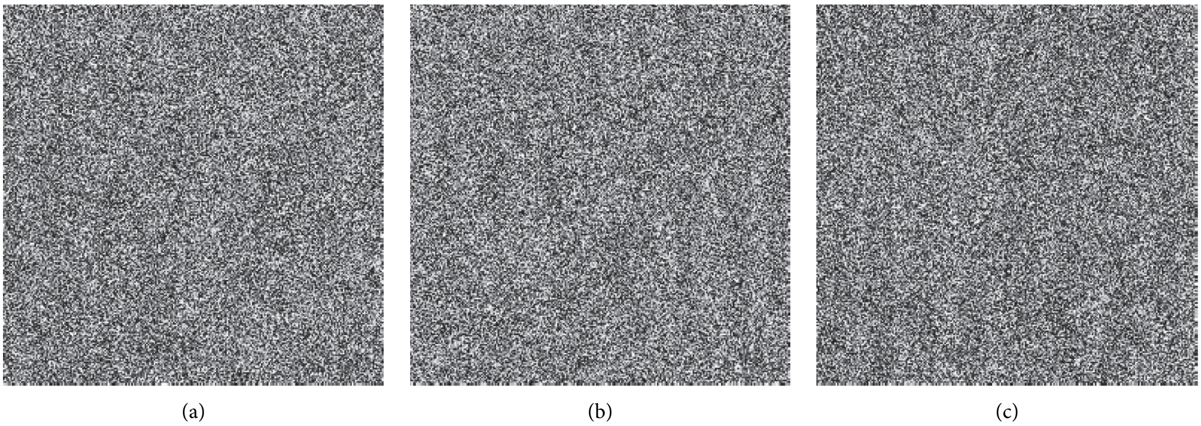


Figure 8: Continued.

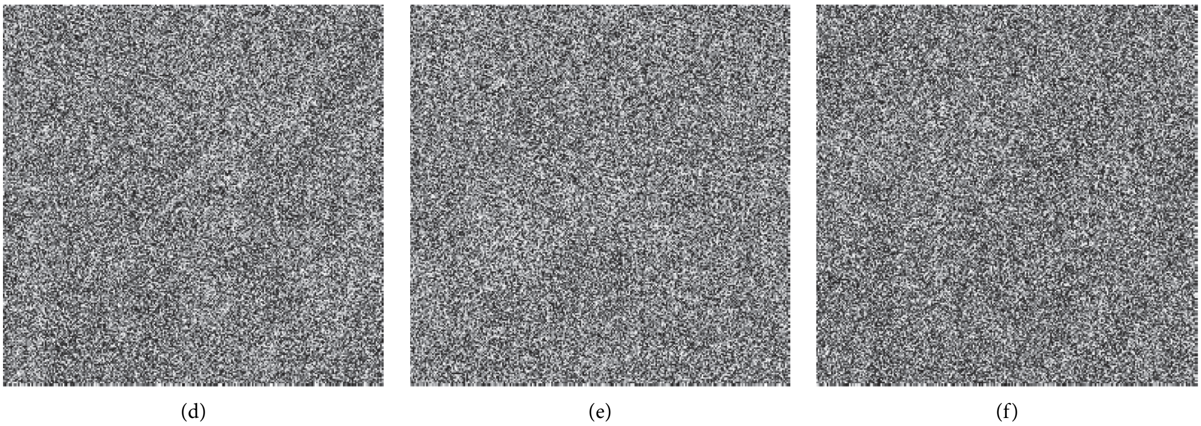
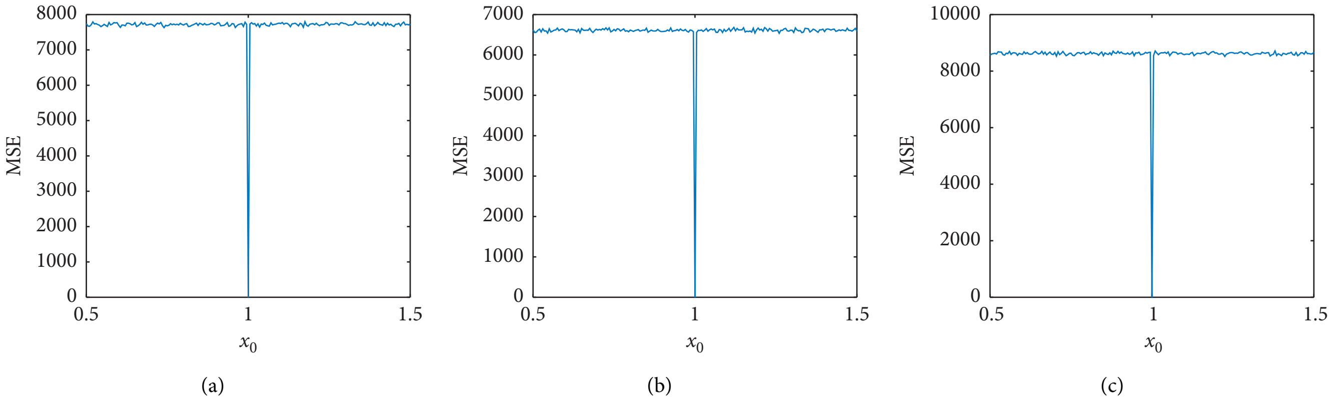

(d) (e) (f)

Figure 8: Decryption images with wrong key $x_0 = 1 + 10^{-15}$: (a) decrypted image "Lena," (b) decrypted image "Liftingbody," and (c) decrypted image "Barbara"; and decryption images with incorrect key $k = 0.2 + 10^{-15}$: (d) decrypted image "Lena," (e) decrypted image "Liftingbody," and (f) decrypted image "Barbara."

Table 4: Comparison of key spaces.

Algorithm	Our algorithm	Anand et al. [30]	Zhou et al. [10]
Key space	2.815×10^{89}	10^{75}	2^{187}



(a) (b) (c)

Figure 9: MSE curves for x_0: (a) "Lena," (b) "Liftingbody," and (c) "Barbara."

Table 5: The NPCR and UACI values for encrypting three test images.

	Lena			Liftingbody			Barbara		
	R	G	B	R	G	B	R	G	B
NPCR	0.9696	0.9696	0.9696	0.9934	0.9934	0.9934	0.9954	0.9954	0.9954
UACI	0.3346	0.3336	0.3349	0.3359	0.3353	0.3357	0.3357	0.3352	0.3374

lossy communication channels, two simulation experiments are performed.

The simulation results of the proposed algorithm are shown in Figures 10 and 11. We first occlude 1/256, 1/64, and 1/16 of Figure 5(a), and the corresponding recovered images with correct keys are shown in Figure 10. It can be seen from the figure that only the basic outline of the original image can be restored when a small part of the pixels in Figure 5(a) are obscured. However, when the occlusion ratio increases, less information is obtained from the original image from the decrypted output image. When adding Salt & Pepper noise (density 0.00001) and Gaussian noise (mean 0 and variance 0.00001) to the encrypted image, the corresponding decrypted images are shown in Figure 11. It is indicated that the proposed encryption algorithm is weakly robust against Salt & Pepper noise attacks. However, it is very poorly robust to Gaussian noise.

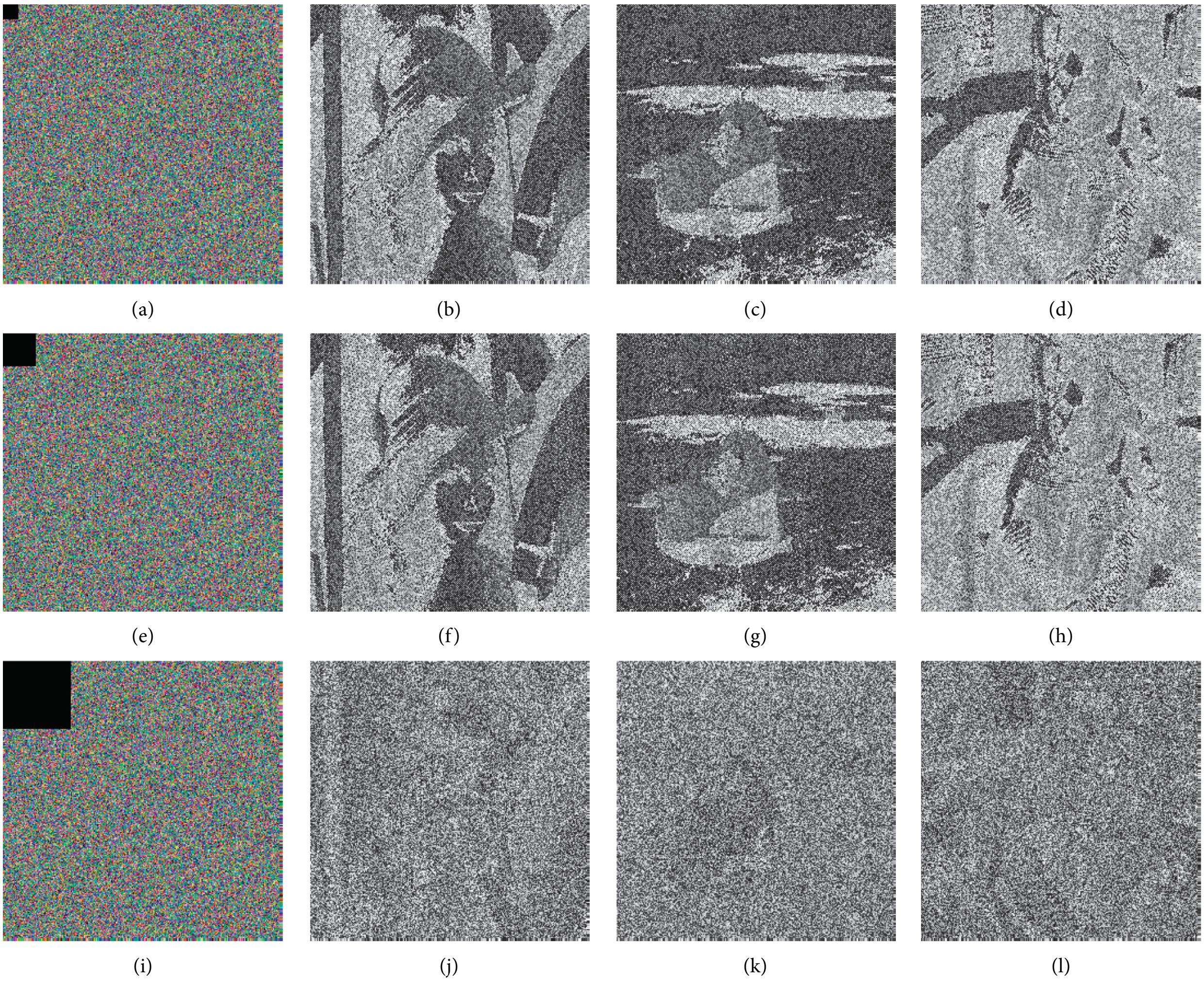


Figure 10: Cut-encrypted images and the corresponding decrypted output: (a) cut 1/256, (b) decrypted "Lena" from (a), (c) decrypted "Liftingbody" from (a), (d) decrypted "Barbara" from (a), (e) cut 1/64, (f) decrypted "Lena" from (e), (g) decrypted "Liftingbody" from (e), (h) decrypted "Barbara" from (e), (i) cut 1/16, (j) decrypted "Lena" from (i), (k) decrypted "Liftingbody" from (i), and (l) decrypted "Barbara" from (i).

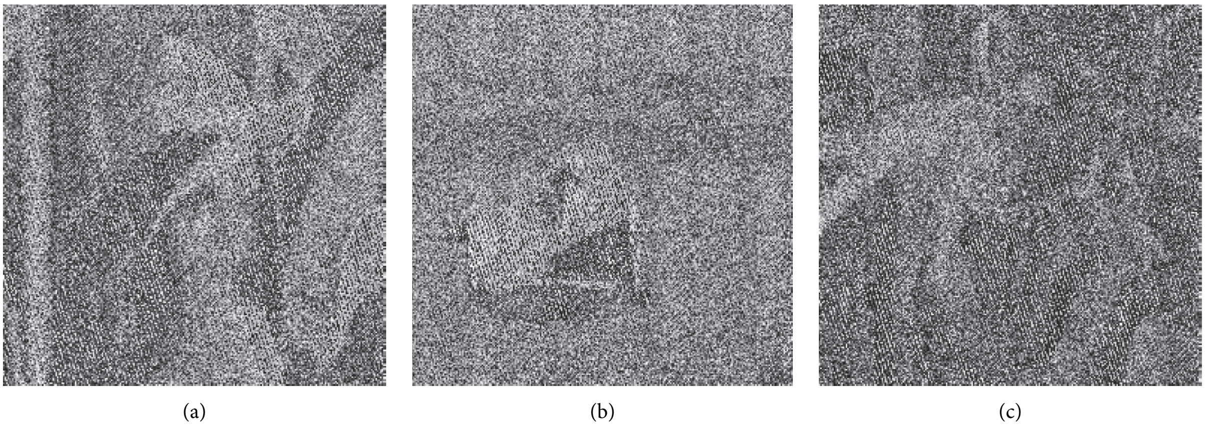


Figure 11: Continued.

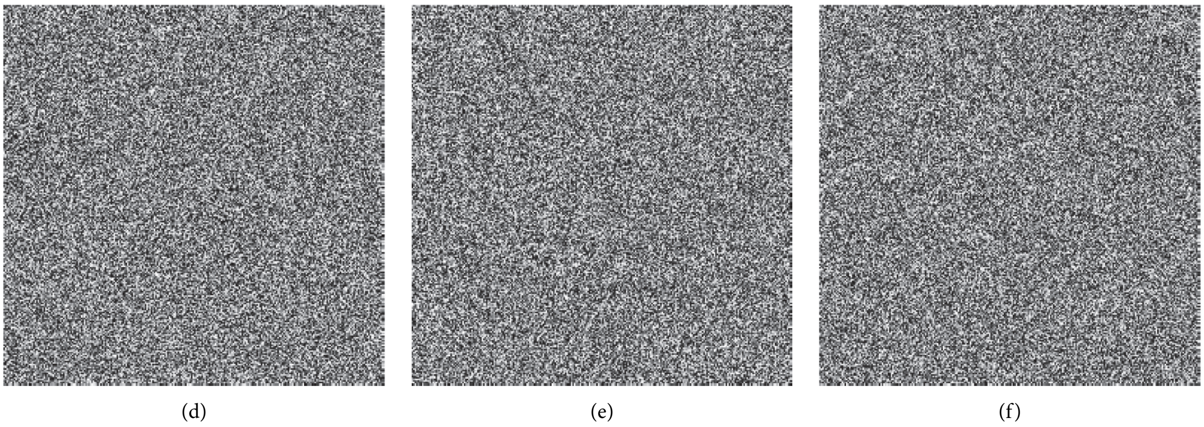


Figure 11: Decrypted output for noise attacked images: (a) decrypted "Lena" from encrypted image to which Salt & Pepper noise is added, (b) decrypted "Liftingbody" from encrypted image to which Salt & Pepper noise is added, (c) decrypted "Barbara" from encrypted image to which Salt & Pepper noise is added, (d) decrypted "Lena" from encrypted image to which Gaussian noise is added, (e) decrypted "Liftingbody" from encrypted image to which Gaussian noise is added, and (f) decrypted "Barbara" from encrypted image to which Gaussian noise is added.

4.5. Computational Complexity. In the following, we analyzed the computational complexity of the proposed algorithm. In fact, the computational complexity of the encryption algorithm is determined by the operations and steps required to complete the encryption/decryption process, if some details such as the operating system, programming language, hardware on which the algorithm runs, and programming skills are ignored [34–36]. For three grayscale images with size of $M \times N$, there are $3 \times M \times N$ pixels. In the process of 3D nonequilateral Arnold transformation, the transformation object is bit element, so the time complexity is $O(24 \times M \times N)$. The diffusion operation is of the pixel level, so the time complexity is $O(3 \times M \times N)$. In addition, during the generation of chaotic sequences, the hyperchaotic system is iterated for many times, and the time complexity of floating-point operation is $O(4 \times M \times N)$. Therefore, the total time complexity of the proposed algorithm is $O(24 \times M \times N)$, which is efficient for real-time applications.

5. Conclusion

In this work, we presented a new scheme for 3-image encryption and decryption. In this scheme, the pixel values of the three grayscale images are converted to binary values first. Next, we put them on top of each other to get a 3D bit matrix and then it is scrambled using the 3D nonequilateral Arnold transformation. Then the scrambled 3D bit matrix is divided into three equal parts and converted into three 2D pixel-level images. Finally, the chaotic sequences generated by the hyperchaotic system are used to diffuse the three 2D pixel-level images and the three diffused images are considered as RGB components of color encrypted image. We know that the method of simply using a 1-image method to encrypt three images, respectively, and then composing the result into an RGB-encrypted image has a disadvantage; that is, the change of pixel value of each image will only affect one component of the RGB-encrypted image. However, the algorithm proposed in this paper overcomes this shortcoming. According to the simulation results of differential analysis, only changing one-pixel value of any image will cause a change of at least 96.96% ciphertext. The extensive experiments demonstrate that our proposed 3-image encryption scheme has the ability to resist several types of attacks such as statistical analysis, brute-force attack, differential attack, and occlusion attack. These results show that the proposed encryption algorithm is promising for secure transmission of three images by single algorithm. In the future, in order to adapt to more and more serious network security problems, the design of image encryption technology with high security and strong robustness is worth further study.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to acknowledge the support from the Natural Science Foundation of Guangdong Province (Grant nos. 2018A030307024, 2018A0303100016, and 2020A1515010561), the Key Research Platform and Research Project of Universities in Guangdong Province (Grant no. 2018KQNCX244), and the Characteristic Innovation Project from the Educational Department of Guangdong Province (Grant nos. 2018GKTSCX043 and 2019KTSCX168).

References

[1] R. Matthews, “On the derivation of a “chaotic” encryption algorithm,” *Cryptologia*, vol. 13, no. 1, pp. 29–42, 1989.

[2] J. Yen and J. Guo, “A new chaotic key-based design for image encryption and decryption,” *IEEE International Symposium On Circuits & Systems*, vol. 4, pp. 49–52, 2000.

[3] T. Xiang, X. Liao, G. Tang, Y. Chen, and K. Wong, “A novel block cryptosystem based on iterating a chaotic map,” *Physics Letters A*, vol. 349, no. 1–4, pp. 109–115, 2006.

[4] G. Zhang and Q. Liu, “A novel image encryption method based on total shuffling scheme,” *Optics Communications*, vol. 284, no. 12, pp. 2775–2780, 2011.

[5] X.-Y. Wang, Y.-Q. Zhang, and X.-M. Bao, “A novel chaotic image encryption scheme using DNA sequence operations,” *Optics and Lasers in Engineering*, vol. 73, pp. 53–61, 2015.

[6] H. Chen, X. Du, and Z. Liu, “Optical hyperspectral data encryption in spectrum domain by using 3d Arnold and gyrator transforms,” *Spectroscopy Letters*, vol. 49, no. 2, pp. 103–107, 2016.

[7] H. Huang and S. Yang, “Colour image encryption based on logistic mapping and double random-phase encoding,” *IET Image Processing*, vol. 11, no. 4, pp. 211–216, 2017.

[8] X. Chai, Z. Gan, Y. Lu, Y. Chen, and D. Han, “A novel image encryption algorithm based on the chaotic system and DNA computing,” *International Journal of Modern Physics C*, vol. 28, no. 5, p. 1750069, 2017.

[9] T. Li, M. Yang, J. Wu, and X. Jing, “A novel image encryption algorithm based on a fractional-order hyperchaotic system and DNA computing,” *Complexity*, vol. 2017, Article ID 9010251, 13 pages, 2017.

[10] N. Zhou, X. Yan, H. Liang, X. Tao, and G. Li, “Multi-image encryption scheme based on quantum 3D Arnold transform and scaled Zhongtang chaotic system,” *Quantum Information Processing*, vol. 17, no. 12, p. 338, 2018.

[11] H. Huang, S. Yang, and R. Ye, “Image encryption scheme combining a modified Gerchberg-Saxton algorithm with hyper-chaotic system,” *Soft Computing*, vol. 23, no. 16, pp. 7045–7053, 2019.

[12] R. Lan, J. He, S. Wang, T. Gu, and X. Luo, “Integrated chaotic systems for image encryption,” *Signal Processing*, vol. 147, no. JUN, pp. 133–145, 2018.

[13] T. Li, J. Shi, X. Li, J. Wu, and F. Pan, “Image encryption based on pixel-level diffusion with dynamic filtering and DNA-level permutation with 3D Latin cubes,” *Entropy*, vol. 21, no. 3, p. 319, 2019.

[14] X. Li, T. Li, J. Wu, Z. Xie, and J. Shi, “Joint image compression and encryption based on sparse Bayesian learning and bit-level 3D Arnold cat maps,” *PLoS One*, vol. 14, no. 11, Article ID e0224382, 2019.

[15] H. Huang, “Novel scheme for image encryption combining 2d logistic-sine-cosine map and double random-phase encoding,” *IEEE Access*, vol. 7, no. 1, pp. 177988–177996, 2019.

[16] S. Li and X. Zheng, “Cryptanalysis of a chaotic image encryption method,” in *2002 IEEE International Symposium on Circuits and Systems. Proceedings (Cat. No. 02CH37353)*, Phoenix-Scottsdale, AZ, USA, May 2002.

[17] C. Li, S. Li, G. Alvarez, G. Chen, and K. Lo, “Cryptanalysis of two chaotic encryption schemes based on circular bit shift and XOR operations,” *Physics Letters A*, vol. 369, no. 1-2, pp. 23–30, 2007.

[18] C. Zhu, C. Liao, and X. Deng, “Breaking and improving an image encryption scheme based on total shuffling scheme,” *Nonlinear Dynamics*, vol. 71, no. 1-2, pp. 25–34, 2013.

[19] H. Liu and Y. Liu, “Cryptanalyzing an image encryption scheme based on hybrid chaotic system and cyclic elliptic curve,” *Optics & Laser Technology*, vol. 56, no. 1, pp. 15–19, 2014.

[20] P. Mario, H. Thomas, K. Stefan, and U. Andreas, “Depreciating motivation and empirical security analysis of chaos-based image and video encryption,” *IEEE Transactions On Information Forensics & Security*, vol. 13, no. 9, pp. 2137–2150, 2018.

[21] Y. Chen, C. Tang, and R. Ye, “Cryptanalysis and improvement of medical image encryption using high-speed scrambling and pixel adaptive diffusion,” *Signal Processing*, vol. 167, p. 107286, 2020.

[22] G. Chen and T. Ueta, “Yet another chaotic attractor,” *International Journal of Bifurcation and Chaos*, vol. 9, no. 7, pp. 1465-1466, 1999.

[23] T. Gao, Z. Chen, Z. Yuan, and G. Chen, “A hyperchaos generated from chen’s system,” *International Journal of Modern Physics C*, vol. 17, no. 4, pp. 471–478, 2006.

[24] T. Gao and Z. Chen, “A new image encryption algorithm based on hyper-chaos,” *Physics Letters A*, vol. 372, no. 4, pp. 394–400, 2008.

[25] S. Yanchuk and T. Kapitaniak, “Symmetry-increasing bifurcation as a predictor of a chaos-hyperchaos transition in coupled systems,” *Physical Review E*, vol. 64, no. 5, p. 056235, 2001.

[26] L. Shao, Z. Qin, H. Gao, and X. Heng, “2-Dimension non equilateral image scrambling transformation,” *Acta Electronica Sinica*, vol. 35, no. 7, pp. 1290–1294, 2007.

[27] Y. Li, Q. Feng, F. Zhou, and Q. Li, “2-D Arnold transformation and non-equilateral image scrambling transformation,” *Computer Engineering and Design*, vol. 30, no. 13, pp. 3133–3135, 2009.

[28] C. Wu and X. Tian, “3-Dimensional non-equilateral Arnold transformation and its application in image scrambling,” *Journal Of Computer-Aided Design & Computer Graphics*, vol. 22, no. 10, pp. 1831–1840, 2010.

[29] M. L. Sahari and I. Boukemara, “A pseudo-random numbers generator based on a novel 3D chaotic map with an application to color image encryption,” *Nonlinear Dynamics*, vol. 94, no. 1, pp. 723–744, 2018.

[30] B. Anand, K. Dhanesh, G. Abdul, and D. Mishra, “Triple color image encryption based on 2d multiple parameter fractional discrete Fourier transform and 3D Arnold transform,” *Optics and Lasers in Engineering*, vol. 133, p. 106139, 2020.

[31] X. J. Tong, Z. Wang, M. Zhang, Y. Xu, and J. Ma, “An image encryption algorithm based on the perturbed high-dimensional chaotic map,” *Nonlinear Dynamics*, vol. 80, no. 3, pp. 1493–1508, 2015.

[32] S. Seyedzadeh and S. Mirzakuchaki, “A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map,” *Signal Processing*, vol. 92, no. 5, pp. 1202–1215, 2012.

[33] T. Hu, Y. Liu, L.-H. Gong, S.-F. Guo, and H.-M. Yuan, “Chaotic image cryptosystem using DNA deletion and DNA insertion,” *Signal Processing*, vol. 134, pp. 234–243, 2017.

[34] Y.-Q. Zhang and X.-Y. Wang, “A symmetric image encryption algorithm based on mixed linear-nonlinear coupled map lattice,” *Information Sciences*, vol. 273, pp. 329–351, 2014.

[35] Y.-Q. Zhang and X.-Y. Wang, “A new image encryption algorithm based on non-adjacent coupled map lattices,” *Applied Soft Computing*, vol. 26, pp. 10–20, 2015.

[36] A. Belazi, A. A. Abd El-Latif, and S. Belghith, “A novel image encryption scheme based on substitution-permutation network and chaos,” *Signal Processing*, vol. 128, pp. 155–170, 2016.