Hindawi
Complexity
Volume 2020, Article ID 3026972, 23 pages
https://doi.org/10.1155/2020/3026972

Research Article

A Novel Image Encryption Scheme Based on PWLCM and Standard Map

Yucheng Chen,[1] **Chunming Tang**,[1,2] **and Zongxiang Yi**[1]

[1]*School of Mathematics and Information Science, Guangzhou University, Guangzhou 510006, China*
[2]*State Key Laboratory of Cryptology, P.O. Box 5159, Beijing 100878, China*

Correspondence should be addressed to Chunming Tang; ctang@gzhu.edu.cn

Received 13 June 2020; Revised 25 October 2020; Accepted 7 December 2020; Published 24 December 2020

Academic Editor: Rosa M. Lopez Gutierrez

In the past decades, considerable attention has been paid to the chaos-based image encryption schemes owing to their characteristics such as extreme sensitivity to initial conditions and parameters, pseudo-randomness, and unpredictability. However, some schemes have been proven to be insecure due to using a single chaotic system. To increase the security, this work proposes a novel image encryption scheme based on the piecewise linear chaotic map (PWLCM) and the standard map. To the best of our knowledge, it is the first chaos-based image encryption scheme combining the PWLCM with the standard map, which adopts permutation-diffusion structure. Unlike the traditional scrambling way, a hierarchical diffusion strategy, which not only changes the pixel position but also modifies the value, is employed in the permutation phase. The operation model of row-by-row and column-by-column is further used to enhance the efficiency in the diffusion process. Consequently, a good trade-off efficiency and security can be achieved. Furthermore, the numerical simulations and performance analyses illustrate that the proposed encryption scheme can be used in practical application scenarios requiring lightweight security.

1. Introduction

With the rapid development of information technologies, the multimedia security has become more important than ever before. Image, as a special multimedia form, is becoming more popular in our daily life because of its intrinsic properties such as more intuitive and vivid than text. However, it would bring privacy problems when it is transmitted in an insecure channel. Therefore, it is very urgent to protect the privacy of image. The encryption scheme is one of the methods to protect the image, which transforms the essential contents of a plain image into a noise-like encrypted image. In the past decades, considerable attention has been paid to design chaos-based image encryption schemes since the chaotic map has many excellent characteristics such as extreme sensitivity to initial conditions and parameters and unpredictability of behavior [1]. Many chaos-based image encryption schemes have been proposed by the researchers working in the field of nonlinear dynamic and information security [2, 3].

In 1998, Fridrich constructed the first image encryption scheme based the discretized Baker map using permutation-diffusion structure [4]. However, it was found that the encryption scheme can be broken with chosen encrypted image attack [5, 6]. To improve Fridrich's work, numerous chaos-based encryption schemes have been proposed.

In particular, the piecewise linear chaotic map (PWLCM), which is efficient to implement in hardware, was used in the encryption schemes presented in [7–12]. Peng et al. designed an image encryption scheme based on PWLCM [7] in 2008, which adopts a modulo addition operation to encrypt plain image with the pseudo-random sequence generated by PWLCM. Although it may be easy to implement, the security level is low because of its too simple structure. In 2011, Abdlrudha et al. utilized a nested PWLCM to encrypt plain image based on permutation-diffusion architecture [8]. The proposed image encryption scheme can achieve both low complexity and high level security performance. Later in 2012, Liu et al. described an image encryption scheme with DNA complimentary and

chaotic maps [9]. The PWLCM was used to generate the pseudo-random sequences in the row and column scrambling phase. Similarly, Wang et al. also used the sequence generated by PWLCM to encrypt permuted image in diffusion process [10, 11]. In 2020, Patro et al. [12] used a cross-coupled PWLCM system to encrypt plain image, which aims to obtain a higher security than using simple chaotic map.

While the one-dimensional chaotic system has low implementation complexity, the multidimensional chaotic system possess more complex behavior [13]. Specifically, the two-dimensional standard map has larger number of control parameters than the one-dimensional PWLCM, which is the reason for designers choosing it. In 2009, Patidar et al. proposed a permutation-diffusion-based image encryption scheme with the standard chaotic system and the logistic map [14]. In 2011, Patidar et al. further proposed an improved image encryption scheme based on the standard map [15]. The operation model of row-by-row and column-by-column was adopted to preliminary permutation, substitution, and main permutation for increasing the speed of the encryption process. Zhang et al. employed the standard map to scramble the pixels of two plain images [16] in 2013. In 2018, Chen et al. proposed an optical hyperspectral image encryption scheme with the standard map and the gyrator transform [17]. The pixels of plain image were scrambled according to the position sequence generated by the improved standard map. Unlike the most common permutation-diffusion architecture, Chen et al. [18] in 2019 suggested a substitution mechanism in the permutation phase via a chaos-based bit-level shuffling method. Thus, the same security level can be obtained in fewer encryption rounds.

Recently, Ye et al. in 2017 proposed a spatial image encryption scheme based on chaotic map and pixel frequency [19]. The authors creatively designed a weight factor related to the pixel frequency, which aims to produce key streams associated with the plain image and thus resists the chosen plain image attack. However, one can easily test its weak position sensitivity using the given definition formula. Let plain image P be all zeros except for 1 in one position; its pixel frequency value would be the same if we move 1 to another position. Specifically, the pixel frequency is identical to the previous one due to the fact that the pixel value and its frequency are unchanged before and after moving position. This may give a chance to the attacker breaking their proposed encryption scheme under the condition of round reduced. Moreover, Patro et al. [20] and Samiullah et al. [21] in 2020 presented a plain image-related encryption scheme by using hash function. As a result, for two different plain images, even a bit of difference would have two different hash values. This means that the hash value should be secretly and timely stored and transmitted to the receiver when performing the real-time decryption process. Namely, for an image database having thousands of images, an equal number of hash values must be generated and securely stored and timely transmitted. This would make the proposed scheme impractical for such a real-time application scenario. In fact, how to securely and timely store and transmit secret key is not easy to process in real world. Besides, setting hash value as part of the secret key against, to some extent, Kerckhoffs' principle, which requires the security of the encryption scheme, completely depends on the secret key [22]. Actually, the encryption and decryption keys should be independent of the plain image in a symmetry encryption scheme. Indeed, many schemes cannot achieve the claimed high security and are impractical for the special application scenario due to the facts pointed in [23–25]. For instance, the encryption scheme presented in [26] was proposed by Wu et al. in 2018, which adopted the deoxyribonucleic acid (DNA) approach to diffuse image and used a two-dimensional Hénon-Sine map- (2D-HSM-) based permutation. Unfortunately, their proposal was broken by the chosen plain image attack in 2020 [27].

Therefore, constructing an encryption scheme which achieves a good trade-off between security and efficiency would be a hard work [28]. Nevertheless, combining the aforementioned analysis, we tentatively propose a novel image encryption scheme based on the PWLCM and the standard map in this study. It is constructed for application scenarios that require lightweight security. Firstly, an external secret key of 256-bit is employed to generate initial states and parameters of the chaotic maps. Then, the permutation-diffusion structure is adopted to encrypt plain image. Different from the pixel-by-pixel way in the permutation process, we perform a bit-level circularly shift operation on the pixels of plain image according to the pseudo-random sequence generated by the PWLCM. Thus, the position and value of the pixel can change simultaneously to enhance the encryption scheme's security. Moreover, the discretized standard map-based permutation method is further employed to improve the security level of the proposed encryption scheme due to its structure of the resembled Feistel network. To help offset efficiency, the operation model of row-by-row and column-by-column is utilized in the diffusion phase to speed up the encryption performance. Furthermore, combining the PWLCM with the standard map, the cryptanalysis complexity of the proposed encryption scheme, which means adding slightly the number of arithmetic operation, will be bigger than only using single chaotic map. Thus, the cryptanalysis for the former will be more difficult than the latter one [2, 29].

The paper contributes the following. (1) Combining the PWLCM (one-dimensional chaotic map) with the standard map (high-dimensional chaotic map) to design encryption scheme, the analysis complexity is increased. (2) A hierarchical diffusion strategy is employed in the standard map-based permutation phase, which enhances the security. (3) The operation model of row-by-row and column-by-column is adopted in the diffusion process to promote the encryption efficiency. (4) A good trade-off between security and efficiency can be achieved, which indicates that the proposed encryption scheme can be used in real applications requiring lightweight security.

The remainder of this paper is structured as follows. Section 2 briefly describes and analyzes the PWLCM and the standard map. Section 3 illustrates the proposed image encryption scheme and the corresponding experimental results. Security analyses are described in Section 4. The paper finishes with concluding remarks.

2. Preliminaries

2.1. PWLCM. The piecewise linear chaotic map (PWLCM) is composed of multiple linear segments [30], which is defined by

$$x_{n+1} = f(x_n, p) = \begin{cases} \dfrac{x_n}{p}, & 0 \le x_n \le p, \\ \dfrac{x_n - p}{0.5 - p}, & p < x_n \le 0.5, \\ f(1 - x_n, p), & 0.5 < x_n \le 1, \end{cases} \quad (1)$$

where $x_n \in (0,1)$, $n \in \{0,1,\ldots,\}$, x_0 is the initial state of PWLCM, and $p \in (0,0.5)$ is the control parameter of the chaotic system. According to the definition of Lyapunov exponent (LE) [31], one can easily get the LE of PWLCM $\lambda = -0.5\ln[p(0.5-p)]$. Clearly, it gets the minimum value $\lambda_{\min} = 2\ln 2 > 0$ at $p = 0.25$ which means the PWLCM has chaotic behavior in the whole definition interval $(0,1)$.

The bifurcations of logistic map [31] and PWLCM are depicted in Figure 1. Figure 2 plots the Lyapunov exponent curves of logistic map and the PWLCM with variation control parameters. From the above results, one can learn that the PWLCM has a wider chaotic range and more complex behavior than logistic map. Therefore, the PWLCM is more suitable for designing image encryption scheme.

2.2. Standard Map. The standard map is an area-preserving map [4], which is defined by

$$\begin{cases} x = (x + y) \bmod (2\pi), \\ y = (y + K\sin(x + y)) \bmod (2\pi), \end{cases} \quad (2)$$

where $(x, y) \in [0, 2\pi] \times [0, 2\pi]$ and K is a positive constant which determines the degree of chaos. One can verify that $(0,0)$, $(\pi, 0)$ are two of the fix points of equation (2). On the other hand, the Kolmogorov-Sinai entropy of the standard map is depicted by $h \approx \ln(K/2)$ valid for $K > 4$ [32].

The phase space of the standard map at $K = 0.971635$ is shown in Figure 3, which also plots its Lyapunov exponent curves with variation of parameters. From the figure, it is clear that the standard map is chaotic in domain except for periodic or quasiperiodic points. To conclude, the standard map has complex dynamic behavior and good chaotic properties. Thus, it can be used for designing image encryption scheme.

For applying equation (2) to permute the image pixels, it is discretized from $[0, 2\pi] \times [0, 2\pi]$ to a square lattice with a width of N [4]. By substituting $x_i = (N/2\pi) \times x$, $y_i = (N/2\pi) \times y$, $i \in \{0,1,\ldots\}$, $K' = (N/2\pi) \times K$ into equation (2), we thus have

$$\begin{cases} x_{i+1} = (x_i + y_i) \bmod N, \\ y_{i+1} = \left(y_i + K\prime \sin\left(\dfrac{2\pi}{N} x_{i+1}\right)\right) \bmod N, \end{cases} \quad (3)$$

where N is the width of the processed image and $K\prime$ is a new positive control parameter. (x_i, y_i) and (x_{i+1}, y_{i+1}) represent the positions of the processed image pixel before and after employing the discretized standard map. Figure 4 plots the permutation results using different rounds of the discretized standard map. From the figure, one can learn that more than 3 rounds should be used to obtain enough security. Accordingly, in this paper, we will apply 3 rounds of the discretized standard map to our proposed encryption scheme, which is deferred further discussion to Section 3.

3. Our Proposed Image Encryption Scheme

3.1. The Encryption Process. In this section, we give in detail the encryption process of the proposed scheme. Without loss of generality, let **P** be the plain image with a size of $MN = M \times N$. Note that **Key** is the external 256-bit secure secret key. To help understand the framework of our proposal, Figure 5 and Algorithm 1 give the flowchart and the pseudo-random code of the proposed image encryption scheme, respectively.

3.1.1. Improved Permutation Process

Step 1. Generating the initial state and the parameter of PWLCM: let $x_h^{(w)}, p_h^{(w)}$ $(h, w = 1, 2)$ be the initial states and the parameters used in permutation and diffusion phases. Given a 256-bit secret **Key**, we calculate two float numbers $x_1^{(1)}, p_1^{(1)}$ using $\mathrm{FN} = \sum_i \mathbf{Key}_i \times 2^{-i}$, $i \in \{1,2,\ldots\}$, based on its first two 64-bit streams, respectively. Similarly, two float numbers $x_1^{(2)}, p_1^{(2)}$ are generated by the last two 64-bit streams of **Key**, respectively.

Step 2. Generating the new initial state and parameter of PWLCM: let x_1^0, p_1 be the new initial state and parameter, respectively. Then, compute their values by $x_1^0 = [(x_1^{(1)} + x_1^{(2)}) \bmod 1]$, $p_1 = [(p_1^{(1)} + p_1^{(2)}) \bmod 0.5]$, where the notation $[\alpha \bmod \beta]$ denotes the remainder of α upon division by β. Namely, for any two real numbers α, $\beta \neq 0$, $[\alpha \bmod \beta] = \alpha - \beta\lfloor \alpha/\beta \rfloor$.

Step 3. Creating the pseudo-random sequence: iterate PWLCM MN times with x_1^0, q_1 to obtain sequence $x_1(1 \times \mathrm{MN})$. To avoid the transient effect, we drop the first N_0 iterated values. We then further quantify x_1 to get $\mathrm{step}_0(1 \times \mathrm{MN})$ using $\mathrm{step}_0 = \mathrm{mod}(\mathrm{round}((|x_1| - \lfloor x_1 \rfloor) \times 10^{15}), 8) + 1$ where $\mathrm{round}(x)$ rounds x to the nearest integer and $b = \mathrm{mod}(a, m)$ returns the remainder after division of a by m. Finally, $\mathrm{step}_0(1 \times \mathrm{MN})$ is reshaped to $\mathrm{step}(M \times N)$ from top to bottom and left to right.

Step 4. Initializing the permutation process: firstly, the 256-bit external secret key is divided into four parts. More concretely, the first three parts are the same length of 53-bit, and the length of the last part is 44-bit. Then, we set each part as the input of $\sum_i \mathbf{Key}_i \times 2^i$, $i \in \{1,2,\ldots\}$ and get the sum of each result K_0. The control parameter PSK is obtained by $[K_0 \bmod 1024]$.

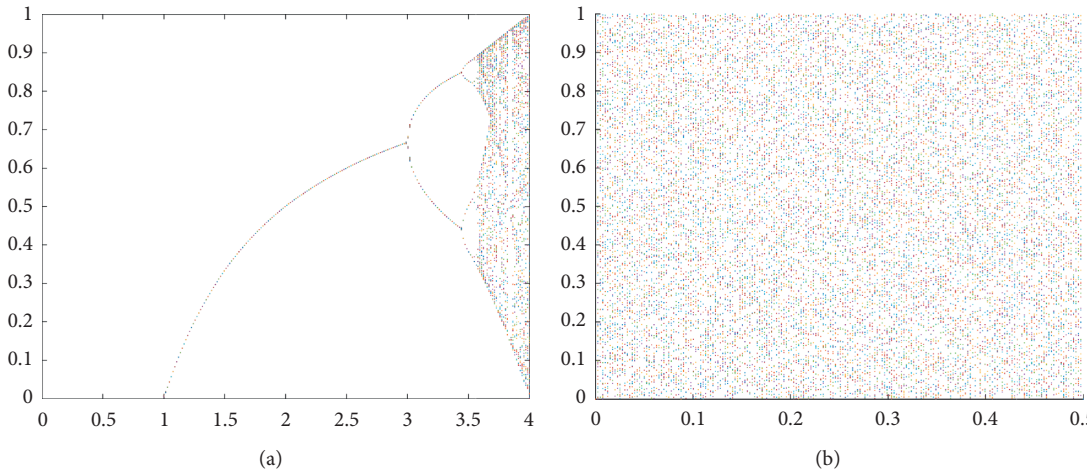


Figure 1: The bifurcation analysis: bifurcations of (a) logistic map and (b) PWLCM.

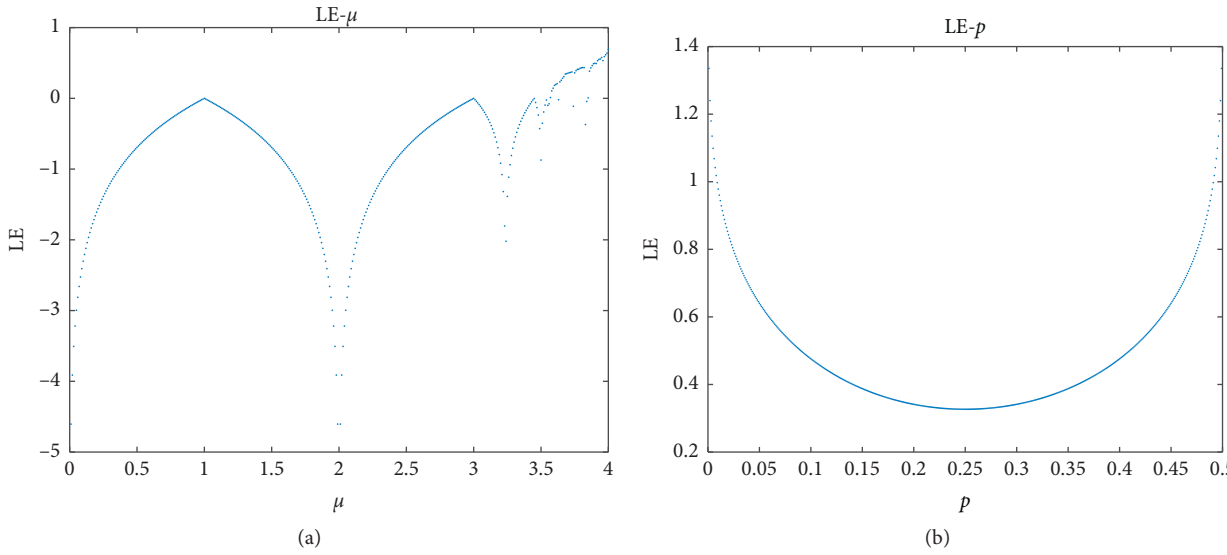


Figure 2: The Lyapunov exponent (LE) analysis: Lyapunov exponent curves of (a) logistic map and (b) PWLCM.

Finally, initialize the permuted image as $\mathbf{P}_1(M \times N)$. Let $i_1 = 0, 1, \ldots, (M-1), i_2 = 0, 1, \ldots, (N-1)$; calculating new position (s_1, s_2) based on $s_1 = [(i_1 + i_2) \bmod M]$, $s_2 = [(i_2 + \text{round}(\text{PSK} \times \sin((2\pi/N)s_1))) \bmod N]$.

Step 5. Performing the bit-level circularly shift and the pixel-level permutation simultaneously: firstly, we obtain the bit-level circle shift direction dir by using $\text{dir} = 2 \times \text{dir}_0 - 1$ where $\text{dir}_0 = [\text{step}(i_1 + 1, i_2 + 1) \bmod 2]$. We then perform a bit shift on the plain image's pixel $P(i_1 + 1, i_2 + 1)$ by a shift amount $\text{SA} = \text{dir} \times \text{step}(i_1 + 1, i_2 + 1)$. In particular, positive SA shifts toward the end of binary array and negative *SA* shifts toward the beginning. At the same time, permute circularly shifted pixel to the corresponding position in permuted image $\mathbf{P}_1(s_1 + 1, s_2 + 1)$. Let $\mathbf{P} = \mathbf{P}_1$ and iterate above shuffling method iter_p times to obtain the final permuted image $\mathbf{P}_1$.

3.1.2. High-Speed Diffusion Process

Step 6. Getting chaotic sequences: similar to the way in the permutation phase, we first calculate the new initial state and the parameter for PWLCM. After assigning the first two 64-bit streams of **Key** to float numbers $p_2^{(1)}, x_2^{(1)}$, their decimal values can obtained by the way in **Step 1**, respectively. In a similar manner, the float numbers $p_2^{(2)}, x_2^{(2)}$ are generated by the last two 64-bit streams of **Key**, respectively. Then, we can obtain the new initial state $x_2^{(0)}$ and the parameter p_2 by $x_2^{(0)} = [(x_2^{(1)} + x_2^{(2)}) \bmod 1]$, $p_2 = [(p_2^{(1)} + p_2^{(2)}) \bmod 0.5]$. Lastly, we iterate PWLCM $(M + N)$ times to get

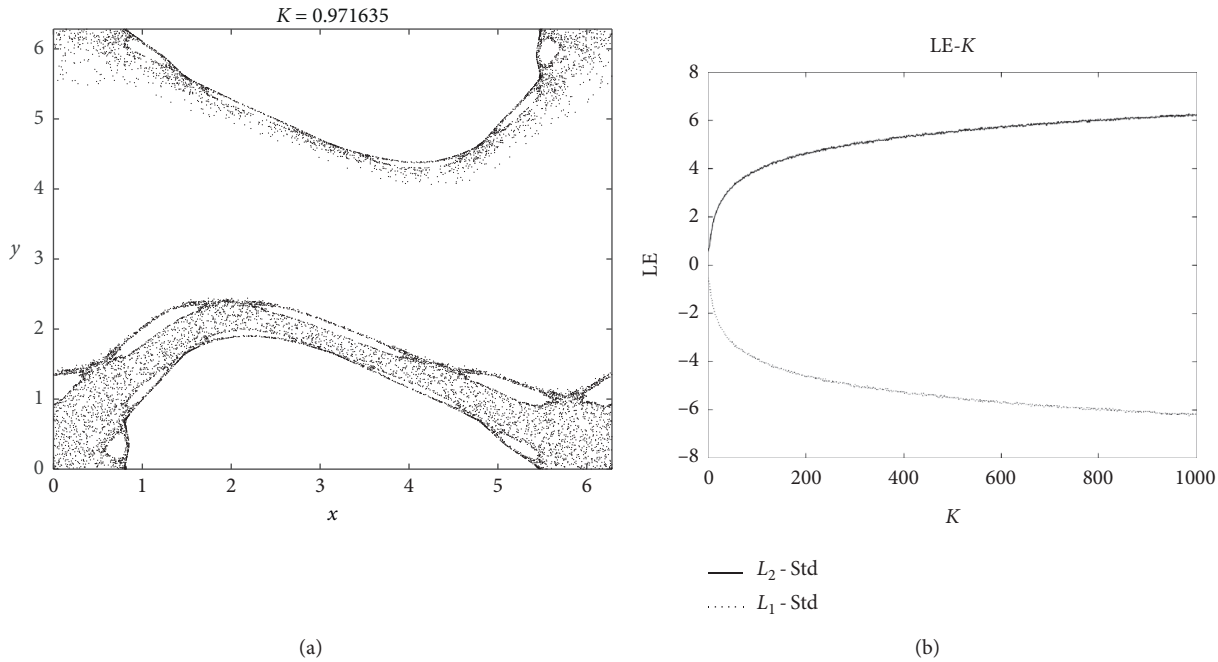


Figure 3: The standard map analysis: (a) the phase space of the standard map at $K = 0.971635$; (b) the Lyapunov exponent curves of the standard map with variation of parameters.

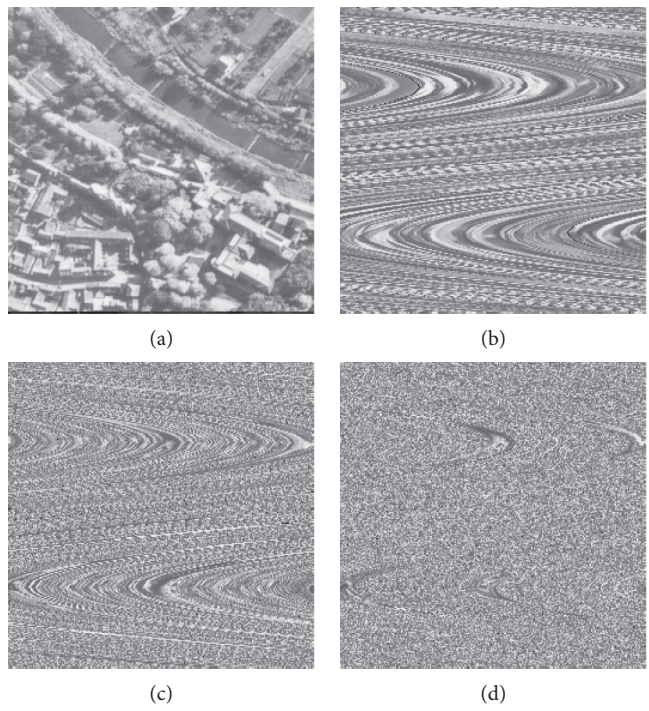


Figure 4: The discretized standard map-based permutation results: (a) the original image 5.1.10.tiff and (b, c, d) permuted results using 1, 2, and 3 rounds of the discretized standard map.

$x_2(1 \times (M+N))$ and further generate a row array $x_2^r(1 \times M)$ and a column array $x_2^c(N \times 1)$.

Step 7. Generating four seed vectors: firstly, we quantify x_2^r, x_2^c to obtain two chaos-based seed vectors $\text{key}_r(1 \times M), \text{key}_c(N \times 1)$ by $\text{key}_r = [\lfloor (x_2^r - \lfloor x_2^r \rfloor) \times 10^{14} \rfloor \bmod 256]$, $\text{key}_c = [\lfloor (x_2^c - \lfloor x_2^c \rfloor) \times 10^{14} \rfloor \bmod 256]$, respectively. Then, in a similar way, two initial vectors $\text{ivr}(1 \times M), \text{ivc}(N \times 1)$ are calculated by $\text{ivr} = [\lfloor |x_2^r| \times 10^{14} \rfloor \bmod 256]$, $\text{ivc} = [\lfloor |x_2^c| \times 10^{14} \rfloor \bmod 256]$, respectively.

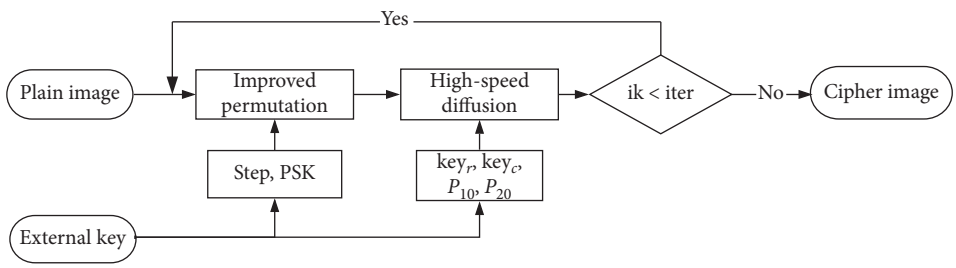


Figure 5: The flowchart of the proposed image encryption scheme.

(1) Input the plain image $\mathbf{P}(M \times N)$.
(2) Create an initial state and a parameter of PWLCM: (x_{10}, p_1).
(3) Generate step$(M \times N)$.
(4) Create a new initial state and a parameter of PWLCM: x_{20}, p_2.
(5) Calculate the vectors: key$_r(1 \times M)$, key$_c(N \times 1)$, ivr$(1 \times M)$, ivc$(N \times 1)$.
(6) **for** $ik = 1$: iter **do**
(7) Initialize permuted image $\mathbf{P}_1(M \times N)$.
(8) **for** $ik_1 = 1$: iter$_p$ % Improved permutation process **do**
(9) **for** $i_1 = 0$: $(M - 1)$ **do**
(10) **for** $i_2 = 0$: $(N - 1)$ **do**
(11) Calculate (s_1, s_2) and a shift amount SA.
(12) $\mathbf{P}_1(s_1 + 1, s_2 + 1) = \text{bitcircshift}(\mathbf{P}(i_1 + 1, i_2 + 1), \text{SA})$.
(13) **end for**
(14) **end for**
(15) $\mathbf{P} = \mathbf{P}_1$.
(16) **end for**
(17) Initialize medium and encrypted image: $\mathbf{P}_2(M \times N)$, $\mathbf{P}_3(M \times N)$.
(18) **for** $ik_2 = 1$: iter$_d$ **do** % High-speed diffusion
(19) **for** $i = 1$: M **do**
(20) Perform row diffusion to get $\mathbf{P}_2(M \times N)$.
(21) **end for**
(22) **for** $j = 1$: N **do**
(23) Perform column diffusion to obtain $\mathbf{P}_3(M \times N)$.
(24) **end For**
(25) $\mathbf{P}_1 = \mathbf{P}_3$.
(26) **end for**
(27) $\mathbf{P} = \mathbf{P}_3$.
(28) **end for**
(29) Output the final encrypted image $\mathbf{P}_3(M \times N)$.

Algorithm 1: The proposed image encryption scheme.

Step 8. Perform the high-speed diffusion process: set $i = 1, 2, \ldots, M$, performing the element-level circularly shift operation on key$_r$ by i shift amounts to get $r_1(1 \times M)$. Then, we can obtain the row-by-row diffusion encrypted image $\mathbf{P}_2$ using equation (4). Similarly, set $j = 1, 2, \ldots, M$, we utilize circularly shift process on key$_c$ by j steps to get a seed vector $c_1(M \times 1)$. Then, we get the column-by-column diffusion encrypted image $\mathbf{P}_3$ based on equation (5). Let $\mathbf{P}_1 = \mathbf{P}_3$; iterate iter$_d$ times to get the final diffusion encrypted image $\mathbf{P}_3$.

$$\mathbf{P}_2(i, :) = \begin{cases} \text{mod}(\mathbf{P}_1(i, :) + r_1, 256) \oplus \text{ivr}, & \text{if} i = 1, \\ \text{mod}(\mathbf{P}_1(i, :) + r_1, 256) \oplus \mathbf{P}_2((i - 1), :), & \text{else}, \end{cases} \tag{4}$$

where $a \oplus b$ returns the bitwise XOR of a and b.

$$\mathbf{P}_3(:, j) = \begin{cases} \text{mod}(\mathbf{P}_2(:, j) + c_1, 256) \oplus \text{ivc}, & \text{if } j = 1, \\ \text{mod}(\mathbf{P}_2(:, j) + c_1, 256) \oplus \mathbf{P}_3(:, (j - 1)), & \text{else}. \end{cases} \tag{5}$$

Step 9. Finally, set $\mathbf{P} = \mathbf{P}_3$, performing repeatedly single round encryption process *iter* times to get the final encrypted image $\mathbf{P}_3$.

3.2. The Decryption Process. The proposed image encryption scheme belongs to a symmetry cipher algorithm. Thus, the decryption process for it is simply the inversion of the encryption process. We do not include the steps here to conserve space.

3.3. Numerical Experimental Results. This section discusses the numerical experimental results for the proposed image encryption scheme. Our experiments in this paper are performed on a laptop computer equipped with an Intel(R) Core(TM) i7-5500U CPU @ 2.40 GHz, 4 GB memory, and Windows 7 operating system. We adopt Matlab platform to implement the proposed encryption scheme. On the other hand, three plain images (5.1.11.tiff to 5.1.13.tiff) are randomly selected from the USC-SIPI Image Database [33]. The function rand which returns uniformly distributed pseudo-random numbers is utilized to generate **Key**. For convenience, we set $(\text{iter}_p, \text{iter}_d, \text{iter}) = (3, 1, 2)$. It is noted that the value of iter should be large enough to maintain the desired security when it comes to the real-world application. After performing the proposed encryption scheme, the obtained experimental results are shown in Figure 6. From the results, the observer cannot obtain any visual information about plain image. The decrypted images are identical to the plain images. Moreover, even if the encrypted image is chosen by any arbitrary way, the receiver can obtain the original image only using the corrective secret key. This indicates that the proposed encryption scheme is effective in the basic sense of symmetry algorithm.

4. Performance Analysis

4.1. Histogram Analysis. An image histogram gives the entire intensity value distribution visually in an image [34]. An ideal encryption scheme should be able to generate final encrypted image uniformly distributed. To see this, Figure 7 plots the histograms of plain images 5.1.11.tiff, 5.1.12.tiff, and 5.1.13.tiff, the final encrypted images generated by the proposed encryption scheme, and the corresponding recovered images, respectively. From the figure, one can obviously observe that the histograms of the plain images are statistically significant while the encrypted images are uniformly distributed. Thus, the histogram will not leak any useful information to attackers.

In addition, the chi-square test [35] is further applied to quantitatively evaluate the uniformity of the histograms. It is mathematically defined by

$$\begin{cases} \chi^2_{\text{test}} = \sum_{v=0}^{255} \dfrac{(f_v - d)^2}{d}, \\ d = \dfrac{M \times N}{256}, \end{cases} \tag{6}$$

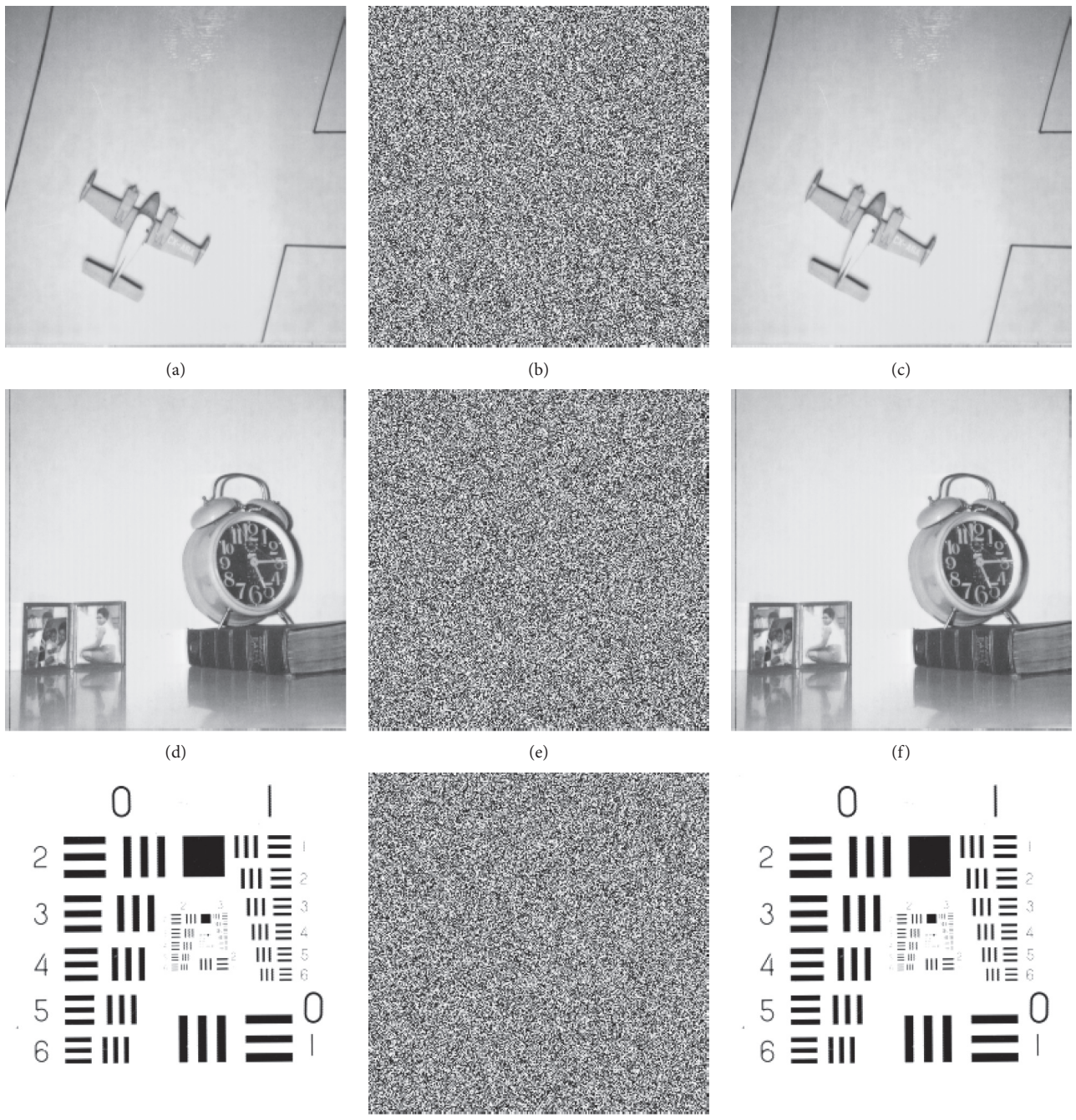
where f_v is the frequency of each gray level $v \in \{0, 1, \ldots, 255\}$ for a 8-bit gray-scale image and $M \times N$ is the size of the tested image. With a significance level of 0.05, the theoretical chi-square value is $\chi^2_{0.05} = 293.2478$ [35]. A smaller experimental value indicates the more uniform distribution for a specific image histogram. In our proposed test analysis, we randomly selected 6 gray-scale plain images (5.1.09.tiff through 5.1.14.tiff). The corresponding test results are listed in Table 1. Moreover, we also perform the chi-square tests on the scheme presented in [18]. Their test results are also included in Table 1. Based on the mean value and pass rate in the table, we can conclude that proposed encryption scheme provides better performance than the compared algorithm.

4.2. Correlation Analysis. The adjacent pixels in a natural image are strongly correlated. A good image encryption scheme should have a low correlation in different adjacent directions [13]. To evaluate the proposed encryption scheme, we randomly select 5000 pairs of pixels of the encrypted image in the horizontal, vertical, and diagonal directions, respectively. Then, the correlation coefficient of each pair, defined in (7), is calculated.

$$\begin{cases} E(x) = \dfrac{1}{N} \sum_{i=1}^{N} x_i, \\ D(x) = \dfrac{1}{N} \sum_{i=1}^{N} \left(x_i - E(x)\right)^2, \\ \text{cov}(x, y) = \dfrac{1}{N} \sum_{i=1}^{N} \left(x_i - E(x)\right)\left(y_i - E(y)\right), \\ r_{xy} = \dfrac{\text{cov}(x, y)}{\sqrt{D(x)} \times \sqrt{D(y)}}, \end{cases} \tag{7}$$

where x and y are the adjacent pixel sequences selected from the different directions, $N = 5000$ is the size of the selected pixel sequence, and $E(x)$ and $D(x)$ are the mean and the standard deviation of x, respectively. If the selected pixel sequences have low correlation, their coefficient calculated by equation (7) should be close to 0. Otherwise, it will be close to 1 [13]. Figure 8 shows the selected pixel sequence distributions in different directions, and Table 2 lists the calculated results where H, V, and D denote the horizontal, vertical, and diagonal directions, respectively. Besides, we also calculate the correlation coefficients for the encrypted image generated by the encryption scheme presented in [18], and the results are also included in Table 2. While the distributions in plain image are close to diagonal line, the distributions in the encrypted image are randomly scattered. Similarly, the quantitative results in plain image are close to 1 while the resulting values in the encrypted image are close to 0. Therefore, both figures and tables show that the plain image has strong relationships but that weakness exists in the encrypted image. Furthermore, in most cases, the proposed encryption scheme outperforms the compared method. From the above analysis, we infer that our proposed encryption scheme can efficiently eliminate the strong correlation of the adjacent pixels.

4.3. Information Entropy Analysis. Information entropy measures the randomness and unpredictability of a random sequence k [36]. It is defined by



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 6: Numerical experimental results: (a, d, g) plain images 5.1.11.tiff, 5.1.12.tiff, and 5.1.13.tiff; (b, e, h) encrypted images corresponding to (a), (d), and (g); (c, f, i) recovered images corresponding to (b), (e), and (h).

$$H(k) = -\sum_{i=0}^{2^L-1} P(k_i)\log_2 P(k_i), \tag{8}$$

where $P(k_i)$, $i \in \{0, 1, \ldots, 2^L - 1\}$, denotes the probability of the ith symbol k_i. The ideal entropy value for an image with 2^8 gray levels should be 8 [37]. We randomly test 6 plain images (5.1.09.tiff through 5.1.14.tiff) from the USC-SIPI Image Database. The calculated results of the encrypted images generated by the proposed encryption scheme and the method presented in [18] are listed in Table 3. From the mean value results, we deduce that the proposed encryption scheme provides better randomness. Moreover, the local Shannon entropy (LSE) is adopted to evaluate the randomness of an image from local view [38]. It is computed by

$$\overline{H}_{k,T_B}(m) = \sum_{i=1}^{k} \frac{H(S_i)}{k}, \tag{9}$$

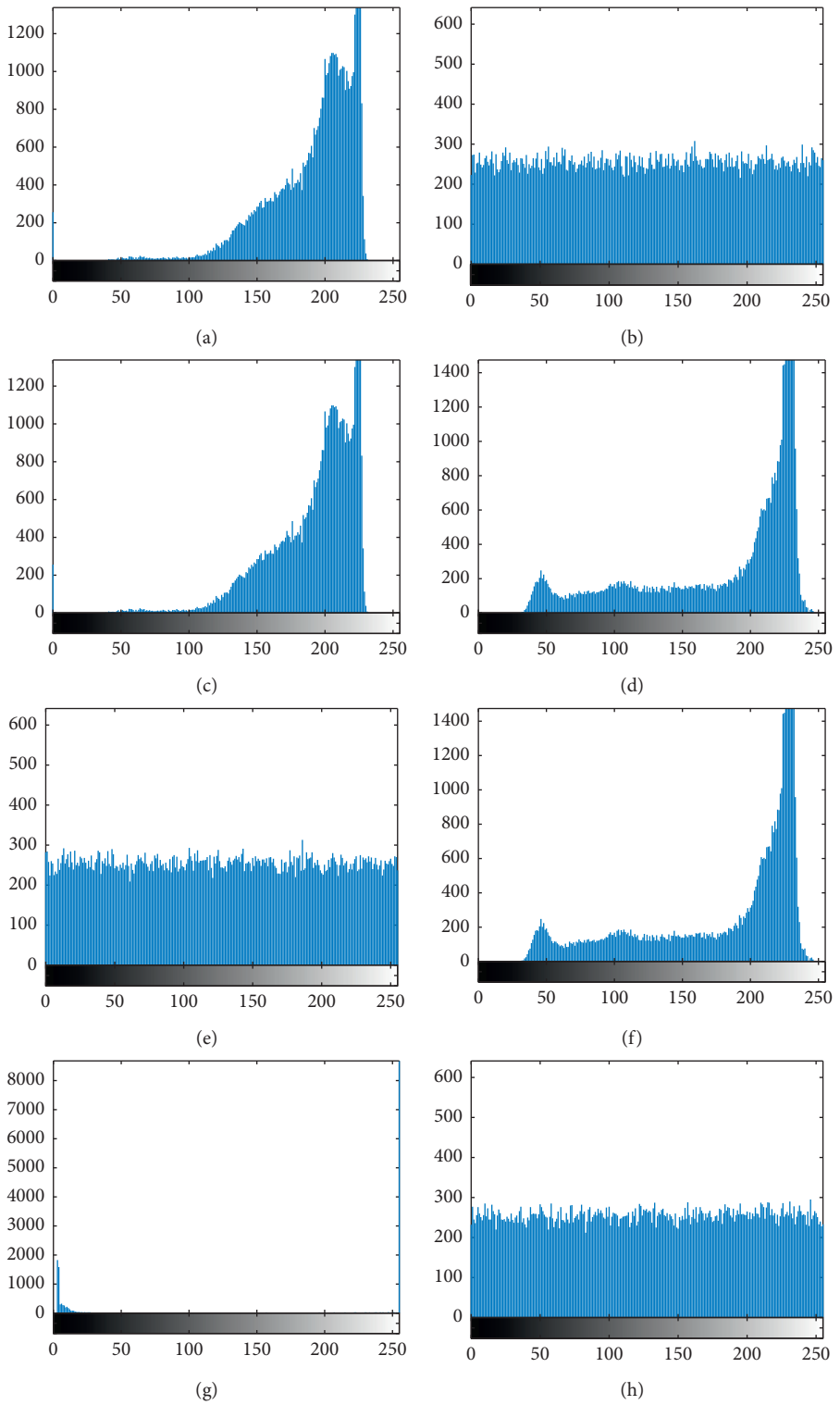
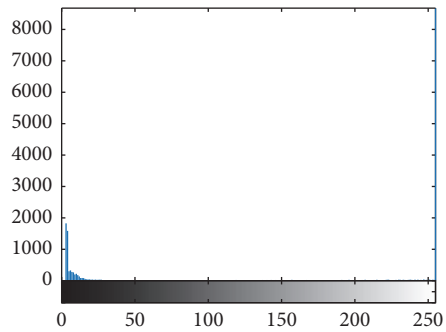


Figure 7: Continued.

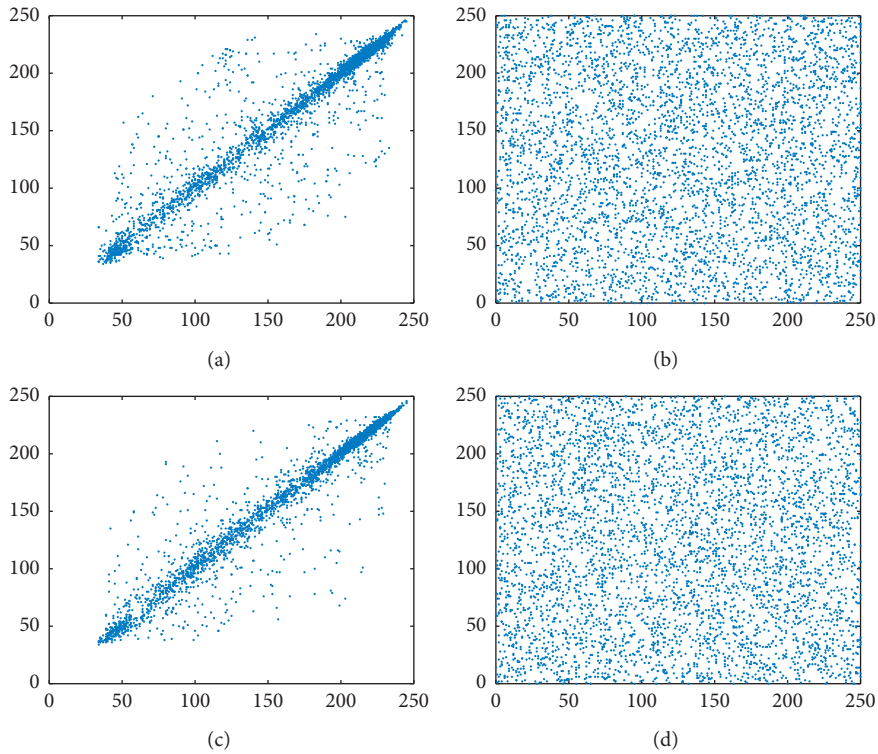


(i)

Figure 7: Histogram analysis: (a, d, g) the histograms of plain images 5.1.11.tiff, 5.1.12.tiff, and 5.1.13.tiff; (b, e, h) the histograms of the respective encrypted results; (c, f, i) the histograms of the respective recovered images.

Table 1: Chi-square test analysis.

File name	$\chi^2_{255,0.05}$	χ^2 (proposed)	χ^2 (reference [18])
5.1.09.tiff	293.2478	251.7578	248.8516
5.1.10.tiff	293.2478	227.8750	217.4219
5.1.11.tiff	293.2478	251.4453	228.6016
5.1.12.tiff	293.2478	276.6172	259.8125
5.1.13.tiff	293.2478	229.3281	362.8203
5.1.14.tiff	293.2478	238.1797	248.6172
Mean	293.2478	211.6004	224.4464
Pass rate	—	6/6	5/6



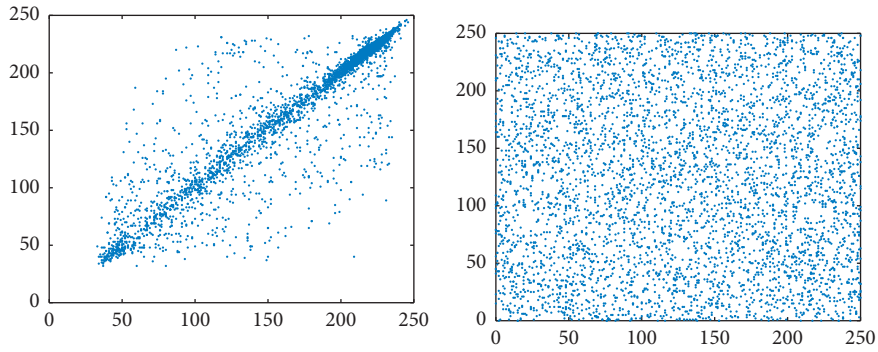


Figure 8: Correlation analysis: (a, c, e) the pixel sequence distributions of the plain image 5.1.12.tiff in the horizontal, vertical, and diagonal directions, respectively; (b, d, f) the pixel distributions of the corresponding encrypted image in the horizontal, vertical, and diagonal directions.

Table 2: Correlation analysis.

File name	Directions	Plain image	Proposed	Reference [18]
5.1.09.tiff	H	0.9051	−0.0062	−0.0109
	V	0.9417	0.0278	−0.0067
	D	0.9078	0.0061	−0.0089
5.1.10.tiff	H	0.9012	0.0251	−0.0138
	V	0.8553	−0.0228	−0.0043
	D	0.8338	−0.0098	−0.0221
5.1.11.tiff	H	0.9592	0.0307	−0.0133
	V	0.9419	−0.0250	0.0121
	D	0.8878	−0.0261	−0.0042
5.1.12.tiff	H	0.9538	−0.0031	−0.0059
	V	0.9747	−0.0196	−0.0148
	D	0.9327	−0.0110	0.0072
5.1.13.tiff	H	0.8762	−0.0185	0.0102
	V	0.8597	0.0043	−0.0104
	D	0.7537	0.0219	−0.0092
5.1.14.tiff	H	0.9477	0.0059	0.0122
	V	0.9009	−0.0149	−0.0227
	D	0.8478	0.0094	−0.0073

Table 3: Information entropy analysis.

File name	Plain image	Proposed	Reference [18]
5.1.09.tiff	6.7093	7.9972	7.9973
5.1.10.tiff	7.3118	7.9975	7.9976
5.1.11.tiff	6.4523	7.9972	7.9975
5.1.12.tiff	6.7057	7.9969	7.9971
5.1.13.tiff	1.5483	7.9975	7.9960
5.1.14.tiff	7.3424	7.9974	7.9973
Mean	6.0116	7.9973	7.9971

where k is the total number of selected nonoverlapping image block S_i and T_B is noted as the number of pixel in each block. With the significance level $\alpha = 0.05$ and the parameters $(k, T_B) = (30, 1936)$, the theoretical LSE interval is $(7.9019, 7.9030)$ for a 8-bit image [38]. To see our proposal, Table 4 gives the LSE values for the proposed method and the compared algorithm presented in [18]. It is obvious that the LSE values of the encrypted images obtained by our proposal are in the theoretical interval, and most of the results are bigger than the compared algorithm, which means that the proposed method has better randomness in terms of local view.

4.4. Key Space Analysis. Key space is composed of all possible secret keys. Specifically, the key space of a high security encryption scheme should larger than 2^{100} to counter brute

Table 4: Local Shannon entropy (LSE) analysis.

File name	Plain image	Proposed	Reference [18]
5.1.09.tiff	6.2033	7.9047	7.9006
5.1.10.tiff	7.0258	7.9024	7.9039
5.1.11.tiff	5.2050	7.9052	7.9034
5.1.12.tiff	5.1867	7.9026	7.9019
5.1.13.tiff	1.4765	7.9047	7.9014
5.1.14.tiff	6.6818	7.9036	7.9043
Mean	5.2965	7.9039	7.9026

force attack [39]. In our proposed encryption scheme, the key space is generated by a 256 random bit string, which is large enough to satisfy the standard requirement. Moreover, Table 5 lists the key spaces of five schemes. It is clear that the proposed encryption scheme can resist the brute force attack and can be comparable with the methods proposed in [18, 40–42].

4.5. Key Sensitivity Analysis. The key sensitivity means that a slight disturbance of the secret key should generate a totally different result in both the encryption and decryption processes [43]. For a strong robust encryption scheme, it should be extremely sensitive to its secret key variations. The two-dimensional correlation coefficient c_{AB} between two random images A, B is adopted to examine the key sensitivity, which is defined as

$$\begin{cases} \overline{A} = \dfrac{1}{H \times W} \displaystyle\sum_{i=1}^{H} \sum_{j=1}^{W} A_{i,j}, \\ \overline{B} = \dfrac{1}{H \times W} \displaystyle\sum_{i=1}^{H} \sum_{j=1}^{W} B_{i,j}, \\ \mathrm{cov}_{AB} = \dfrac{1}{H \times W} \displaystyle\sum_{i=1}^{H} \sum_{i=1}^{W} \left(A_{i,j} - \overline{A}\right)\left(B_{i,j} - \overline{B}\right), \\ D(A) = \dfrac{1}{H \times W} \displaystyle\sum_{i=1}^{H} \sum_{j=1}^{W} \left(A_{i,j} - \overline{A}\right), \\ D(B) = \dfrac{1}{H \times W} \displaystyle\sum_{i=1}^{H} \sum_{j=1}^{W} \left(B_{i,j} - \overline{B}\right), \\ c_{AB} = \dfrac{\mathrm{cov}_{AB}}{\sqrt{D(A)} \times \sqrt{D(B)}}, \end{cases} \tag{10}$$

where $H \times W$ is the size of the tested image. Generally, the smaller the coefficient value, the lower correlation between two random images [14, 44].

In our proposed key sensitivity analysis, we first randomly generate a 256-bit secret **Key**. To obtain two new keys with only 1-bit difference, which are denoted as $\mathbf{Key}_i$ $(i = 1, 2)$, respectively, we repeatedly change one bit of **Key** two times. We then perform the proposed encryption scheme with **Key** and $\mathbf{Key}_i$ on the plain image 5.1.13.tiff to get the respective encrypted images **C** and $\mathbf{C}_i$ $(i = 1, 2)$. The different cipher images are shown in Figure 9, which also plots the differences $|\mathbf{C} - \mathbf{C}_i|$ $(i = 1, 2)$. As can be seen, the encrypted images generated by the keys with slight difference are completely different. Additionally, we calculate the respective two-dimensional correlation coefficients between **C** and $\mathbf{C}_i$. Table 6 gives the calculated results which are close to 0. Hence, the proposed encryption scheme is sensitive to its secret key in the cipher process.

For the decryption analysis process, we decipher **C** with **Key** and $\mathbf{K}_i$ $(i = 1, 2)$ to obtain the recovered images **D** and $\mathbf{D}_i$, respectively. The different decipher results are shown in Figure 10, which also plots the differences $|\mathbf{D} - \mathbf{D}_i|$ $(i = 1, 2)$. Moreover, the correlation coefficients between **D** and $\mathbf{D}_i$ are listed in Table 7. From the figure and table, one can see that the proposed encryption scheme is sensitive to its secret key in the decipher process. Combining the aforementioned encryption key sensitivity analysis, we conclude that the proposed image encryption scheme is extremely sensitive to secret key.

4.6. Noise and Data Loss Attack Analysis. Traditionally, when an encrypted image is transmitted through an unsecure network or stored in physical medias, it is easy to surfer from the noise perturbation or the data loss. So, a strong robust image encryption scheme should have the ability to minimize the bad effect caused by the noise perturbation and the data loss [45]. In our proposed analysis, we adopt peak signal-to-noise ratio (PSNR) to measure the quality of the reconstruction [46, 47]. Given a gray-scale image I_1 with size of $M \times N$ and its noisy approximation I_2, it is defined as

$$\begin{cases} \mathrm{MSE} = \dfrac{1}{M \times N} \displaystyle\sum_{i=1}^{M} \sum_{j=1}^{N} \left(I_1(i, j) - I_2(i, j)\right)^2, \\ \mathrm{PSNR} = 10 \times \log_{10} \dfrac{255 \times 255}{\mathrm{MSE}} \ (\mathrm{dB}), \end{cases} \tag{11}$$

where MSE denotes the mean square error between I_1 and I_2. Typical values for the PSNR in lossy image compression are between 30 dB and 50 dB, provided the bit depth is 8 bits, where the higher is better [48]. We perform the proposed encryption scheme on plain image 5.1.10.tiff to generate the encrypted image. Then, we add salt and pepper noises (SPNs) to the encrypted image with 0.01, 0.02, and 0.03 densities, respectively. As a result, Figure 11 plots these noisy results and their recovered images, respectively. As can be seen, the

Table 5: Key space analysis.

Algorithm	Reference [40]	Reference [41]	Reference [18]	Reference [42]	Proposed
Key size	2^{256}	2^{299}	2^{199}	$2^{384} \times 10^{38}$	2^{256}

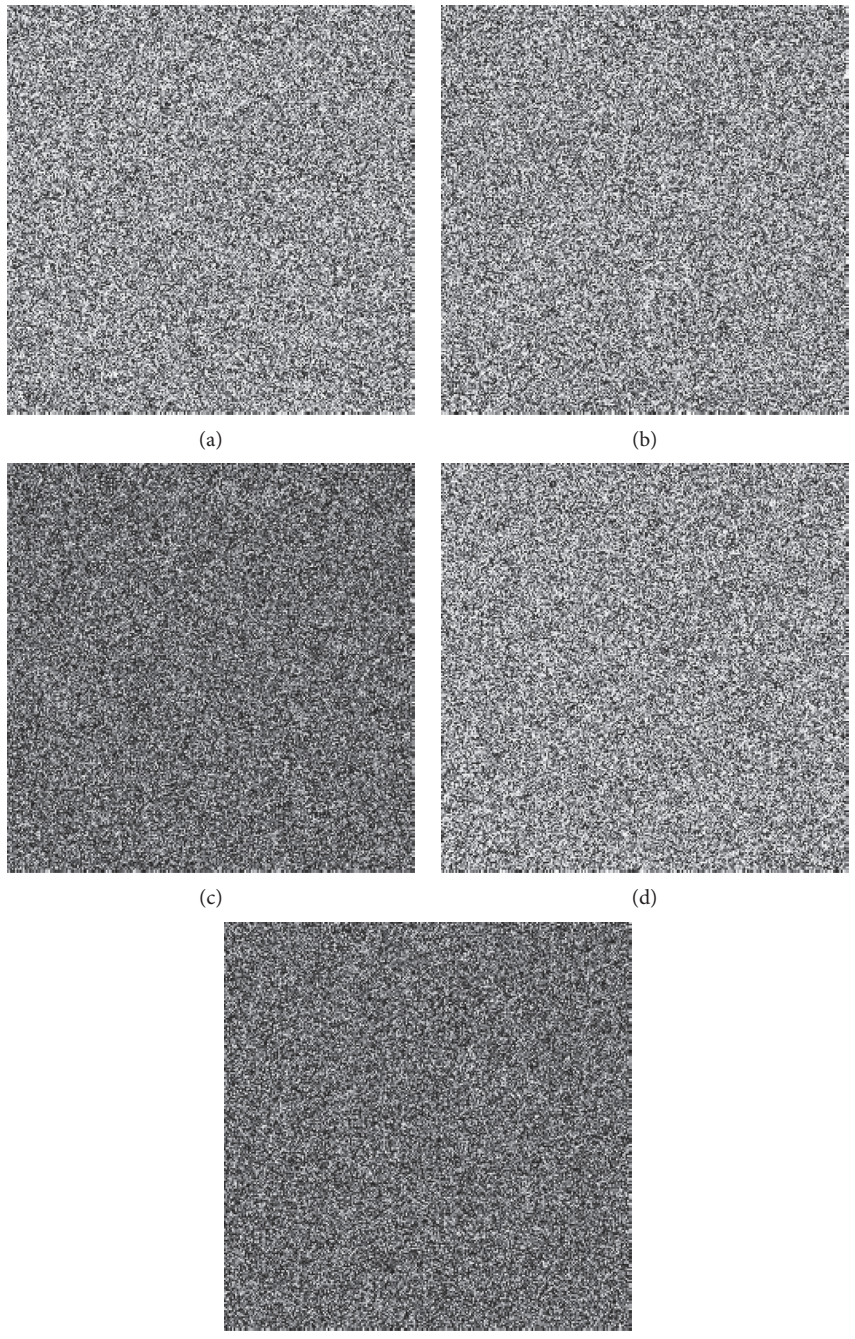


Figure 9: Encryption key sensitivity analysis: (a) the cipher image $\mathbf{C}$; (b, d) the cipher images $\mathbf{C}_i\,(i = 1, 2)$; (c, e) the differences $|\mathbf{C} - \mathbf{C}_i|\,(i = 1, 2)$.

recovered images are visually recognizable. Moreover, Table 8 lists the PSNR (in dB) values which indicates that the proposed encryption scheme can resist noise affect. It also shows the results computed by the compared algorithms presented

Table 6: Encryption key sensitivity analysis.

	$\mathbf{C}_1$	$\mathbf{C}_2$
$\mathbf{C}$	−0.0060	$-2.1728e-04$

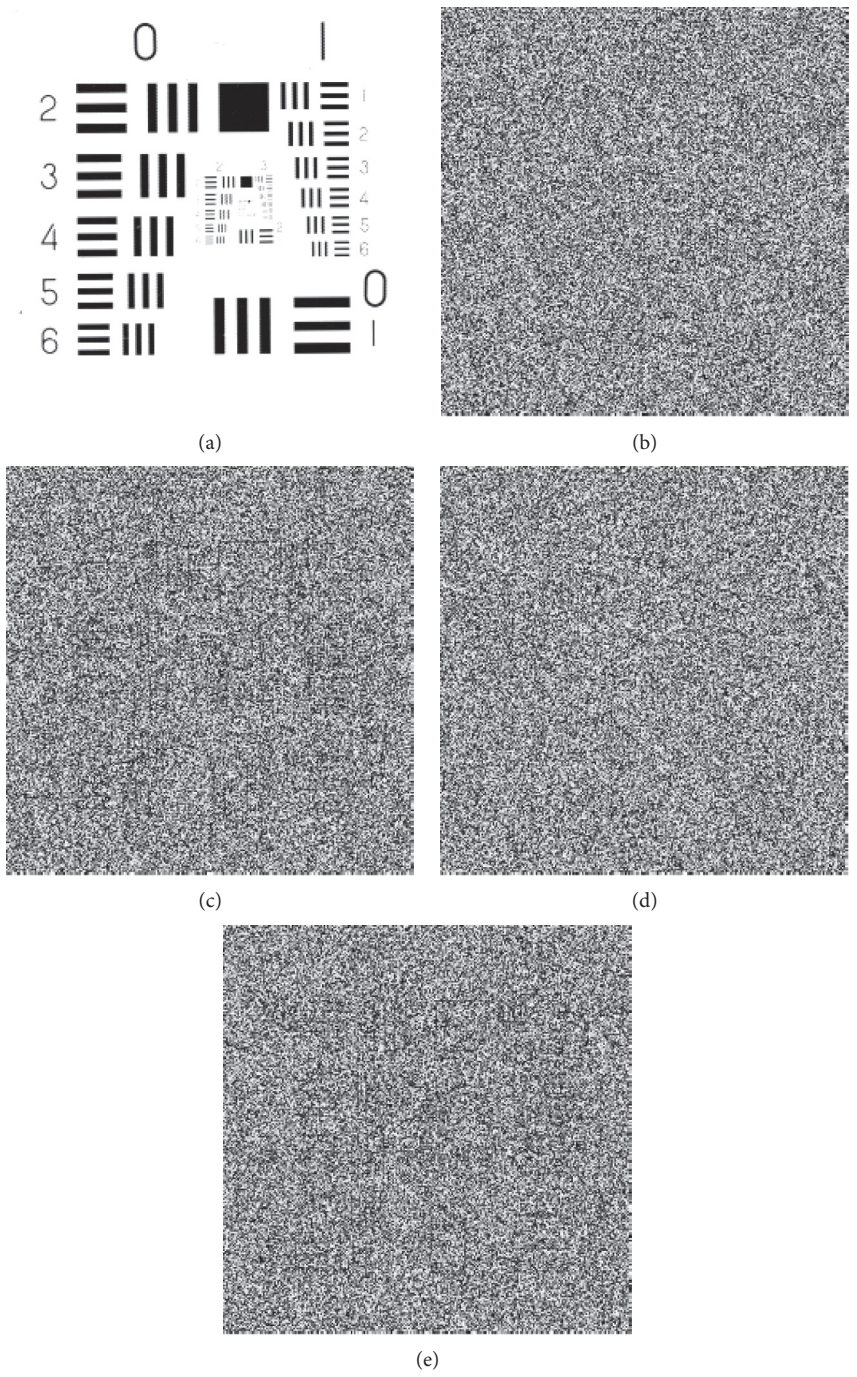


Figure 10: Decryption key sensitivity analysis: (a) the recovered image **D**; (b, d) the recovered images $\mathbf{D}_i$ $(i = 1, 2)$; (c, e) the differences $|\mathbf{D} - \mathbf{D}_i|$ $(i = 1, 2)$.

in [18, 40–42]. Clearly, our proposal can be comparable with the recent works. Likewise, the Gaussian white noise with constant mean and different variances is also applied to evaluate the ability of resisting noise attack. The tested results are given in Table 9 which carries out that our proposal can also resist the Gaussian white noise.

Similarly, we perform the occlusion attack analysis. Figure 12 shows the loss results by cutting (1/8), (1/8), and (1/4) of the encrypted image. On the other hand, Table 10 gives the PSNR values between the plain image and the reconstructed results. Both figure and table demonstrate that our proposed encryption scheme can offer

Table 7: Decryption key sensitivity analysis.

	$\mathbf{D}_1$	$\mathbf{D}_2$
D	0.0014	0.0040

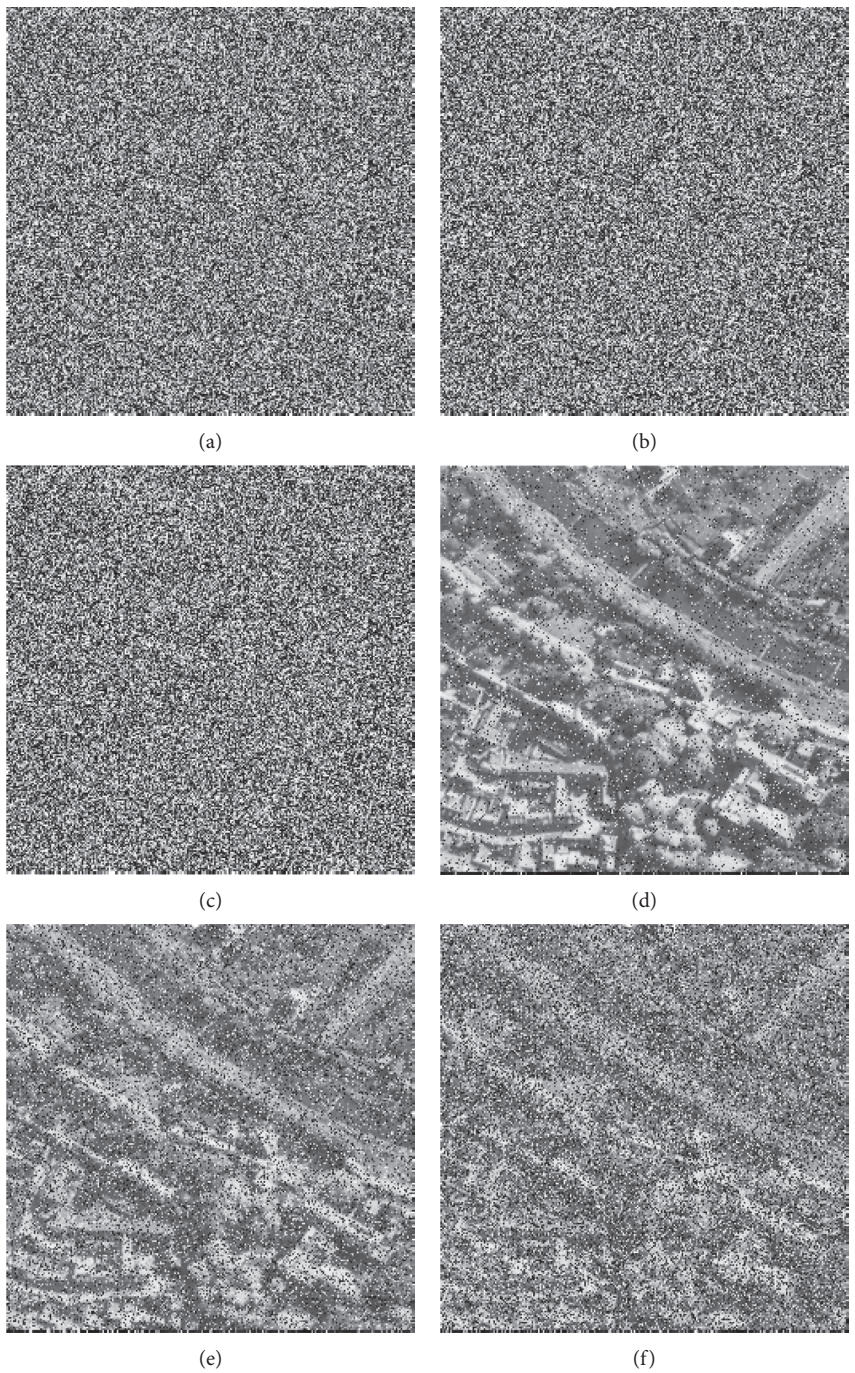


Figure 11: Noise attack analysis: (a, b, c) the noisy encrypted images by SPN with 0.01, 0.03, and 0.05 densities; (d, e, f) the deciphered images of (a), (b), and (c).

Table 8: Noise attack analysis-I (dB).

Noise density	0.01	0.03	0.05
Proposed	32.8218	30.7871	29.9716
Reference [18]	28.6923	28.6937	28.7211
Reference [40]	28.7044	28.7071	28.6974
Reference [41]	32.5622	29.9543	29.7082
Reference [42]	30.2173	29.4494	29.3079

Table 9: Noise attack analysis-II (dB).

Variances	0.00001	0.00005	0.0007
Proposed	28.7121	28.7019	28.6937
Reference [18]	28.7113	28.7000	28.7132
Reference [40]	28.7113	28.7207	28.6930
Reference [41]	28.7114	28.7080	28.7246
Reference [42]	28.7114	28.7259	28.7147

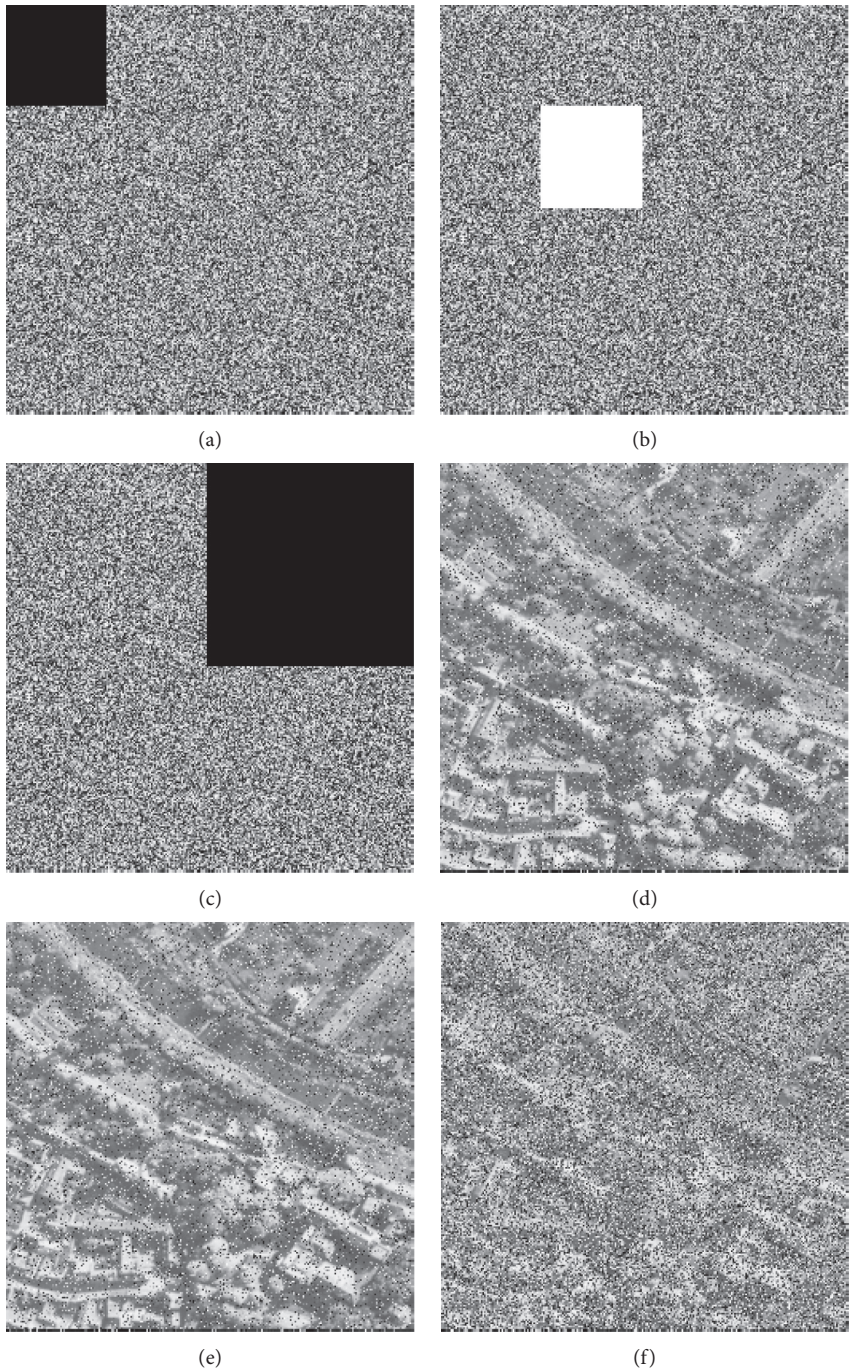


Figure 12: Data loss analysis: (a, b, c) the loss results by cutting (1/8), (1/8), and (1/4) of the encrypted image; (d, e, f) the deciphered images of (a), (b), and (c).

Table 10: Data loss analysis (dB).

Loss rate	1/8	1/8	1/4
Proposed	32.1492	32.3773	29.8749
Reference [18]	28.6953	28.7031	28.7158
Reference [40]	28.6997	28.6939	28.6945
Reference [41]	30.9643	30.9368	30.9225
Reference [42]	34.8468	33.4362	33.4350

cutting resistance. Combining with above noise attack analysis, we infer that the proposed encryption scheme provides resistance against noise and data loss effects.

4.7. Differential Attack Analysis. The differential attack analysis is referred to how tiny change in the plain image can influence the encrypted image. More specifically, the invader can make a slight change in the plain image, and then examine their encryption difference, which aims to get the information about the secret key. Therefore, a high-level security encryption scheme should withstand this kind of attack [29, 49]. The number of pixels change rate (NPCR) and the unified average changing intensity (UACI), defined in equation (12), are used to numerically evaluate the ability to resist differential attack.

$$\begin{cases} \text{NPCR} = \dfrac{\sum_{i,j} D(i,j)}{M \times N} \times 100\%, \\ \text{UACI} = \dfrac{1}{M \times N}\left[\dfrac{\sum_{i,j}\left|C_1(i,j) - C_2(i,j)\right|}{255}\right] \times 100\%, \end{cases} \tag{12}$$

where $C_1(i,j)$ and $C_2(i,j)$, $i \in \{1,\ldots,M\}$, $j \in \{1,\ldots,N\}$, are, respectively, two encrypted images whose plain images have only 1-bit difference. $D(i,j) = 1$ if $C_1(i,j) \neq C_2(i,j)$; otherwise, $D(i,j) = 0$. $M \times N$ is the size of plain image. With the significance level $\alpha = 0.05$, the theoretical NPCR value is 99.5693% for an image of size 256×256. For the UACI, its ideal interval is (33.2824%, 33.6447%) [1]. The closer to the theoretical value or interval, the stronger the ability to resist the differential attack.

In the proposed analysis, we randomly choose one pixel in six plain images and change its value by 1-bit to generate changed six plain images. They are then encrypted by the proposed encryption scheme and the method presented in [18] to obtain the corresponding encrypted images. Finally, the NPCR and UACI values between these results are listed in Table 11. This indicates that the proposed scheme has qualified performance to resist differential attack in the statistical sense.

4.8. Chosen Plain Image and Known Plain Image Attack Analysis. Security against chosen plain image attack means that the attacker cannot tell which of these two possible plain images were encrypted with probability significantly better than random guessing [50]. In the proposed encryption scheme, we adopt a bit-level circularly shift operation on plain image in the permutation phase. As a result, different encrypted images will be generated by a slightly changed plain image. In addition, chaos-based pseudo-random sequences are further employed to diffuse the permuted image. Therefore, the proposed encryption scheme could successfully resist the chosen plain image and known plain image attack.

4.9. Time Complexity Analysis. The time complexity measures the amount of time taken by an encryption scheme to run as a function of the size of plain image [51–53]. Clearly, the lower the value of running time, the higher the efficiency. However, a complete comparison would be unfair due to the incommensurable difference in terms of the programming skills and the equipment adopted for testing. Nevertheless, in order to evaluate the proposed encryption scheme, Table 12 lists the one round running times on the same platform and the same plain images (5.1.10.tiff (256×256), 5.1.08.tiff (512×512), and 5.3.01.tiff (1024×1024)) for our proposal and the compared algorithms presented in [18, 40–42]. On the other hand, one can learn that the computational cost of the proposed encryption scheme is O(MN) with the big-O notation, where M and N represent the size of the plain image in pixels. It is clear that the proposed scheme can be comparable with the existing algorithms.

4.10. Randomness of the Encrypted Image Analysis. In this section, the National Institute of Standards and Technology (NIST) 800.22 statistical test suite is adopted for the randomness test of the encrypted image generated by the proposed encryption scheme. The test results for plain image 5.1.12.tiff are listed in Table 13. Clearly, each P value corresponding to a particular test is bigger than 0.01 [54]. Hence, the acquired encrypted image can pass all the NIST 800.22 statistical tests at the 1% level of significance, which implies that the encrypted image generated by our proposal possesses qualified randomness.

4.11. Graphic Autocorrelation Analysis. Autocorrelation is the correlation of an image with a delayed copy of itself as a function of delay, which is a mathematical tool for finding repeating patterns in signal processing, such as the presence of a periodic signal obscured by noise [55]. Thus, a secure encryption scheme should generate an encrypted image that presents a flat and uniform graphic autocorrelation to avoid bad effect [53]. Following the way presented in [53], we compute, respectively, the graphic autocorrelations of the plain image 5.1.10.tiff and its encrypted image generated by the proposed encryption scheme, which are graphically shown in Figure 13. As can be seen, the graphic autocorrelation to the plain image shows waves and a cone in the

Table 11: Differential attack analysis (%).

Test image	NPCR		UACI	
	Proposed	Reference [18]	Proposed	Reference [18]
5.1.09.tiff	99.5758	30.4524	33.6181	11.1730
5.1.10.tiff	99.6094	72.0840	33.3383	24.3893
5.1.11.tiff	99.6307	91.1133	33.6026	37.8987
5.1.12.tiff	99.5682	83.5251	33.3514	33.3899
5.1.13.tiff	99.5789	92.4042	33.2812	31.0640
5.1.14.tiff	99.6201	98.5596	33.4738	32.8520
Average	99.5972	78.3564	33.4442	28.4612

Table 12: Time complexity analysis (second).

Image size	Reference [40]	Reference [41]	Reference [18]	Reference [42]	Proposed
256×256	0.0282	0.4762	1.4993	18.4454	1.0196
512×512	0.0437	1.3843	5.9568	73.0198	3.9924
1024×1024	0.1785	5.5640	23.9620	291.6047	16.0123

Table 13: Randomness of the encrypted image analysis.

Test items	P value	Results
The frequency (monobit) test	0.6035	Pass
Frequency test within a block	0.3020	Pass
The runs test	0.3760	Pass
Tests for the longest-run-of-ones in a block	0.6157	Pass
The binary matrix rank test	0.1485	Pass
The discrete Fourier transform (spectral) test	0.5617	Pass
The non-overlapping template matching test	0.9584	Pass
The overlapping template matching test	0.8423	Pass
Maurer's "universal statistical" test	0.1707	Pass
The linear complexity test	0.5492	Pass
The serial test	0.0835	Pass
The approximate entropy test	0.9292	Pass
The cumulative sums (cusums) test	0.9374	Pass
The random excursions test	0.6948	Pass
The random excursions variant test	0.2333	Pass

center, while the encrypted image is uniformly distributed and shows no visible pattern.

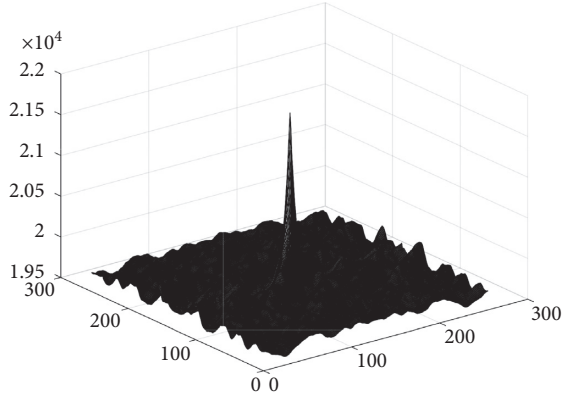
4.12. Encryption Quality Analysis. Encryption quality is the difference between the frequency of occurrence for each pixel gray level before and after encryption [56], which is defined by

$$\mathrm{EQ} = \frac{\sum_{i=0}^{2^L-1} \left| o_i(P) - o_i(C) \right|}{2^L}, \tag{13}$$

where $o_i(C)$ and $o_i(P)$ are, respectively, the observed occurrence for gray level i in the encrypted image C and L bit plain image P. For $L = 8$ bit plain image, the maximal value of EQ is $(510 \times M \times N/2^L \times 2^L)$ where $M \times N$ is the size of plain image. Thus, the closer to the maximal value, the better the encryption quality of the proposed encryption scheme. Table 14 gives the test values for our cipher algorithm and the compared method. It is clear that our proposed encryption scheme has higher cipher quality than the compared method.

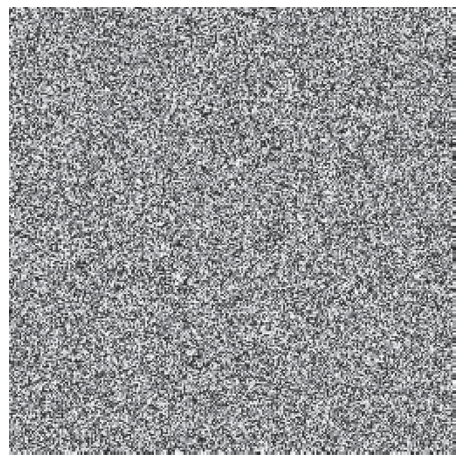
4.13. Comparison with Existing Works. The aforementioned analyses demonstrate the performance features of the proposed encryption scheme. However, when it comes to compare with the existing works, it is not easy to make a fair comparison between two encryption schemes because of the adopted test standard and the purpose used in different real scenarios. Nevertheless, we approximately compute some statistical metrics of the encrypted images generated by the proposed encryption scheme and the compared methods presented in [18, 40–42]. For fairness, all the encryption schemes are performed on the same plain image with the equipment described in Section 3.3. Finally, the computed results are listed in Tables 15 and 16, which show that the proposed encryption scheme is comparable with the existing works. In more detail, Table 15 gives the coefficients

(a)

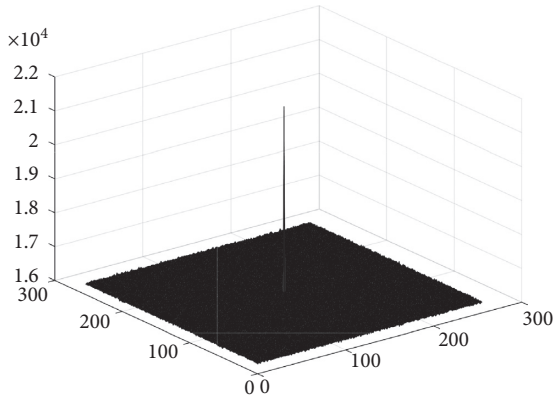


(b)

Figure 13: Continued.



(c)



(d)

Figure 13: Graphic autocorrelation analysis: (a, c) the plain image 5.1.10.tiff and its encrypted image; (b, d) the graphic autocorrelations of (a) and (c).

Table 14: Encryption quality analysis.

File name	Proposed	Reference [18]
5.1.09.tiff	302.0781	300.6094
5.1.10.tiff	188.8203	188.5703
5.1.11.tiff	301.4609	301.1797
5.1.12.tiff	242.6406	243.6875
5.1.13.tiff	455.1406	454.9141
5.1.14.tiff	208.1250	205.8203
Mean	283.0443	282.4635

Table 15: Comparison analysis-I.

Algorithm	Directions		
	Horizontal	Vertical	Diagonal
5.1.10.tiff	0.9012	0.8553	0.8338
Reference [40]	−0.0040	0.0227	0.0050
Reference [41]	−0.0027	−0.0111	0.0290
Reference [18]	−0.0138	−0.0043	−0.0221
Reference [42]	0.0144	0.0057	0.0043
Proposed	0.0251	−0.0228	−0.0098

Table 16: Comparison analysis-II.

Algorithm	NPCR	UACI	Entropy	LSE	χ^2	EQ
5.1.10.tiff	—	—	7.3118	7.0258	—	—
Reference [40]	99.4949	33.5437	7.9972	7.9034	253.8672	189.2266
Reference [41]	99.5682	33.4563	7.9972	7.9041	253.6484	187.7031
Reference [18]	99.2658	33.9685	7.9976	7.9039	217.4219	188.5703
Reference [42]	99.9625	33.5557	7.9973	7.9023	249.4219	188.2422
Proposed	99.6262	33.4770	7.9975	7.9024	227.8750	188.8203

calculated from the plain image 5.1.10.tiff and the different encrypted images generated by compared methods. Similarly, Table 16 lists the NPCR, UACI, entropy values, LSE values, chi-square values, and EQ values for the different schemes.

5. Conclusion

This paper proposes a novel image encryption scheme based on the PWLCM and the standard map, which adopts the well-known permutation-diffusion structure. Different from the traditional pixel-level scrambling way, a hierarchical diffusion process, which not only changes the pixel position but also modifies the pixel value, is employed in the permutation phase. In the pixel diffusion process, the row-by-row and column-by-column operation model is further applied to enhance efficiency. The simulations and performance analyses imply that the proposed scheme may achieve a better trade-off between security and efficiency than other state-of-the-art encryption schemes. Moreover, the proposed encryption scheme provides a good flexibility since it supports any size of plain gray-scale image or color image. More specifically, when it comes to color image, we first reshape the 3-dimensional plain image to the 2-dimensional one from the top to bottom and left to right and then process it with the proposed encryption scheme to generate the final encrypted image. Furthermore, the proposed encryption scheme can achieve more simplicity than other advanced works due to the fact that it only uses the iterative substitution-permutation architecture. In addition, this work can promote the practical application of the nonlinear dynamic system. In the future, to further evaluate the security, we will try to give a rigorously mathematical proof like the conventional cryptography did. On the other hand, to see a higher efficiency, we will implement the proposed encryption scheme using parallel computing.

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

This study was supported in part by the National Natural Science Foundation of China under grant no. 61772147, in part by the Major Basic Research and Cultivation Project of Guangdong Province Natural Science Foundation under grant no. 2015A030308016, in part by the Project of Ordinary University Innovation Team Construction of Guangdong Province under grant no. 2015KCXTD014, in part by the Collaborative Innovation Major Projects of Bureau of Education of Guangzhou City under grant no. 1201610005, and in part by the National Cryptography Development Fund under grant no. MMJJ20170117. The research of Zongxiang Yi was supported by the China Postdoctoral Science Foundation (no. 2019M662834) and Young Innovative Talents Project of General Colleges and Universities in Guangdong Province (no. 2019KQNCX112). Thanks to the graduate overseas joint-training program of Guangzhou University.

References

[1] Z. Hua, Y. Zhou, and H. Huang, "Cosine-transform-based chaotic system for image encryption," *Information Sciences*, vol. 480, pp. 403–419, 2019.

[2] F. Özkaynak, "Brief review on application of nonlinear dynamics in image encryption," *Nonlinear Dynamics*, vol. 92, no. 2, pp. 305–313, 2018.

[3] C. Li, Y. Zhang, and E. Y. Xie, "When an attacker meets a cipher-image in 2018: a year in review," *Journal of Information Security and Applications*, vol. 48, 2019.

[4] J. Fridrich, "Symmetric ciphers based on two-dimensional chaotic maps," *International Journal of Bifurcation and Chaos*, vol. 8, no. 6, pp. 1259–1284, 1998.

[5] E. Solak, C. Çokal, O. T. Yildiz, and T. Biyikoğlu, "Cryptanalysis of fridrich's chaotic image encryption," *International Journal of Bifurcation and Chaos*, vol. 20, no. 5, pp. 1405–1413, 2010.

[6] E. Y. Xie, C. Li, S. Yu, and J. Lü, "On the cryptanalysis of Fridrich's chaotic image encryption scheme," *Signal Processing*, vol. 132, pp. 150–154, 2017.

[7] J. Peng, J. Shangzhu, Y. Yongguo, Y. Zhiming, Y. Mingying, and P. Yangjun, "A novel scheme for image encryption based on piecewise linear chaotic map," *Cybernetics and Intelligent Systems*, vol. 2008, 2008.

[8] H. H. Abdlrudha and Q. Nasir, "Low complexity high security image encryption based on nested PWLCM chaotic map," 2011.

[9] H. Liu, X. Wang, and A. kadir, "Image encryption using DNA complementary rule and chaotic maps," *Applied Soft Computing*, vol. 12, no. 5, pp. 1457–1466, 2012.

[10] X. Wang and C. Jin, "Image encryption using game of life permutation and PWLCM chaotic system," *Optics Communications*, vol. 285, no. 4, pp. 412–417, 2012.

[11] X. Wang and D. Xu, "A novel image encryption scheme based on brownian motion and PWLCM chaotic system," *Nonlinear Dynamics*, vol. 75, no. 1-2, pp. 345–353, 2014.

[12] K. A. K. Patro, A. Soni, P. K. Netam, and B. Acharya, "Multiple grayscale image encryption using cross-coupled chaotic maps," *Journal of Information Security and Applications*, vol. 52, p. 102470, 2020.

[13] W. Cao, Y. Mao, and Y. Zhou, "Designing a 2D infinite collapse map for image encryption," *Signal Processing*, vol. 171, p. 107457, 2020.

[14] V. Patidar, N. K. Pareek, and K. K. Sud, "A new substitution-diffusion based image cipher using chaotic standard and logistic maps," *Communications in Nonlinear Science and Numerical Simulation*, vol. 14, no. 7, pp. 3056–3075, 2009.

[15] V. Patidar, N. K. Pareek, G. Purohit, and K. K. Sud, "A robust and secure chaotic standard map based pseudorandom permutation-substitution scheme for image encryption," *Optics Communications*, vol. 284, no. 19, pp. 4331–4339, 2011.

[16] Y. Zhang and D. Xiao, "Double optical image encryption using discrete Chirikov standard map and chaos-based fractional random transform," *Optics and Lasers in Engineering*, vol. 51, no. 4, pp. 472–480, 2013.

[17] H. Chen, C. Tanougast, Z. Liu, W. Blondel, and B. Hao, "Optical hyperspectral image encryption based on improved Chirikov mapping and gyrator transform," *Optics and Lasers in Engineering*, vol. 107, pp. 62–70, 2018.

[18] J. Chen, L. Chen, L. Y. Zhang, and Z.-l. Zhu, "Medical image cipher using hierarchical diffusion and non-sequential encryption," *Nonlinear Dynamics*, vol. 96, no. 1, pp. 301–322, 2019.

[19] G. Ye and X. Huang, "Spatial image encryption algorithm based on chaotic map and pixel frequency," *Science China Information Sciences*, vol. 61, no. 5, 2018.

[20] K. A. K. Patro and B. Acharya, "A secure block operation based bit-ane image encryption using chaotic maps," 2020.

[21] M. Samiullah, W. Aslam, H. Nazir et al., "An image encryption scheme based on DNA computing and multiple chaotic systems," *IEEE Access*, vol. 8, 2020.

[22] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*, CRC Press, London, UK, 2014.

[23] M. Preishuber, T. Hütter, S. Katzenbeisser, and A. Uhl, "Depreciating motivation and empirical security analysis of chaos-based image and video encryption," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 9, pp. 2137–2150, 2018.

[24] Y. Ma, C. Li, and B. Ou, "Cryptanalysis of an image block encryption algorithm based on chaotic maps," *Journal of Information Security and Applications*, vol. 54, 2020.

[25] J. Chen, L. Chen, and Y. Zhou, "Universal chosen-ciphertext attack for a family of image encryption schemes," *IEEE Transactions on Multimedia*, vol. 54, 2020.

[26] J. Wu, X. Liao, and B. Yang, "Image encryption using 2D Hénon-Sine map and DNA approach," *Signal Processing*, vol. 153, pp. 11–23, 2018.

[27] J. Chen, L. Chen, and Y. Zhou, "Cryptanalysis of a DNA-based image encryption scheme," *Information Sciences*, vol. 520, pp. 130–141, 2020.

[28] F. Dachselt and W. Schwarz, "Chaos and cryptography," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 12, pp. 1498–1509, 2001.

[29] G. Alvarez and S. Li, "Some basic cryptographic requirements for chaos-based cryptosystems," *International Journal of Bifurcation and Chaos*, vol. 16, no. 08, pp. 2129–2151, 2006.

[30] S. Li, G. Chen, and X. Mou, "On the dynamical degradation of digital piecewise linear chaotic maps," *International Journal of Bifurcation and Chaos*, vol. 15, no. 10, pp. 3119–3151, 2005.

[31] R. C. Robinson, "An introduction to dynamical systems: continuous and discrete," *American Mathematical Society*, vol. 19, 2012.

[32] B. V. Chirikov, "Research concerning the theory of noninear resonance and stochasticity," 1971.

[33] A. G. Weber, "The USC-IPI image database: version 5, original release: october 1997, signal and image processing institute, University of Southern California, department of Electrical Engineering," 2014.

[34] R. C. Gonzalez and R. E. Woods, "Image processing, digital image processing," 2007.

[35] N. Tsafack, J. Kengne, B. Abd-El-Atty, A. M. Iliyasu, K. Hirota, and A. A. Abd EL-Latif, "Design and implementation of a simple dynamical 4-D chaotic circuit with applications in image encryption," *Information Sciences*, vol. 515, pp. 191–217, 2020.

[36] A. Hasheminejad and M. J. Rostami, "A novel bit level multiphase algorithm for image encryption based on PWLCM chaotic map," *Optik*, vol. 184, pp. 205–213, 2019.

[37] Y.-Q. Zhang, Y. He, P. Li, and X.-Y. Wang, "A new color image encryption scheme based on 2DNLCML system and genetic operations," *Optics and Lasers in Engineering*, vol. 128, 2020.

[38] Z. Hua, B. Xu, F. Jin, and H. Huang, "Image encryption using josephus problem and filtering diffusion," *IEEE Access*, vol. 7, pp. 8660–8674, 2019.

[39] A. Mansouri and X. Wang, "A novel one-dimensional sine powered chaotic map and its application in a new image encryption scheme," *Information Sciences*, vol. 520, pp. 46–62, 2020.

[40] W. Liu, K. Sun, and C. Zhu, "A fast image encryption algorithm based on chaotic map," *Optics and Lasers in Engineering*, vol. 84, pp. 26–36, 2016.

[41] L. Xu, X. Gou, Z. Li, and J. Li, "A novel chaotic image encryption algorithm using block scrambling and dynamic index based diffusion," *Optics and Lasers in Engineering*, vol. 91, pp. 41–52, 2017.

[42] S. Amina and F. K. Mohamed, "An efficient and secure chaotic cipher algorithm for image content preservation," *Communications in Nonlinear Science and Numerical Simulation*, vol. 60, pp. 12–32, 2018.

[43] J. A. Michel-Macarty, M. A. Murillo-Escobar, R. M. López-Gutiérrez, C. Cruz-Hernández, and L. Cardoza-Avendaño, "Multiuser communication scheme based on binary phase-shift keying and chaos for telemedicine," *Computer Methods and Programs in Biomedicine*, vol. 162, pp. 165–175, 2018.

[44] Y. Zhang, "The fast image encryption algorithm based on lifting scheme and chaos," *Information Sciences*, vol. 520, pp. 177–194, 2020.

[45] M. A. Murillo-Escobar, L. Cardoza-Avendaño, R. M. López-Gutiérrez, and C. Cruz-Hernández, "A double chaotic layer encryption algorithm for clinical signals in telemedicine," *Journal of Medical Systems*, vol. 41, no. 4, p. 59, 2017.

[46] X. Chai, X. Fu, Z. Gan, Y. Lu, and Y. Chen, "A color image cryptosystem based on dynamic DNA encryption and chaos," *Signal Processing*, vol. 155, pp. 44–62, 2019.

[47] I. Yasser, F. Khalifa, M. A. Mohamed, and A. S. Samrah, "A new image encryption scheme based on hybrid chaotic maps," *Complexity*, vol. 5, 2020.

[48] S. T. Welstead, "Fractal and wavelet image compression techniques," *Spie Press*, vol. 40, 1999.

[49] L. Xu, Z. Li, J. Li, and W. Hua, "A novel bit-level image encryption algorithm based on chaotic maps," *Optics and Lasers in Engineering*, vol. 78, pp. 17–25, 2016.

[50] M. A. B. Farah, A. Farah, and T. Farah, "An image encryption scheme based on a new hybrid chaotic map and optimized substitution box," *Nonlinear Dynamics*, vol. 99, no. 4, p. 3041, 2019.

[51] M. A. Murillo-Escobar, C. Cruz-Hernández, F. Abundiz-Pérez, R. M. López-Gutiérrez, and O. R. Acosta Del Campo, "A RGB image encryption algorithm based on total plain image characteristics and chaos," *Signal Processing*, vol. 109, pp. 119–131, 2015.

[52] W. Song, Y. Zheng, C. Fu, and P. Shan, "A novel batch image encryption algorithm using parallel computing," *Information Sciences*, vol. 518, pp. 211–224, 2020.

[53] M. A. Murillo-Escobar, M. O. Meranza-Castillón, R. M. López-Gutiérrez, and C. Cruz-Hernández, "Suggested integral analysis for chaos-based image cryptosystems," *Entropy*, vol. 21, no. 8, p. 815, 2019.

[54] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," 2001.

[55] R. P. Heilbronner, "The autocorrelation function: an image processing tool for fabric analysis," *Tectonophysics*, vol. 212, no. 3-4, pp. 351–370, 1992.

[56] M. Farajallah, "Chaosased crypto and joint cryptoompression systems for images and videos," 2015.