Hindawi
Complexity
Volume 2021, Article ID 5499538, 19 pages
https://doi.org/10.1155/2021/5499538

Research Article

Analytic Study of a Novel Color Image Encryption Method Based on the Chaos System and Color Codes

Shamsa Kanwal,[1] **Saba Inam**,[1] **Omar Cheikhrouhou**,[2] **Kinza Mahnoor**,[1] **Atef Zaguia**,[3] **and Habib Hamam**[4,5]

[1]*Deparment of Mathematical Sciences, Faculty of Science and Technology, Fatima Jinnah Women University, The Mall, Rawalpindi, Pakistan*
[2]*CES Laboratory, ENIS, University of Sfax, Sfax 3038, Tunisia*
[3]*College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia*
[4]*Faculty of Engineering, Uni de Moncton, Moncton, Canada*
[5]*International Institute of Technology (IIT), Sfax, Tunisia*

Correspondence should be addressed to Omar Cheikhrouhou; omar.cheikhrouhou@isetsf.rnu.tn

Received 6 April 2021; Revised 2 May 2021; Accepted 25 May 2021; Published 3 June 2021

Academic Editor: Jorge-Antonio Lopez-Renteria

Due to the growing of the use of Internet and communication media, image encryption is rapidly increased. Image sharing through unsafe open channels is vulnerable for attacking and stealing. For protecting the images from attacks, encryption techniques are required. Recently, new and efficient chaos-based techniques have been suggested to develop secure image encryption. This study presents a novel image encryption framework based on integrating the chaotic maps and color codes. Three phases are involved in the proposed image encryption technique. Piecewise chaotic linear map (PWLCM) is used in the first phase for permuting the digital image. In the second phase, substitution is done using Hill cipher which is the mixing of color codes with the permuted image. The third phase is implemented by XORing, a sequence generated by the chaotic logistic map (CLM). The proposed approach enhances the diffusion ability of the image encryption making the encrypted images resistant to the statistical differential attacks. The results of several analyses such as information entropy, histogram correlation of adjacent pixels, unified average changing intensity (UACI), number of pixel change rate (NPCR), and peak signal-to-noise ratio (PSNR) guarantee the security and robustness of the proposed algorithm. The measurements show that the proposed algorithm is a noble overall solution for image encryption. Thorough comparison with other image encryption algorithms is also carried out.

1. Introduction

Images are a substantial source of information not limited to the daily routine of a common person, but having diverse applications in various fields of military, medical, and industry. For example, we may enumerate military image records, trusted video conferencing, satellite imagery, planetary motion images, and keeping a person's medical record [1]. The requirements of consistent, fast, and robust techniques to store and transmit digital images have led to the development of novel encryption techniques. The information conveyed through images is very complex as compared to simple text. Data sent through open channels such as Internet can be illegally accessed and restored. Therefore, the progress in the field of image encryption creates diverse opportunities and applications in upcoming future. Several assessment criteria including the information entropy, correlation between adjacent pixels, peak signal-to-noise ratio (PSNR), the number of pixels change rate (NPCR), and unified average changing intensity (UACI) related to the image encryption are essential for performance evaluation of the encryption algorithms. The algorithm for which the values of these criteria fulfill the standard expectation level can resist the statistical and differential

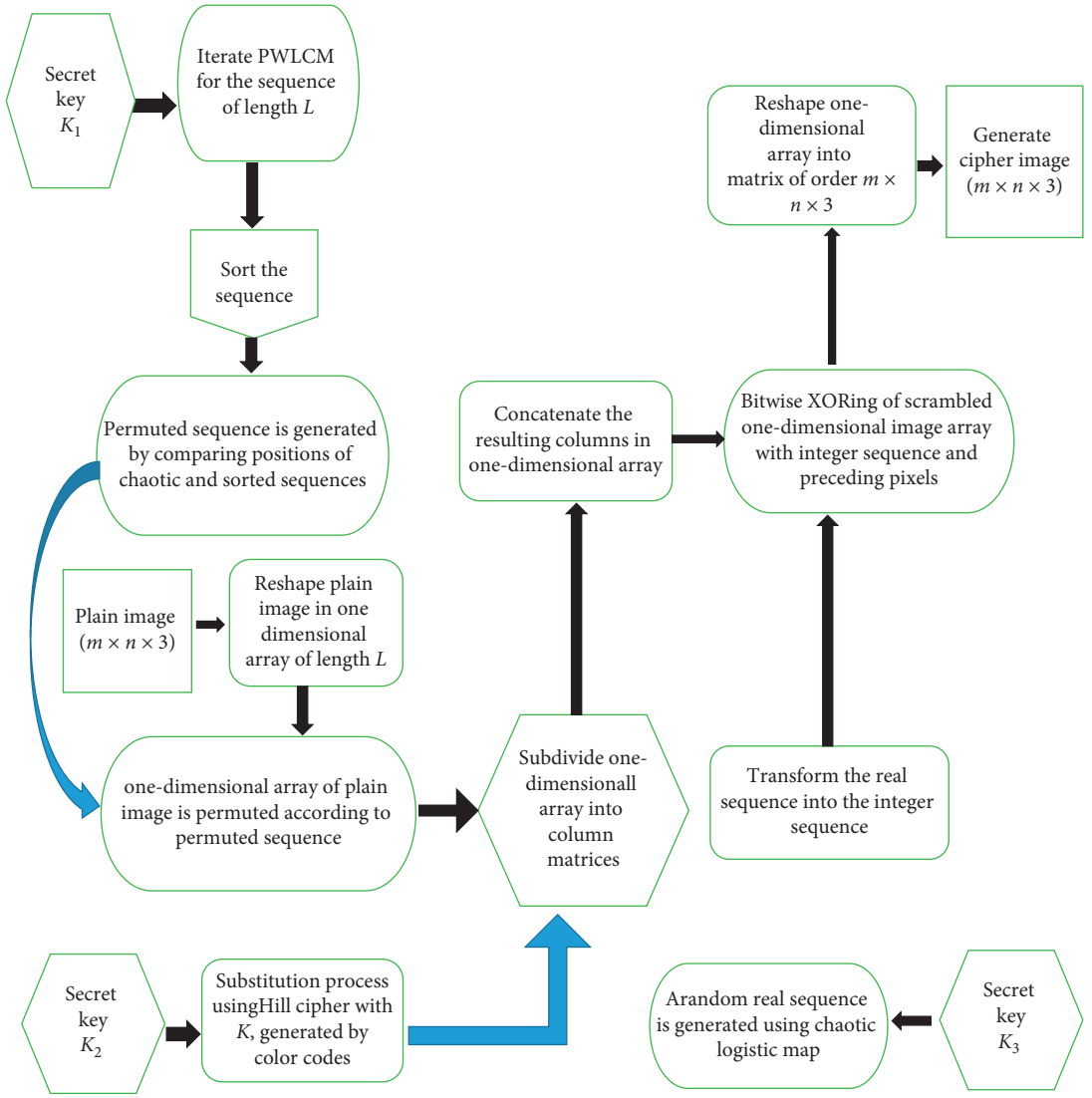


Figure 1: Flowchart of the proposed image encryption algorithm.

Input. Color image I, secret key $K_1 = (\xi_0, \eta)$, PWLCM (1)
Output. Image array PM with scrambled pixels
Step 1. One-dimensional array$P$ of size $L = m \times n \times 3$ is created by reshaping the original image matrix I to one-dimensional array, where m, n are the number of rows and columns, respectively, of the original image matrix I
Step 2. Using PWLCM (1) with the keyK_1, generate the chaotic sequence $X = \{x_1, x_2, ..., x_L\}$ and sort the resulting sequence $\overline{X} = \{\overline{x}_1, \overline{x}_2, ..., \overline{x}_L\}$ in ascending order
Step 3. Compute the position vector of X in $\overline{X}$ and note down the transformed positions TRAN $= \{p_1, p_2, ..., p_L\}$
Step 4. The array P is permuted using TRAN to get PM

Algorithm 1: (Pixel permutation).

attacks [2]. Moreover, for resisting the brute-force attacks, an algorithm with large key space and sensitive to initial conditions is recommended.

Imaging technology meets chaos and propagation requirements compared with traditional encryption systems; chaotic systems [3] have powerful features, such as nonperiodicity, nonlinearity, unpredictability, and extreme sensitivity to initial conditions [4]. Matthews [5] introduced the concept of chaotic function in cryptography. He suggested that a random sequence can be generated by iterating a nonlinear function with certain conditions. In 1998, [6] Friedrich first applied the chaotic system to image encryption. Since then, image encryption based on chaotic systems has gradually become the main field of cryptography

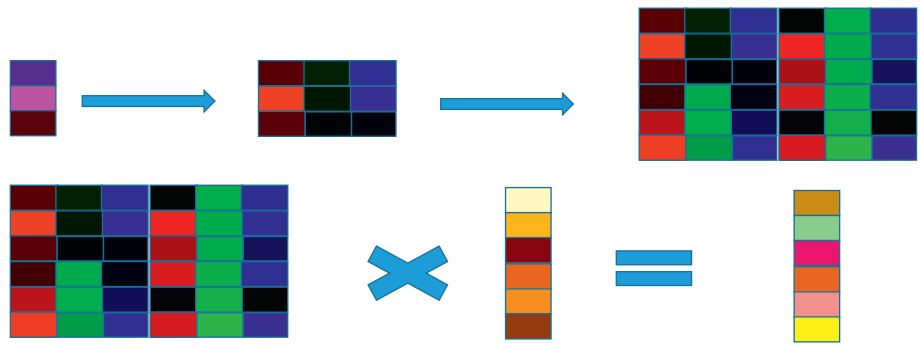


Figure 2: Schematic representation of key mixing with color codes.

Input. Permuted array PM, $K_2 = (\text{color } 1, \text{color } 2, \text{color } 3)$, where color $1 = (R_1, G_1, B_1)$,color $2 = (R_2, G_2, B_2)$, and color $3 = (R_3, G_3, B_3)$ are any three random secret colors in (R, G, B) format and k is any random integer, such that $\gcd(k, 256) = 1$.
Output. An array Q of order L
Step 1. Computing self-invertible matrix
(a) Make a matrix K_{11} (mod256) of order 3×3 as $K_{11} = \begin{bmatrix} R_1 & G_1 & B_1 \\ R_2 & G_2 & B_2 \\ R_3 & G_3 & B_3 \end{bmatrix}$
(b) Take a random integer $k \in (1, 256)$, such that $\gcd(k, 256) = 1$
(c) Calculate $K_{12} = k(I_3 - K_{11}) \text{mod} 256$, $K_{21} = k^{-1}(I_3 + K_{11}) \text{mod} 256$, and $K_{22} = -K_{11} \text{mod} 256$, where I_3 is the identity matrix.
(d) Form a 6×6 self-invertible matrix K_p as $K_p = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix}$
Step 2. Making submatrices M_i
(a) Convert one-dimensional array PM into submatrices of order6×1. The i^{th} matrix is M_i, where $i = 1, 2, \ldots, (L/6)$.
(b) Key mixing is performed using the subsequent formula of Hill cipher $C_i = K_p \times M_i \, (\text{mod} 256)$.
(c) Concatenate all the C_i's in the form of one-dimensional array again as $Q = \{C_1 \| C_2 \| \ldots \| C_{(L/6)}\}$.

Algorithm 2: (Key mixing with color codes).

Input. An arrayQ, secret key $K_3 = (\phi_0, \beta)$, CLM (2).
Output. Encrypted image CI
Step 1. With key K_3 and CLM (2), generate a sequence $R = \{r_1, r_2, \ldots, r_L\}$
Step 2. The sequence R is transformed into a sequence of integers using the following formula: $\text{DF} = \text{floor}(\text{mod}(R_i \times 10^{14}, 256))$.
Step 3. Bitwise XOR each element of Q with element of DF at the corresponding positions and preceding ciphered pixel as $C_i = \text{DF}_i \oplus Q_i \oplus C_{i-1}$, $i = 1, 2, \ldots, L$.
Step 4. Reshape array C in the form of a matrix CI of order $L = m \times n \times 3$
Step 5. Convert resulting matrix in step (4) to get the cipher image

Algorithm 3: (Pixel diffusion).

[7]. Chen and Mao used chaotic 3D cat maps [8] and Baker maps [9] to create permuted image in their proposals. Guan used a chaotic 2D cat map [10] to swap pixels in 2005. Patidar et al. [11] presented image encryption scheme based on substitution-diffusion using chaotic standard map and chaotic logistic maps.

In 2014, [12] Zhang and Wang proposed a new multi-image encryption algorithm based on mixed pixels and piecewise linear chaotic mapping. It is the fastest way to solve the problem. Many researchers have designed image encryption techniques by using various combinations of chaotic maps such as logistic map and Baker map [13], tent and logistic map [14], and the logistic-sine-coupling map [15]. The security and efficiency of algorithms is improved by these suggestions. Liao et al. [16] recently implemented a shorthand strategy based on the enlarged channel model's probability. He also used critical functions and pixel correlation functions [17] for stenographic purpose.

Chaos system plays a vital role in the different fields of mathematics. Many complicated systems can be investigating through chaos systems. Chaotic maps have very interesting features such as sensitivity to the initial value: a completely different sequence is generated with the small change in the initial value. Other features may include nonperiodicity, the map which is used to generate the chaotic sequence is nonperiodic, and randomness behavior, the chaotic sequences which are generated by the chaotic map are mostly pseudorandom sequences with complex

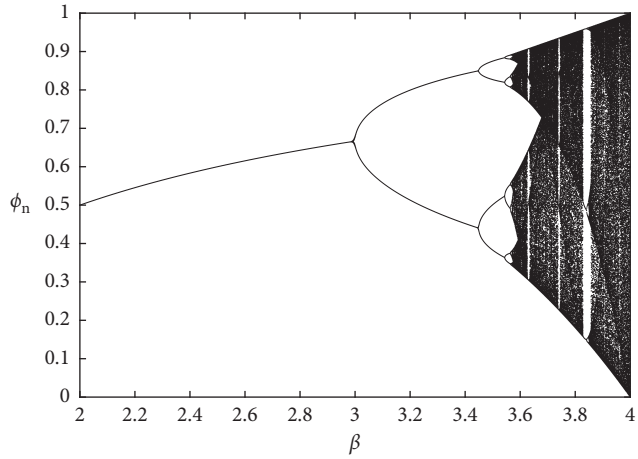


Figure 3: Bifurcation diagram of CLM.

Input. Encrypted imageCI, secret keys K_1, K_2, K_3, PWLCM (1), CLM (2).
Output. Plain color image I
Step 1. The encrypted image matrix CI is placed in an array of size $L = m \times n \times 3$
Step 2. As in step 1 and step 2 in Algorithm 3, the receiver generates a sequence R of size L by secret key K_3 and CLM (2)
Step 3. Each element of CI in step 2 is passing through the following formula:
$D_j = \mathrm{CI}_j \oplus \mathrm{DF}_j \oplus D_{j-1}$, $j = 1, 2, \ldots, L$.
Step 4. By using key K_2, receiver generates matrix K_p as in Algorithm 2, which is self-invertible matrix
Step 5.Convert one-dimensional array D into submatrices $\mathrm{DM}_j$of order 6×1
Step 6. Key mixing is reversed by using the formula
$B_j = K_p \times \mathrm{DM}_j \,(\mathrm{mod}256)$, $j = 1, 2, \ldots, L$.
Step 7. Rewrite all B_j's in the form of one-dimensional array DQ
Step 8. By iterating the PWLCM and using the shared secret key K_1, get a sequence X and get $\overline{X}$ by sorting X in ascending order
Step 9. The permutation array is computed by inverse transform position $(\mathrm{TRAN})^{-1}$
Step 10. Use $(\mathrm{TRAN})^{-1}$ on DQ to get P
Step 11. Reshape P in a matrix form of order $L = m \times n \times 3$ and converted to image I

Algorithm 4: (Image decryption).

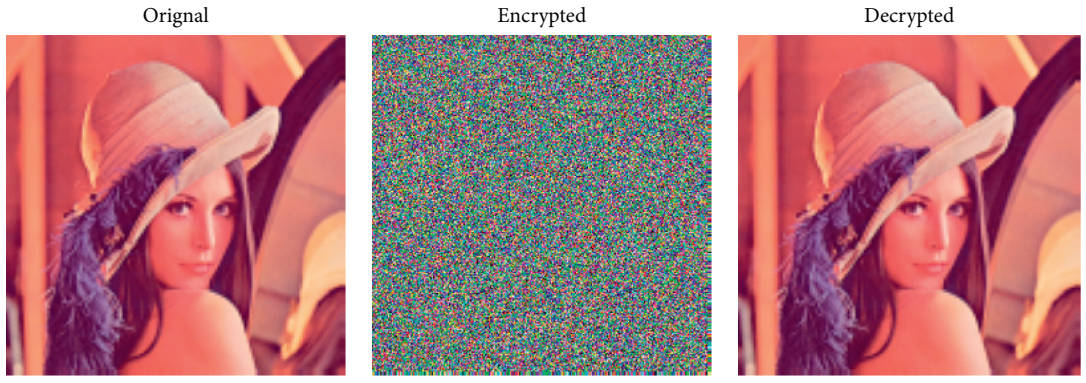


Figure 4: Sample Lena (colored 256×256 pixels). (a) Original image. (b) Encrypted image. (c) Decrypted image.

structures. Due to these features, security of image encryption can be improved because without knowing the correct values of control parameters and initial conditions, an attacker cannot predict the chaos map. These features of chaotic maps enable them to be highly recommended for creating the confusion and diffusion in image encryption. For instance, see references [18–23].

The present study is inspired by the above cited investigations and their applications to different areas. The core goal of this work is to make advanced venture in the regime

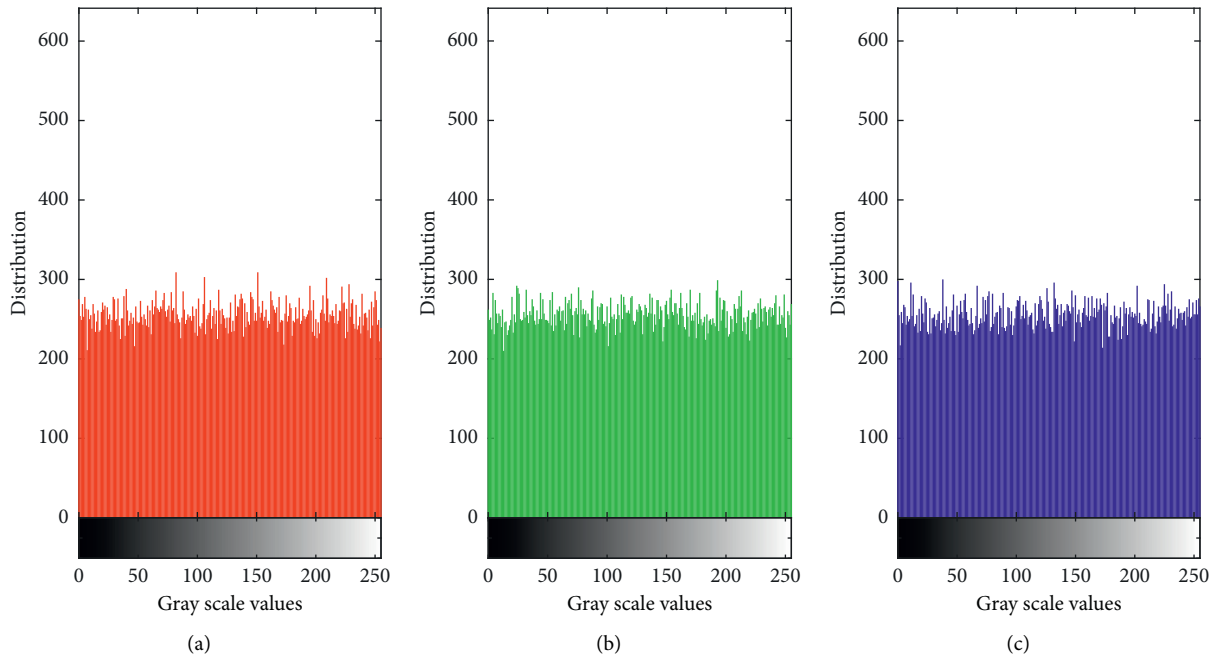


Figure 5: Histogram analysis of encrypted image of Lena (colored 256×256 pixels). (a) Red component. (b) Green component. (c) Blue component.

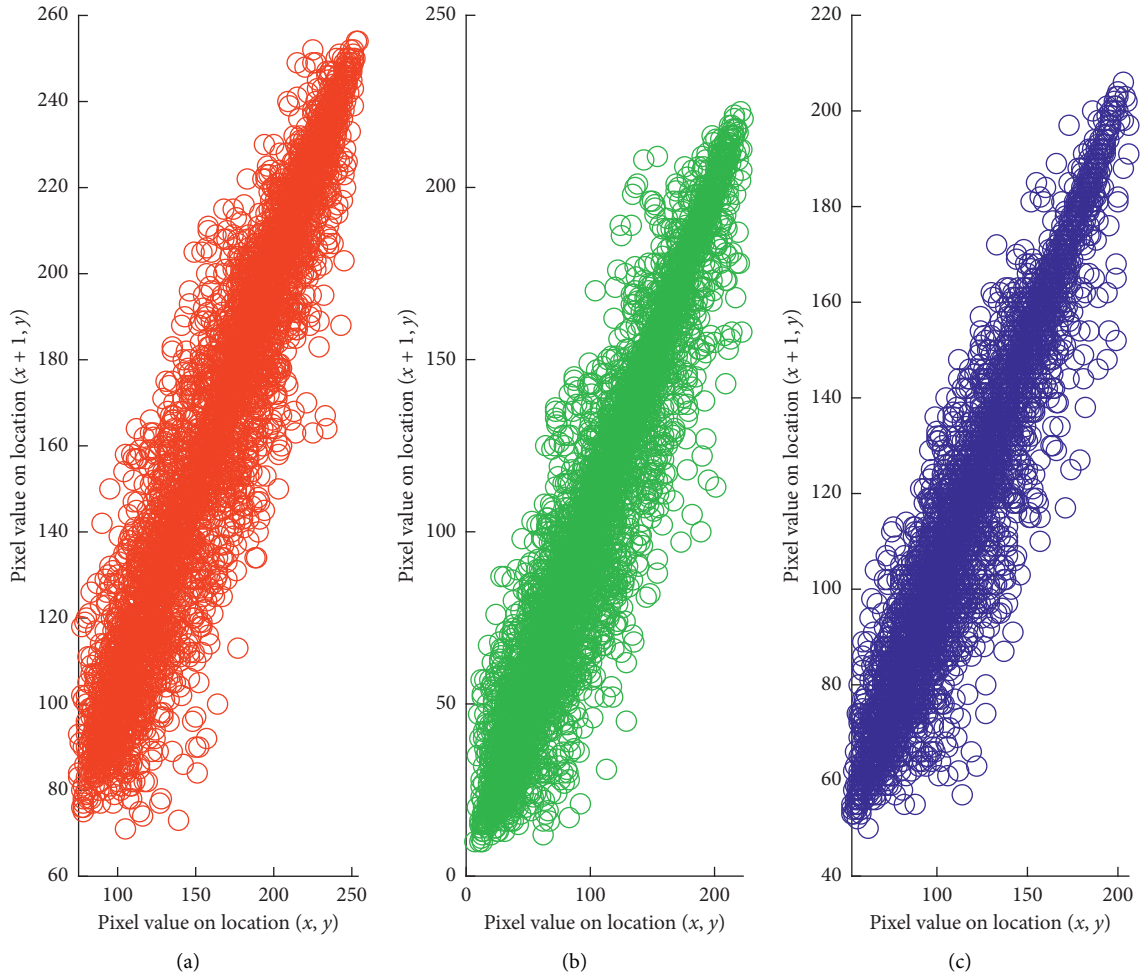


Figure 6: Correlation (row wise) of original image of Lena. (a) Red component. (b) Green component. (c) Blue component.

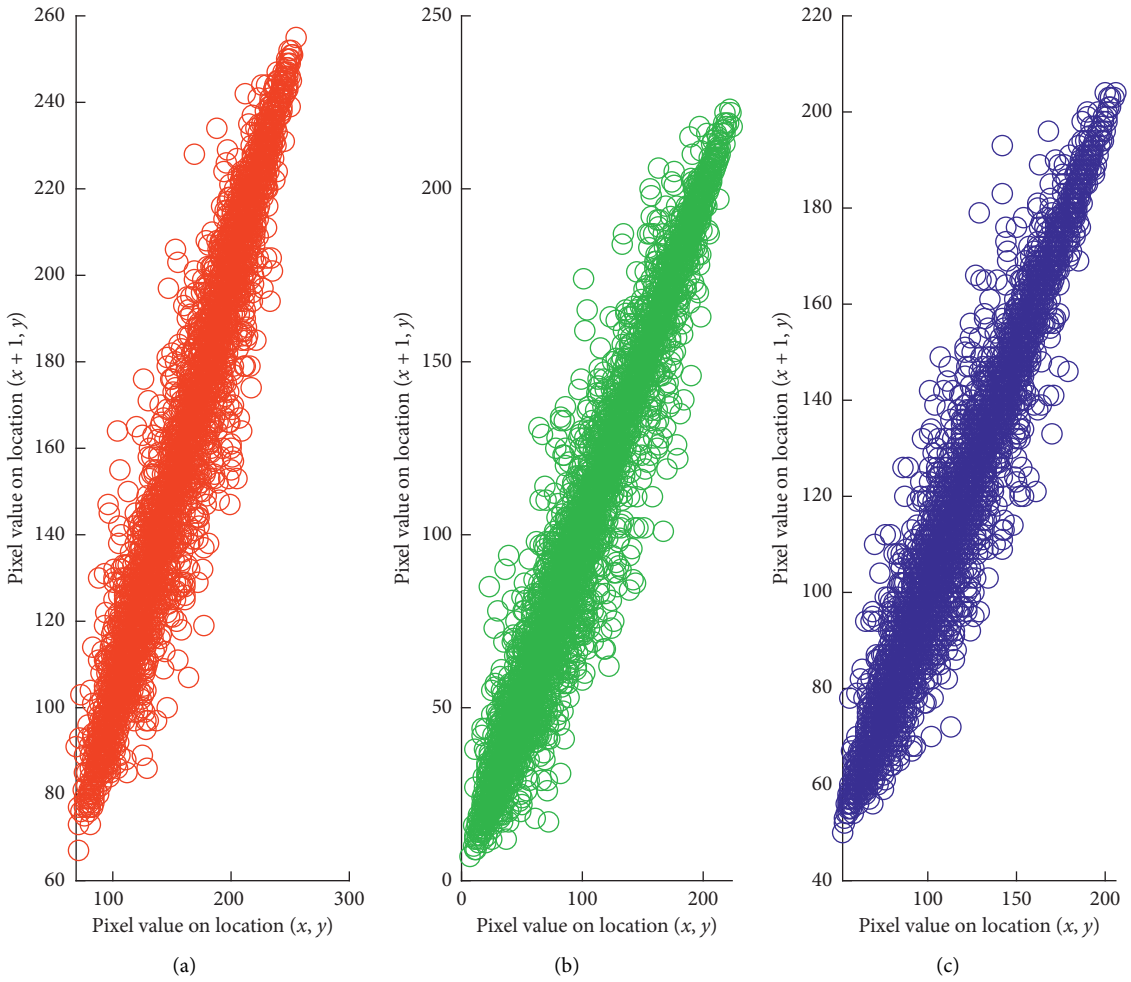


Figure 7: Correlation (column wise) of original image of Lena. (a) Red component. (b) Green component. (c) Blue component.

of image encryption using chaotic maps. More accurately, this manuscript deals with developing and analyzing a novel image encryption that comprises three phases: pixel permutation process, substitution process, and pixel diffusion process. The permutation sequence for the first phase is generated by PWLCM, and the pixels of the plain image are then permuted according to the permutation sequence. Instead of using S-boxes for substitution phase, the substitution of pixels in the permuted image is determined by Hill cipher whose key is generated by color codes. The same key is used in the decryption process because it is self-invertible. At the end, the diffusion process is completed by CLM to ensure the secrecy of the entire image encryption technique. The effectiveness of the proposal is shown by several experimental results. By using information entropy analysis along with other indicative parameters such as entropy, PSNR, UACI, NPCR, and correlation factors, the proposed image encryption technique is compared with some existing techniques.

The remaining study is outlined as follows. The proposed image encryption algorithm is given in Section 2. In Section 3, we present the decryption process. Section 4 is based on the details of implementation results generated by executing the encryption and decryption algorithm to some test images. Section 5 consists of assessments of the algorithm in different aspects. Section 6 concludes the presented work.

2. The Proposed Image Encryption Algorithm

To develop an algorithm, following three aspects should be considered:

(1) The evaluation and implementation of the algorithm must be simple and easy

(2) The design of the encryption algorithm must resist the known attacks

(3) For the algorithms, the concepts and basic ideas must be well established and reliable

Keeping in mind all the three aspects, an efficient and secure technique for image encryption is proposed here, using the chaotic logistic map and color codes.

For the image selection, the size of $m \times n \times 3$ pixels image is recommended for its encryption. The original image is processed into one-dimensional array for encryption, but the encrypted image is again of the size $m \times n \times 3$.

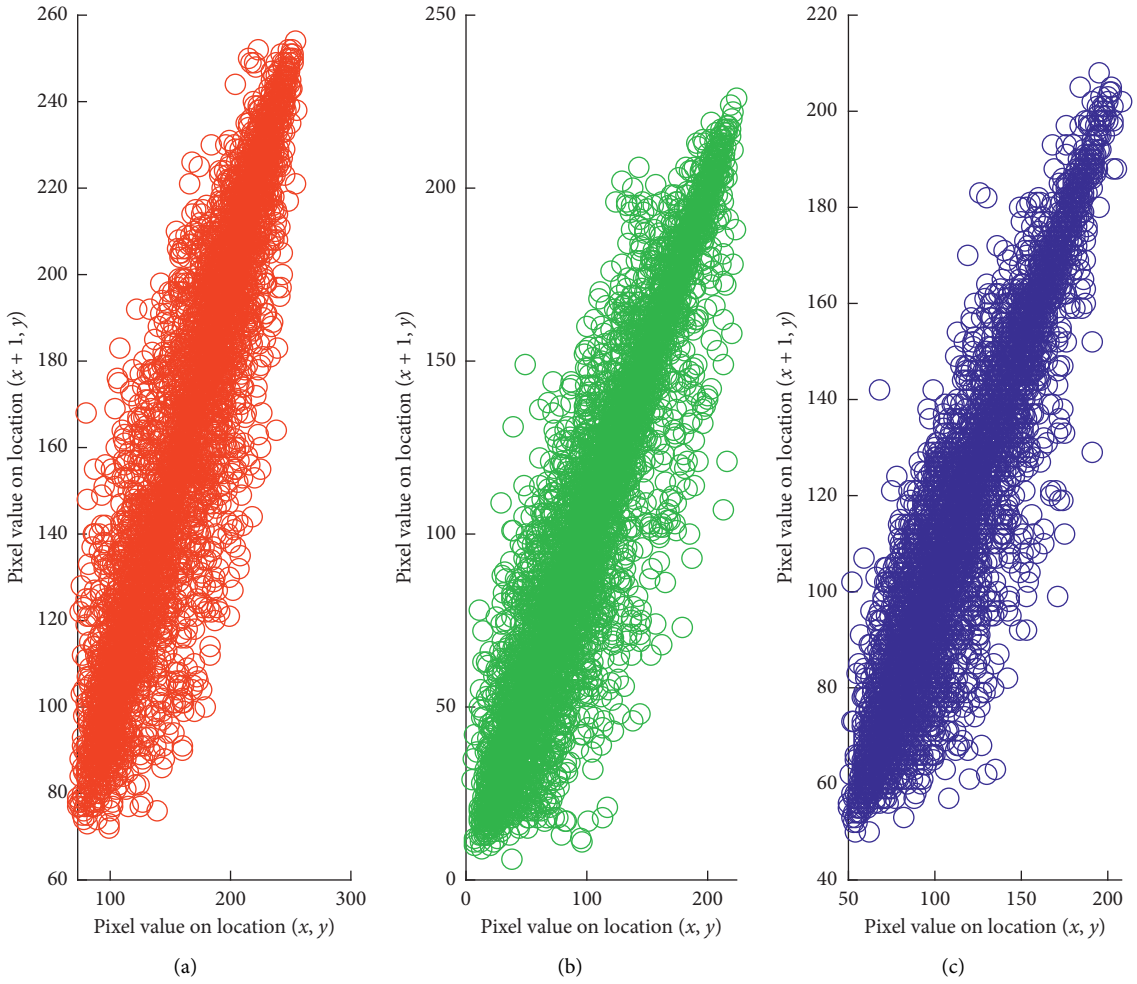


Figure 8: Correlation (diagonal wise wise) of original image of Lena. (a) Red component. (b) Green component. (c) Blue component.

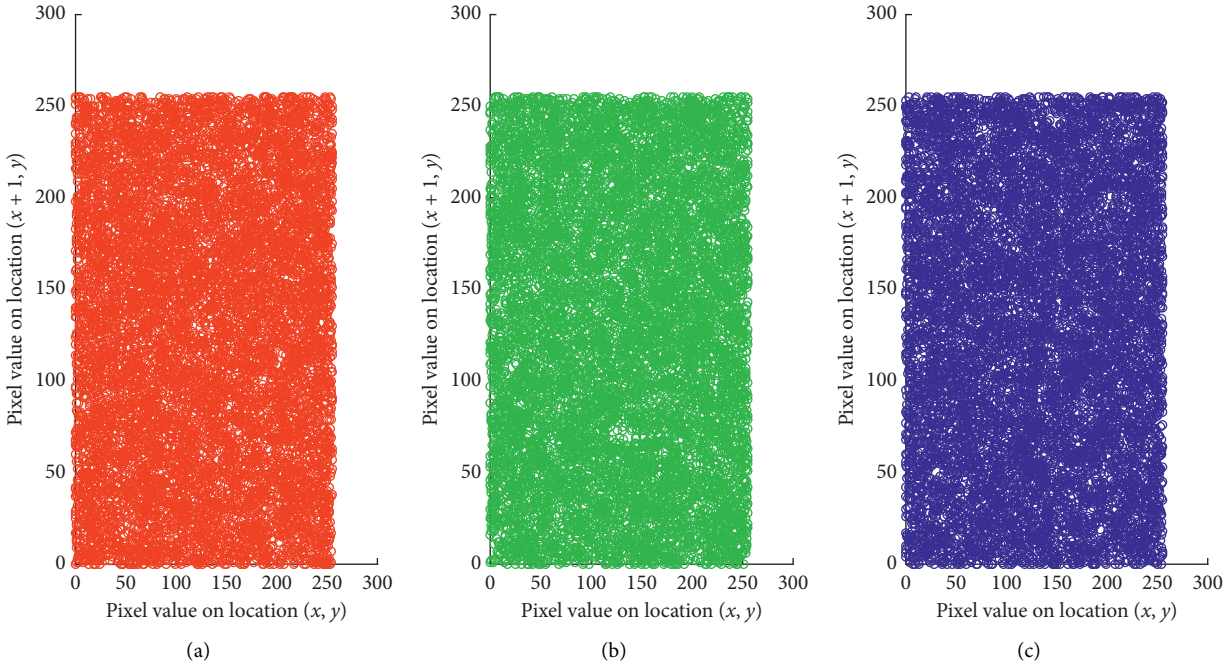


Figure 9: Correlation (row wise) of encrypted image of Lena. (a) Red component. (b) Green component. (c) Blue component.

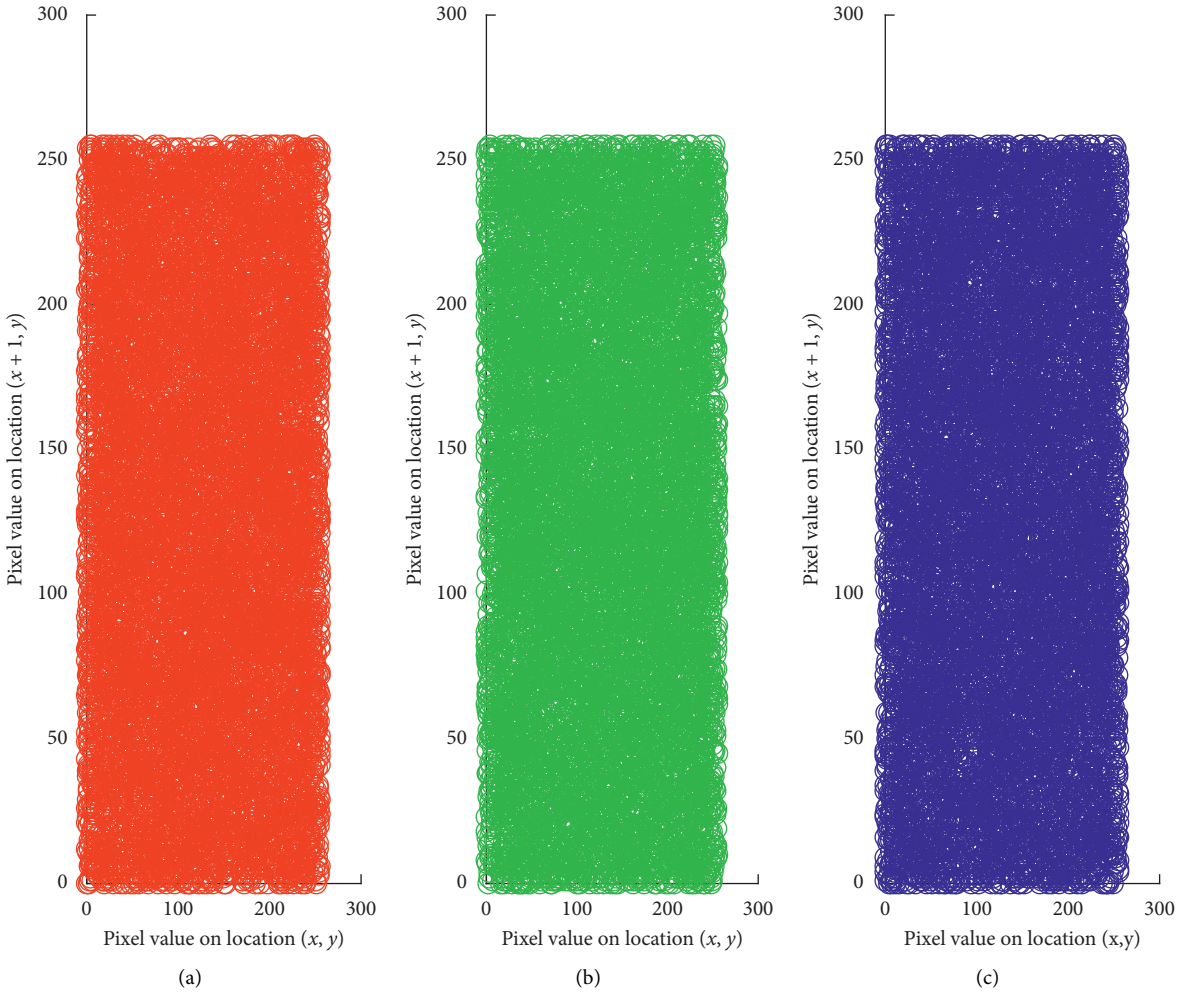


Figure 10: Correlation (column wise) of encrypted image of Lena. (a) Red component. (b) Green component. (c) Blue component.

There are three phases involved in encryption; pixel permutation, substitution process using Hill cipher with color codes, and pixel diffusion. In the first phase, the piecewise linear chaotic map is used for permuting the pixels, so that the statistical structure of the plain image is dissipated into long-range statistics of the cipher image. The permuted image is then mixed with a self-invertible key matrix generated by secret color codes, in the second phase. Finally, confusion is achieved by XORing with another chaotic map to make the relationship between the statistics of the cipher image and the value of the key as complex as possible to thwart attempts of cryptanalyst. The designed flowchart shown in Figure 1 summarizes our proposed encryption algorithm.

2.1. Permutation Process. Three keys K_1, K_2, and K_3 are used in three phases, respectively, of our proposed encryption algorithm. The first phase changes the position of pixels of the original image I. The piecewise chaotic linear map is used to permute the pixels. Using K_1, iterate the piecewise chaotic linear map (PWLCM) to get a chaotic sequence and sort the obtained chaotic sequence in ascending order. By comparing the positions of the chaotic sequence and sorted sequence, obtain the permutation sequence. This permutation sequence is used to permute the one-dimensional array of the plain image.

2.1.1. Piecewise Linear Chaotic Map (PWLCM). There are many different ways to generate the chaotic sequences or the piecewise chaotic maps for the encryption. The authors of [24] proposed hyperchaotic encryption based on multiscroll piecewise linear systems. The manuscript [25] describes maximal unstable dissipative interval to preserve multiscroll attractors via multisaturated functions.

The piecewise linear chaotic map is defined [26] as

$$\xi_{n+1} = f(\xi_n, \eta) = \begin{cases} \dfrac{\xi_n}{\eta}, & \text{if } 0 \le \xi_n \le \eta, \\ \dfrac{(\xi_n - \eta)}{(0.5 - \eta)}, & \text{if } \eta < \xi_n \le 0.5, \\ 1 - \xi_n, & \text{if } 0.5 < \xi_n < 1, \end{cases} \quad (1)$$

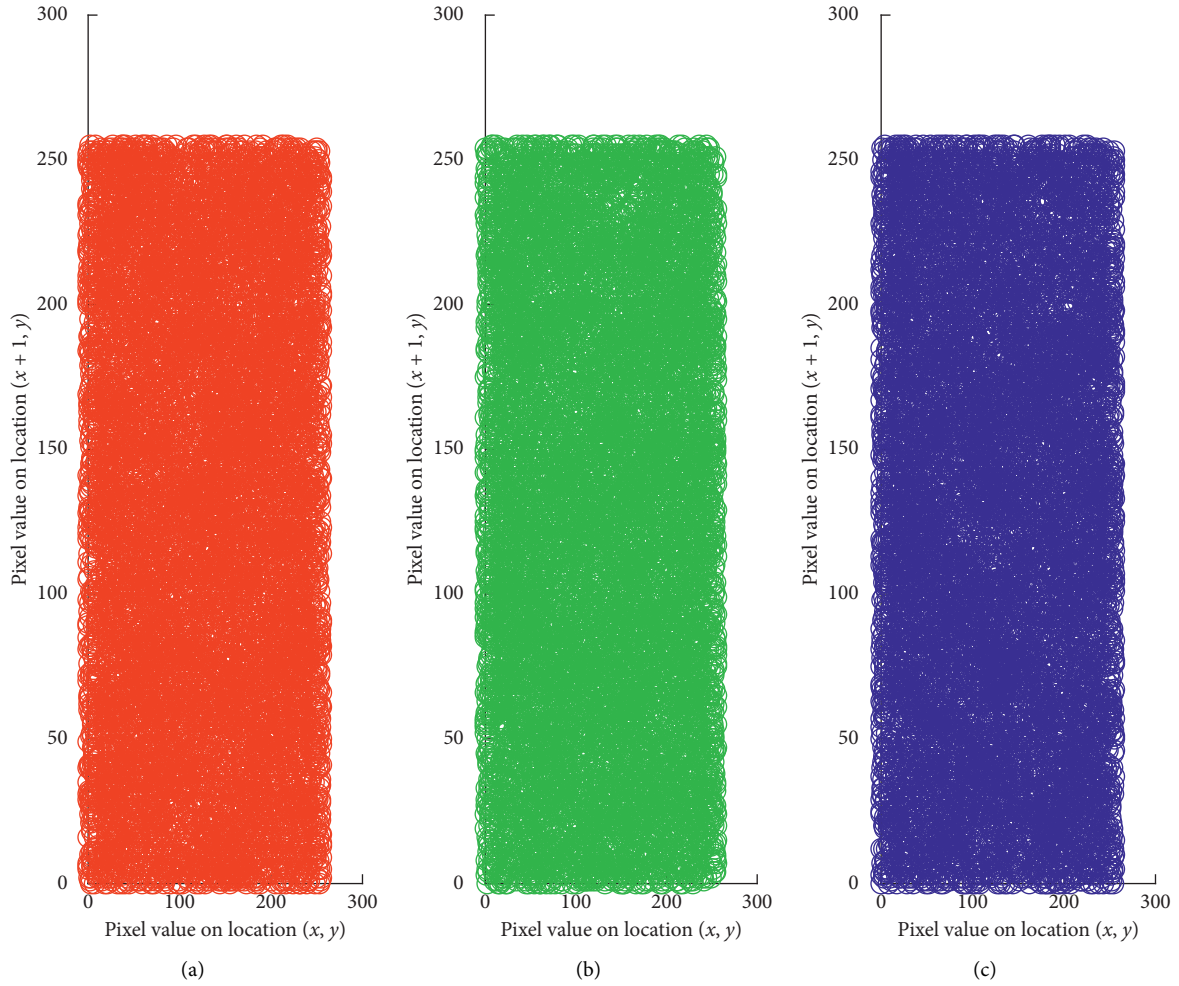


Figure 11: Correlation (diagonal wise) of encrypted image of Lena. (a) Red component. (b) Green component. (c) Blue component.

Table 1: Lena (colored 256 × 256 pixels) image correlation coefficient values.

Direction	Red		Green		Blue	
	Original	Cipher	Original	Cipher	Original	Cipher
Horizontal	0.9910	0.0046	0.9889	0.0005	0.9846	0.0084
Vertical	0.9781	0.0009	0.9741	0.0028	0.9709	−0.0032
Diagonal	0.9648	−0.0012	0.9613	0.0030	0.9563	−0.0022

Table 2: The comparison of values of information entropy.

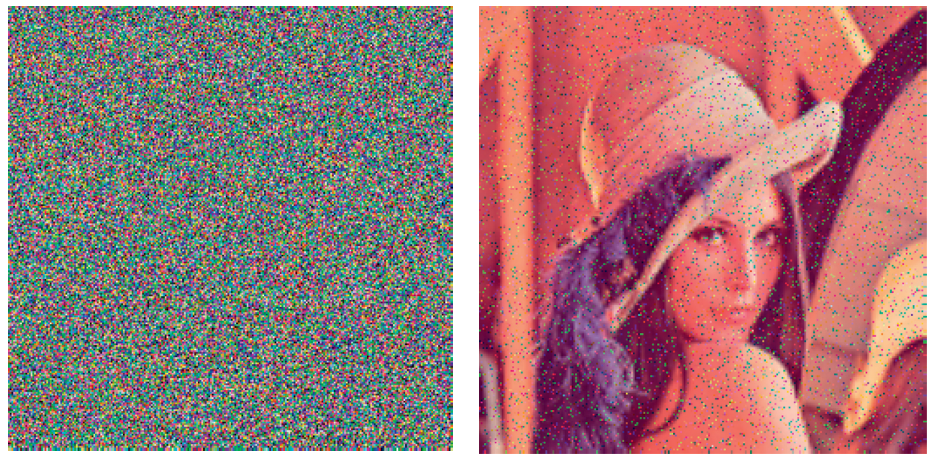
Image encryption algorithm	Entropy values
Reference [30]	7.9967
Reference [31]	7.9970
Proposed algorithm	7.9990

Table 3: Comparison of NCPR and UACI values.

Image encryption algorithm	NPCR	UACI
Reference [30]	99.61	33.46
Reference [31]	99.22	33.40
Reference [32]	99.61	33.41
Proposed algorithm	99.61	33.46

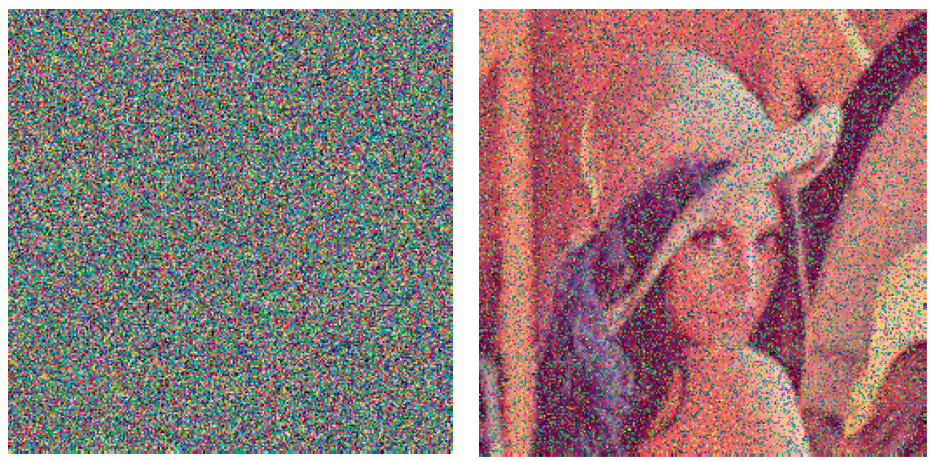
Table 4: Estimate of critical values of NPCR and UACI.

Image encryption algorithm	Obtained value	NPCR test results		
		0.05 level	0.01 level	0.001 level
			Theoretical NPCR values	
		99.5693%	99.5527%	99.5341%
	99.61%	Pass	Pass	Pass
			UACI test results	
Proposed algorithm		0.05 level	0.01 level	0.001 level
			Theoretical UACI values	
		33.2824–33.6447%	33.2255–33.7016%	33.1594–33.7677%
Proposed algorithm	33.46%	Pass	Pass	Pass



(a) (b)

Figure 12: Experimental results for the performance evaluation of data loss attacks. (a), (b) Cipher images and decryption result of corresponding images using our algorithm with 1% salt and pepper noise.



(a) (b)

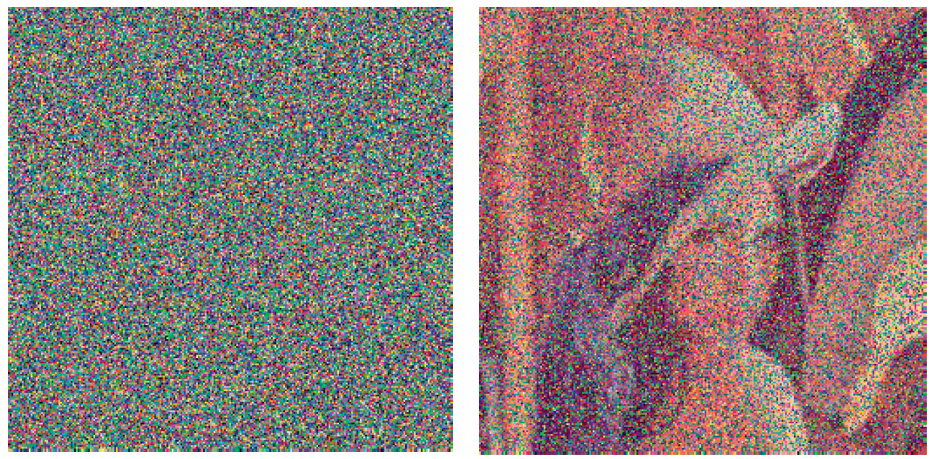
Figure 13: Experimental results for the performance evaluation of data loss attacks. (a), (b) Cipher images and decryption result of corresponding images using our algorithm with 5% salt and pepper noise.

has many dynamic properties, for example, Lyapunov exponent, random-like behavior, and uniform unvarying density function. For these attributes, PWLCM is highly recommended for cryptographic purposes. The conditions and parameters of PWLCM are as follows:

(1) $\xi_0 \in [0, 1)$, where ξ_0 is the initial value

(2) $\eta \in (0, 0.5)$, where η is the control parameter

(3) $K_1 = (\xi_0, \eta)$, where K_1 is the secret key of the permutation process

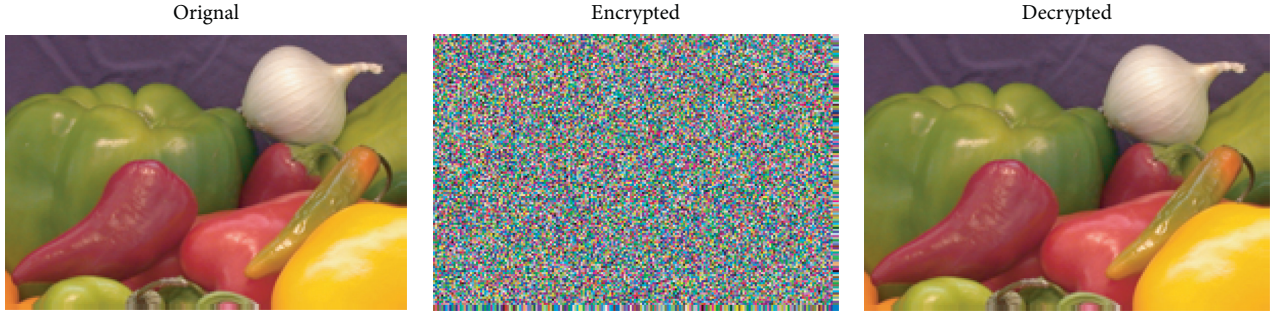


(a) (b)

Figure 14: Experimental results for the performance evaluation of data loss attacks. (a), (b) Cipher images and decryption result of corresponding images using our algorithm with 10% salt and pepper noise.

Table 5: Performance of MSE and PSNR.

Salt and pepper noise (%)	MSE	PSNR
1	8716.6	8.7273
5	8815.9	8.8263
10	8926.2	9.6253



(a) (b) (c)

Figure 15: Sample onion (colored 198 × 135 pixels). (a) Original image. (b) Encrypted image. (c) Decrypted image.

The following Algorithm 1 describes the permutation process.

2.2. Substitution Process Using Hill Cipher and Color Codes. The Hill cipher [27, 28] is a polygraphic block cipher invented by Lester S. Hill in 1929. It serves a significant role in cryptography because of its simplicity, high speed, high throughput, and resistance against frequency analysis attack.

The Hill cipher method requires an invertible key matrix, so that the decryption can be allowed. To overcome the difficulty of having an invertible key matrix, self-invertible matrix is introduced by Acharya et al. [29]. The substitution process is carried out by employing Hill cipher which uses the self-invertible key matrix based on color codes, making the substitution phase simple and efficient.

RGB color format is a model that adds red, blue, and green colors in different quantities and produces new colors. Total bits that each color uses are 8, and hence, they can have any integer value from 0 to 255. There are $256 \times 256 \times 256 = 16777216$ possibilities of generating different colors. Any three colors, color1, color2, and color3, from these possible colors can be selected as our second secret keyK_2. Now, $K_2$is used to generate a self-invertible matrix of order 6×6. The permuted image array PM is divided into $(L/6)$ submatrices of order 6×1. These submatrices are multiplied one by one withK_2. The resulting matrices are combined once again to make a one-dimensional array Q. Figure 2 shows the schematic representation of key mixing

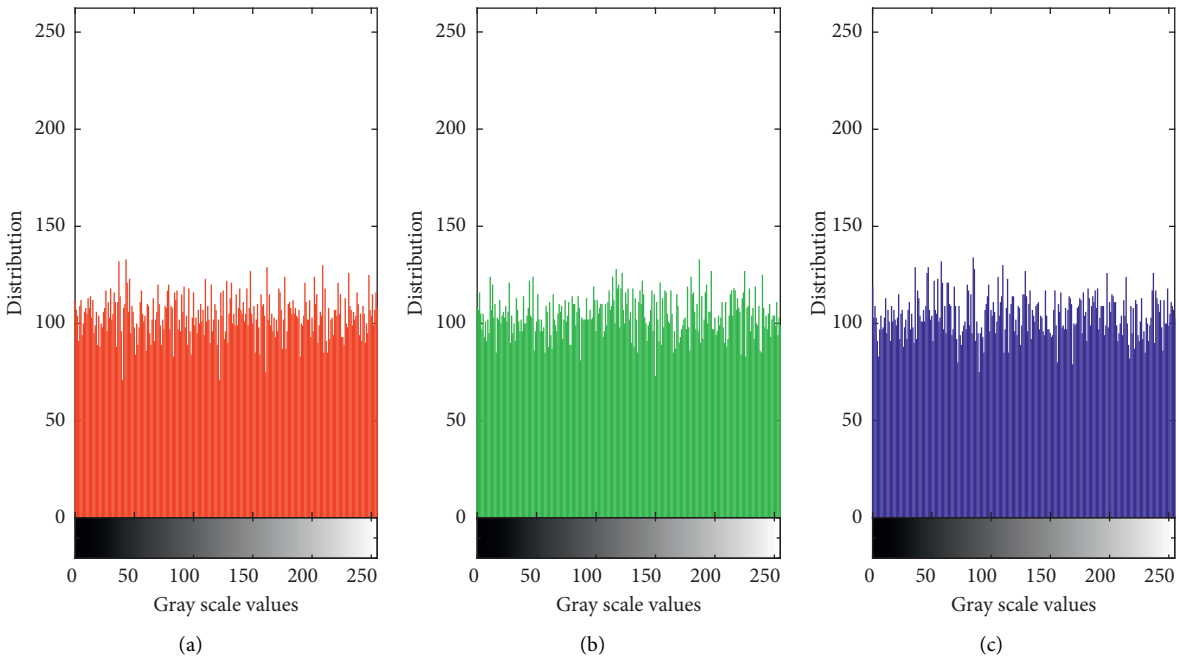


Figure 16: Histogram of cipher image of onion (colored 198 × 135 pixels). (a) Red component. (b) Green component. (c) Blue component.

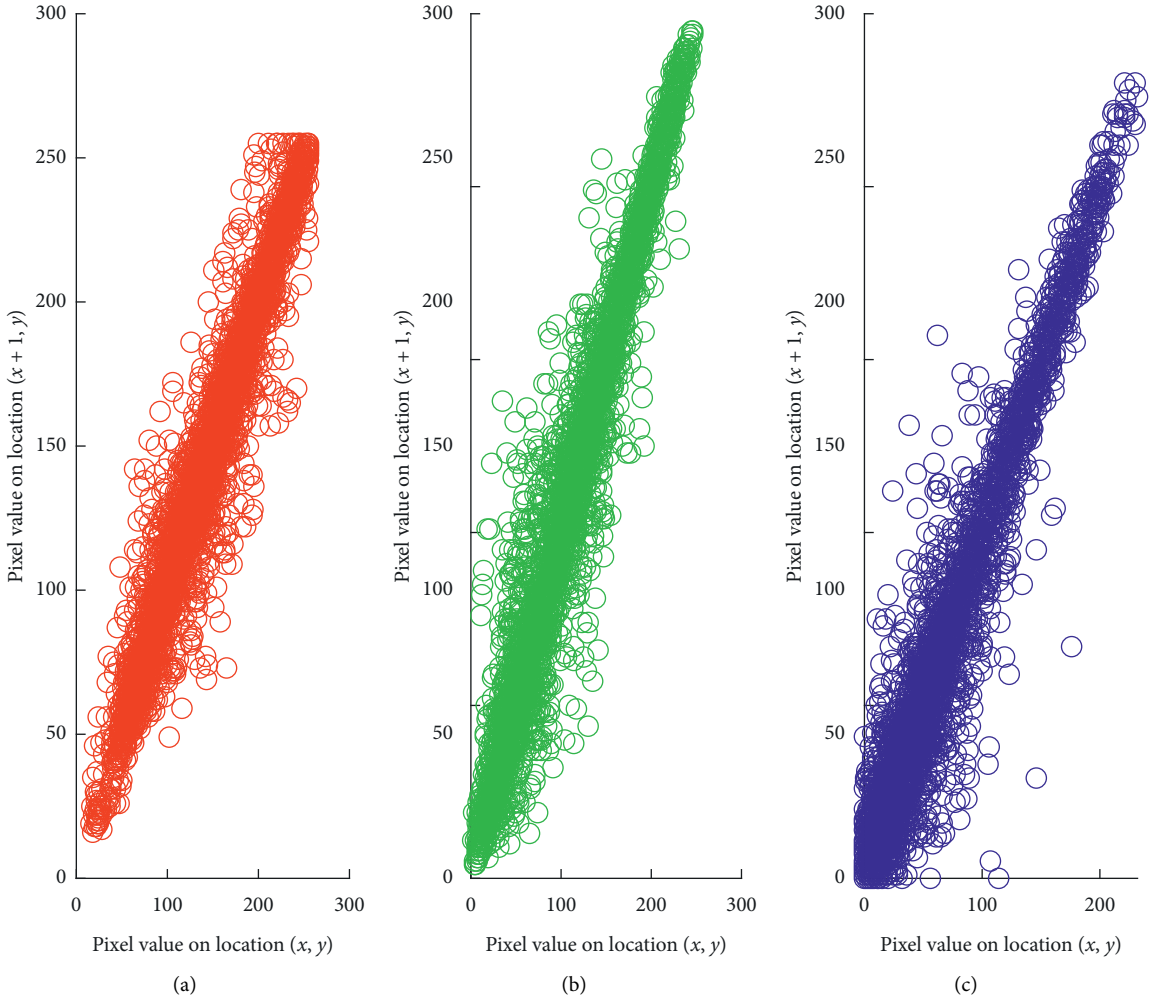


Figure 17: Correlation (row wise) plot of plain onion image. (a) Red component. (b) Green component. (c) Blue component.

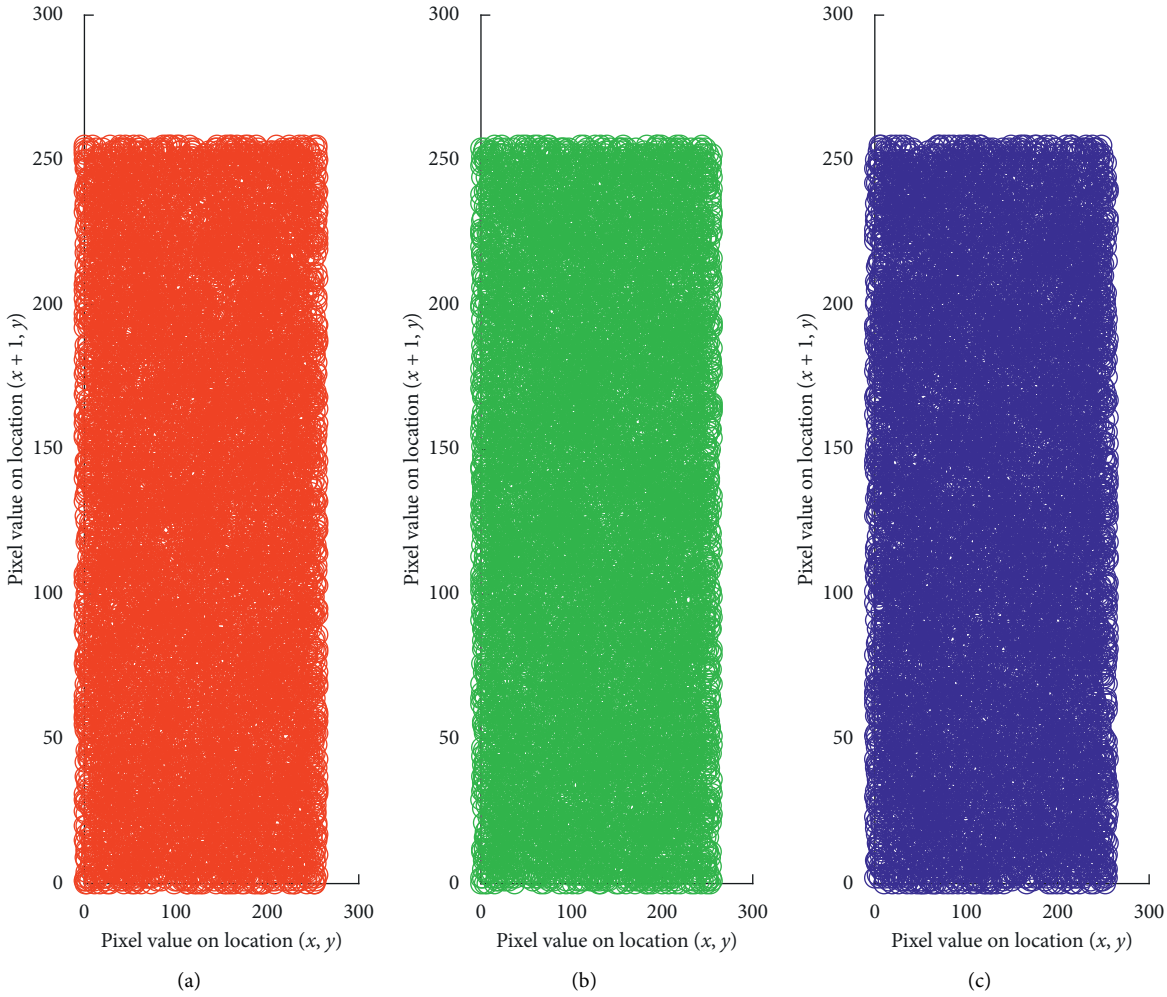


Figure 18: Correlation (row wise) plot of color components of onion cipher image. (a) Red component. (b) Green component. (c) Blue component.

with color codes. The Algorithm 2 describes the substitution process.

2.3. Pixel Diffusion Process. In the final phase, using key K_3, a sequence of real numbers, is generated by iterating CLM (2) and converted into integer's sequence using Algorithm 3. To create diffusion, one-dimensional array Q is bitwise XORed with the integer sequence. The resulting one-dimensional array is reshaped as a matrix of order $m \times n \times 3$ again, and cipher image is generated by this matrix.

2.3.1. Chaotic Logistic Map (CLM). The final phase is a combination of a chaotic logistic map and XOR operation to apply the diffusion of pixels. Due to this change of pixel value, the pixels of the cipher image drastically change with even small one bit change in the plain image. For this process, we generate a random sequence using CLM which is defined as follows:

$$\phi_{n+1} = g(\phi_n) = \beta\phi_n(1 - \phi_n). \quad (2)$$

The conditions and parameters of CLM are defined as

(1) $\phi_0 \in (0, 1)$, where ϕ_0 is the initial state of the system

(2) $\beta \in (0, 4)$, where β is the bifurcation control parameter

The chaotic behavior of the CLM with infinite period is shown in Figure 3.

The following Algorithm 3 describes the diffusion process.

3. Image Decryption Process

The following image decryption algorithm is used to revert back to the encryption algorithm for getting the original image. The decryption process also comprises three stages. In the first stage, the XOR operation is eradicated with the sequence generated with key K_3. The effect of color mixing is wiped out by multiplying with the self-invertible key matrix generated by key K_2. Finally, a random sequence and ultimately the inverse of permutation is constructed using key K_1. To reverse the

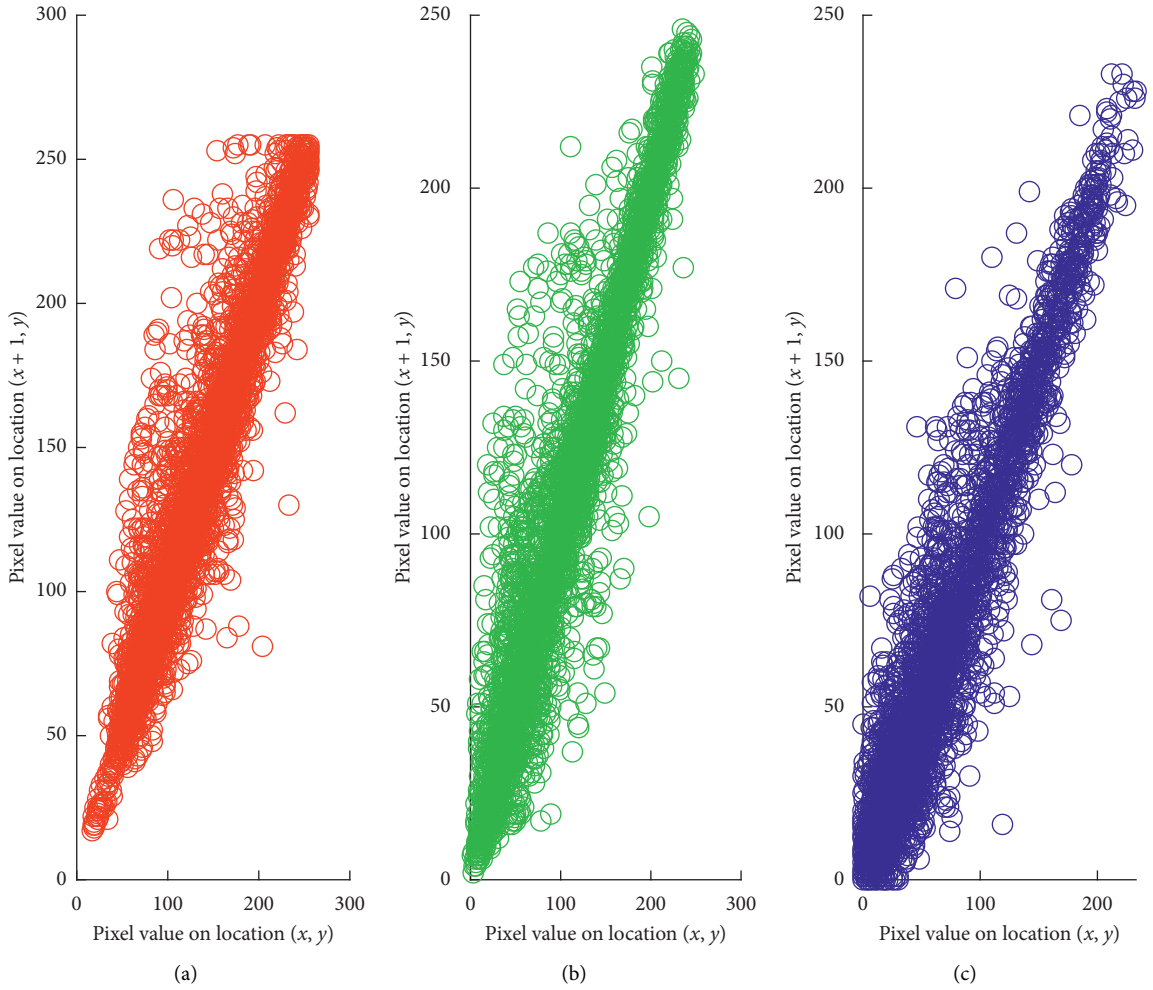


Figure 19: Correlation (column wise) of original image of onion. (a) Red component. (b) Green component. (c) Blue component.

permutation, the inverse permutation is used. The original image is obtained by transforming the subsequent array into image form.

The following Algorithm 4 describes the decryption process.

4. Implementation of Proposed Algorithms

For the evaluation of the proposed scheme, we used Matlab 2018a. The algorithms of pixel permutation, key mixing using Hill cipher with color codes, and pixel diffusion are executed to get the encrypted image and decryption algorithm to again get the plain image back. The standard colored images of Lena with (256×256) pixels are taken for the testing of our proposal. We perform the encryption using $K_1 = (0.766, 0.3432)$, $K_2 =$ (purple haze, bright neon pink, fire brick, 123), and $K_3 = (0.7666, 3.999)$. For comparison purpose, we take image of Lena to compare our results with many other schemes present in the literature. The sample input and output of Lena image by proposed algorithms is shown in Figure 4. The proposed algorithm takes 12.41 seconds to encrypt the Lena image.

5. Results, Analysis, and Performance Evaluation

In this section, proposed algorithm is evaluated by analyzing the statistical and differential parameters. We have developed the guidelines, both generally and specifically to compare the algorithm with different techniques. For performing correct encryption and decryption, these guidelines should be followed when choosing certain parameters involved in the algorithms.

5.1. Statistical Histogram Analysis. Figure 5 shows the histogram of red, green, and blue channels of the cipher image. It is clearly observed that the histogram of the cipher image is fairly uniform. It is evident that no information is leaked from the cipher image of the dispersal of pixels in the original image.

5.2. Correlation Analysis of Adjacent Pixels. The confusion and diffusion can be tested by using correlation analysis between neighboring pixels in the original image and the

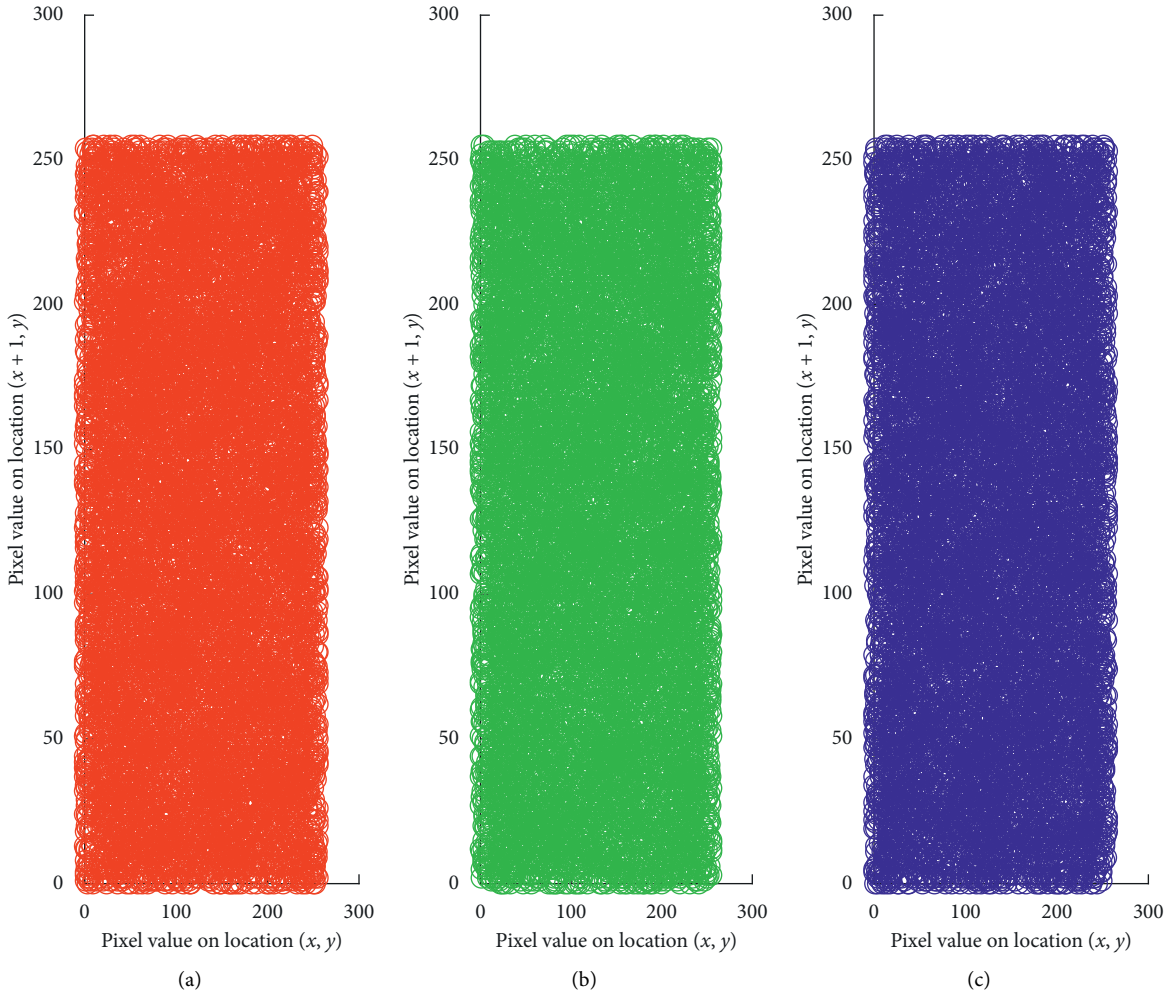


Figure 20: Correlation (column wise) of encrypted image of onion. (a) Red component. (b) Green component. (c) Blue component.

corresponding encrypted image. The correlation is calculated by using the following formula:

$$C_r = \frac{n\left(\sum_{t=1}^{n} x_t y_t - \sum_{t=1}^{n} x_t \sum_{t=1}^{n} y_t\right)}{\left(n\sum_{t=1}^{n}(x_t)^2 - \left(\sum_{t=1}^{n} x_t\right)^2\right)\left(n\sum_{t=1}^{n}(y_t)^2 - \left(\sum_{t=1}^{n} y_t\right)^2\right)}, \tag{3}$$

where x_t and y_t are the values of two neighboring pixels and n is the total number of pixels taken for calculating correlation. The highest value of correlation coefficient equals 1 and shows that the adjacent pixels are having high correlation. So, our encryption algorithm must encrypt the image with correlation coefficients very small and near to zero, so that the cryptanalyst cannot get any valuable information. Figures 6–8 display the correlation of the original image pixels in row, column, and diagonal directions, respectively. Figures 9–11 show the correlation of the cipher image pixels in row, column, and diagonal directions, respectively. Table 1 gives the values of correlation of scattering pixels in the horizontal, vertical, and diagonal directions for the plain and cipher image. The value obtained from equation (3) for cipher image is close to zero which shows that adjacent pixels in cipher image are almost uncorrelated.

5.3. Information Entropy Analysis. Entropy is a measurement of unpredictability of the pixel concentrations in the encrypted image. For an 8 bit image, the encryption algorithm with a value of the entropy close to 8 is considered as a good algorithm. It is calculated by the following equation:

$$H(C) = \sum_{i=0}^{2^N-1} P(C_i)\log_2\frac{1}{P(C_i)}, \tag{4}$$

where C be a ciphered image and $P(C_i)$ is the probability of character C_i in encrypted image. For the security of the image encryption algorithm, it should be least possible to predict the original image from the encrypted image. With the entropy value 8, there are less chances of predicting plain image from cipher image. Using Matlab R2018a, the entropy value of encrypted image obtained from the proposed encryption turns out to be 7.9990. Table 2 gives a brief comparison of obtained information entropy value with various image encryption algorithms. The resulting value depicts that entropy of the proposed encryption is close enough to the ideal value 8. It guarantees that there is no loss of information.

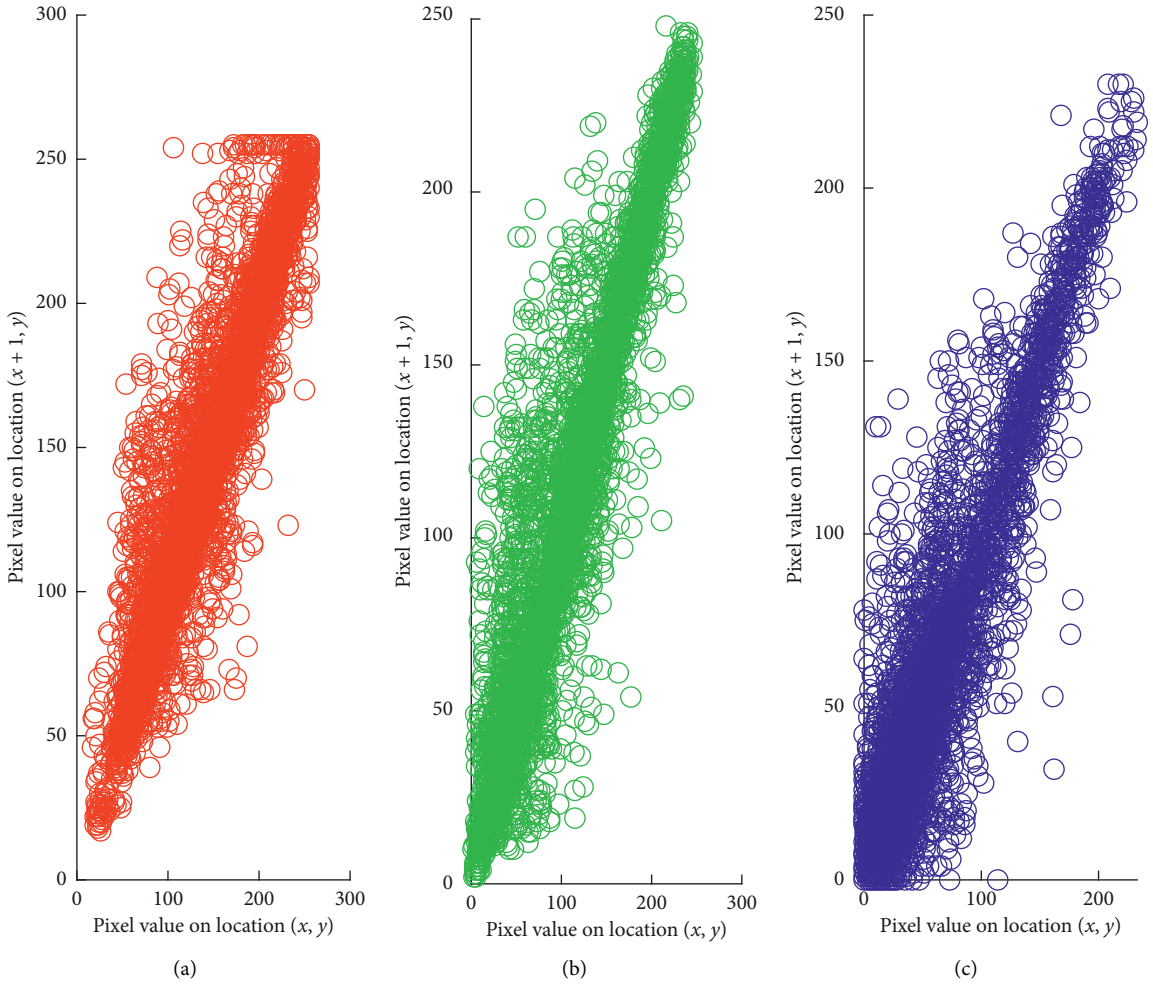


Figure 21: Correlation (diagonal wise) of original image of onion. (a) Red component. (b) Green component. (c) Blue component.

5.4. Sensitivity Analysis of the Proposed Algorithm. The net pixel change rate (NPCR) and unified average changing intensity (UACI) are two measuring criteria used for investigating the effect of altering one pixel of the plain image on the cipher image. Both indicators are defined by the following formulas, respectively:

$$\text{NPCR} = \frac{\sum_{i,j} K(i,j)}{w \times h} \times 100, \tag{5}$$

$$\text{UACI} = \frac{1}{w \times h}\left[\sum_{i,j} \frac{\left|X(i,j) - X'(i,j)\right|}{255}\right] \times 100, \tag{6}$$

where w and h show the width and height of ciphered image, respectively. X represents cipher image corresponding to plain image, while X' represents the cipher image corresponding to plain image with change of one pixel, respectively. If $X(i,j) \neq X'(i,j)$, then $K(i,j) = 1$; else, $K(i,j) = 0$.

The NPCR and UACI measures indicate the resistance of the algorithm against differential attacks, such as a ciphertext-only attack, a plaintext attack, or a known plaintext attack. The higher values of NPCR and UACI give the best security measures. The comparison of the NPCR and UACI values of encrypted Lena image is given in Table 3. The estimate of critical values of NPCR and UACI of proposed scheme is given in Table 4.

5.5. Mean Square Error Analysis. In the cipher image of test image Lena, we add 1%, 5%, and 10% salt and pepper noise as shown in Figures 12(a), 13(a), and 14(a) , respectively. The corresponding decrypted images of noised cipher images are shown in Figures 12(b), 13(b), and 14(b), respectively. From these figures, it is evident that when the cipher image bear salt and pepper noise or data loss attacks, the decrypted image preserves huge majority of original image information having only a small portion of uniformly distributed noise.

The mean square error (MSE) is the measurement of difference between the original and cipher images. The high value of MSE is related to a high difference between original image and cipher image. It can be calculated by the following equation:

$$\text{MSE} = \frac{1}{m \times n \times 3} \sum_{i=0}^{m-1} \sum_{i=0}^{n-1} \left(I_P(i,j) - I_D(i,j)\right)^2, \tag{7}$$

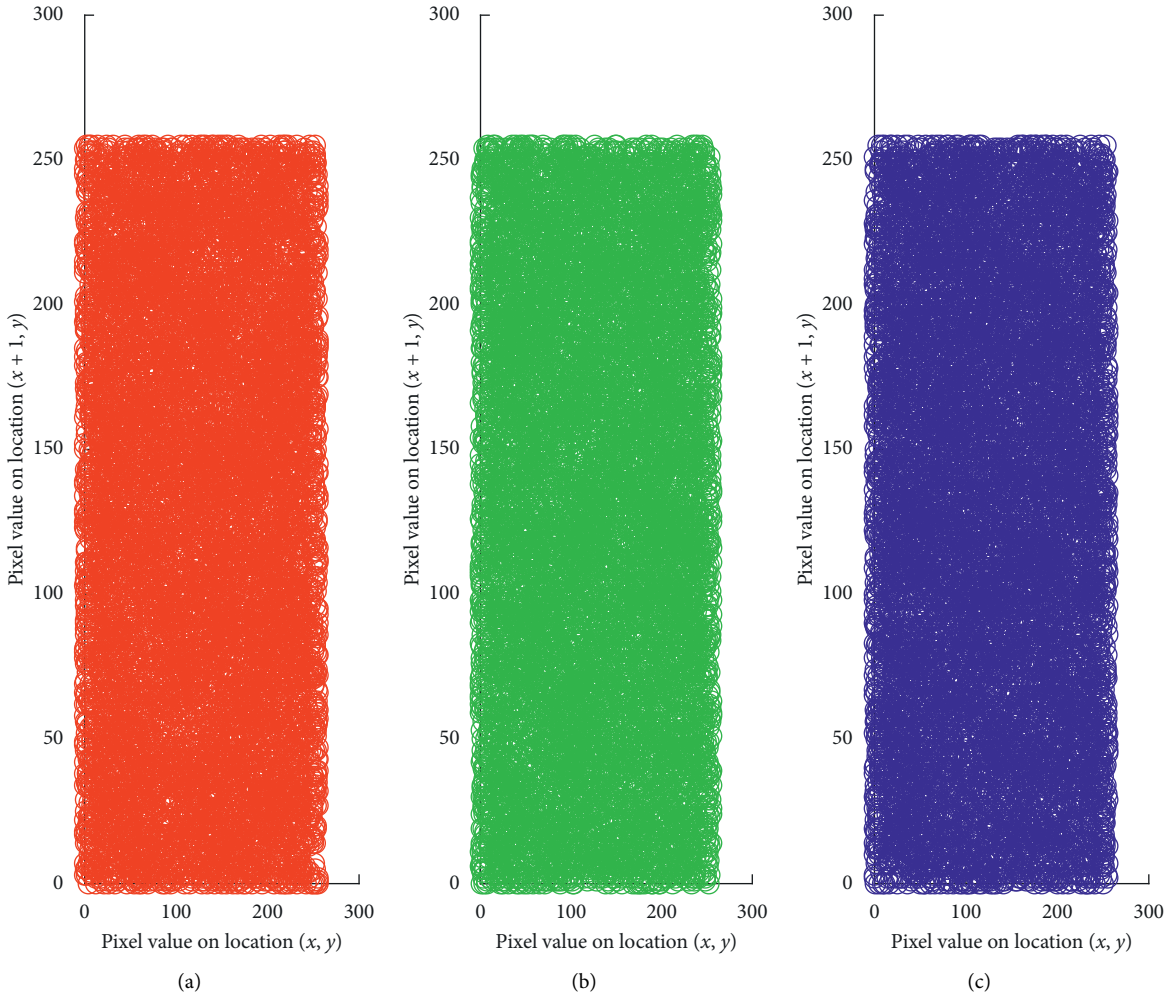


Figure 22: Correlation (diagonal wise) of encrypted image of onion. (a) Red component. (b) Green component. (c) Blue component.

Table 6: Correlation coefficient values of two adjacent pixels of onion (198 × 135 pixels) ciphered image.

Direction	Red		Green		Blue	
	Original image	Cipher image	Original image	Cipher image	Original image	Cipher image
Horizontal	0.9826	−0.0007	0.9786	−0.0068	0.9648	0.0003
Vertical	0.9900	−0.0034	0.9880	0.0046	0.9751	−0.0018
Diagonal	0.9721	−0.0054	0.9675	−0.0083	0.9427	−0.0021

where m, n represent the number of rows and columns, respectively. I_P and $I_D$represent the plain image and cipher image, respectively. For the difference between the plain image and cipher image, MSE ≥ 30 db. The MSE of proposed image algorithm is given in Table 5.

5.6. Peak Signal-to-Noise Ratio Analysis. The peak signal-to-noise ratio (PSNR) measures the conformity between the plain and cipher images. It can be calculated using the following formula:

$$\text{PSN} = 10.\ \log\frac{255^2}{\text{MSE}}\ (\text{db}). \tag{8}$$

Table 7: Comparison of size of key space.

Image encryption algorithms	Size of key space
Reference [30]	2^{193}
Reference [31]	2^{233}
Reference [32]	2^{138}
Reference [34]	2^{194}
Proposed algorithm	2^{282}

The value of PSNR should be as low as possible between the plain and cipher images for good encryption algorithms. The value of PSNR of the proposed algorithm is given in Table 5.

Table 8: Summary of properties comparison of different algorithms.

Algorithms	NPCR	UACI	Correlation coefficients			Entropy
			Horizontal	Vertical	Diagonal	
Reference [31]	99.22	33.40	0.0042	0.0033	0.0024	7.9967
Reference [32]	99.61	33.41	−0.0026	−0.0038	0.0017	7.9970
Reference [35]	99.61	33.47	−0.0075	−0.0011	−0.0012	7.9998
Reference [36]	99.62	30.91	−0.0049	0.0067	0.0010	7.9960
Proposed algorithm	99.61	33.46	0.0045	0.00016	−0.0013	7.9990

The proposed algorithms are also applied to another sample colored image of onion (198×135 pixels). The entropy value of onion image is 7.9975. The resulting encrypted and decrypted images are shown in Figure 15. The histogram of cipher image and correlation of neighboring pixels of plain and cipher images are shown in Figures 16–22 , respectively. Table 6 illustrates the values of correlation of neighboring pixels of cipher image of onion.

5.7. Key Space Analysis. The key space is all the possibilities of keys that can be utilized in the encryption algorithm. The size of key space is treated as a significant aspect of the algorithm. It should be huge enough to avoid brute-force attacks. With today's computing abilities, an algorithm can resist exhaustive attacks [33] if the size of key space is larger than 2^{128}. There are three keys involved in our proposed image encryption algorithm. The secret keys $K_1$and K_3contain parameters of associated chaotic maps which are ξ_0, η, ϕ_0, and β.By considering the precision of these parameters to be 2^{52}, the total number of possibilities of choosing these two keys will be $(2^{52})^2 \times (2^{52})^2 = 2^{208}$. The key $K_2 = (\text{color}\,1, \text{color}\,2, \text{color}\,3, k)$ is a combination of three random colors and a random number k. The number of possibilities for choosing three colors are ${}^{16777216}P_3 = 4.722365638 \times 10^{21} = 2^{72}$. The integers that satisfy the condition $\gcd(k, 256) = 1$ are $128 = 2^7$. So the total possibilities of choosing K_2 are $2^{72} \times 2^2 = 2^{74}$. The total size of key space is $2^{208} \times 2^{74} = 2^{282} > 2^{128}$. Therefore, our proposed algorithm is resistant against the brute-force attacks because the size of the key space is large enough. Table 7 lists the key space size of several schemes.

The computational complexity is analyzed as follows.

Assume that a fastest computer can calculate 2^{80} computations in one second. So, in one year, the number of computations performed by the computer is $2^{80} \times 365\,(\text{days}) \times 24\,(\text{hr}) \times 60\,(\text{min}) \times 60\,(\text{sec})$. Hence, the total of $(2^{282}/2^{80} \times 365 \times 24 \times 60 \times 60) = 10^{53}$ years is required. To resist the brute-force attack against this encryption algorithm, this computational load is large enough.

5.8. Key Sensitivity Analysis. An image encryption algorithm should be highly sensitive to its secret key, that is, a variation of single bit in secret key should yield a totally different cipher result. A highly sensitive key may contribute towards the security of the image encryption algorithm. The output of our decryption algorithm is totally changed with a slight modification in any part of the key $K = (K_1, K_2, K_3)$. Making even a slight variation in value of one part of encryption key ξ_0 as 0.7660000000000001, the image will be produced but not same as plain image. So, it is observed that the cipher image does not contain any clue or gesture about the original image. The proposed algorithm is highly sensitive to secret keys.

6. Conclusion

This study presents a novel color image scheme based on chaotic maps. In contrast to the traditional chaos-based cryptosystems, the suggested cryptosystem is proposed using Hill cipher and color codes. The confusion phase is done by the piecewise chaotic linear map. The Hill cipher with color codes is employed for the substitution phase. The diffusion process is performed by a chaotic logistic map and bitwise XOR. The key space size of the encryption algorithm is adequately high to combat brute-force attacks. Also, the algorithm is highly sensitive to keys. Several experimental tests have been carried out with detailed numerical analysis which exhibits the robustness of the suggested algorithm against numerous attacks such as statistical and differential attacks. The proposed image encryption algorithm is highly secure which is demonstrated by performing different assessment tests. The results of these experiments and performance tests are compared with different algorithms and summarized in Table 8.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors thank Taif University, Taif, Saudi Arabia, for its support under the project Taif University Researchers Supporting Project number (TURSP-2020/114).

References

[1] B. Park, A. Korbach, and R. Brünken, "Does thinking-aloud affect learning, visual information processing and cognitive load when learning with seductive details as expected from self-regulation perspective?" *Computers in Human Behavior*, vol. 111, Article ID 106411, 2020.

[2] B. Norouzi, S. Mirzakuchaki, S. M. Seyedzadeh, and M. R. Mosavi, "A simple, sensitive and secure image

encryption algorithm based on hyper-chaotic system with only one round diffusion process," *Multimedia Tools and Applications*, vol. 71, no. 3, pp. 1469–1497, 2014.

[3] E. N. Lorenz, "Atmospheric predictability as revealed by naturally occurring analogues," *Journal of the Atmospheric Sciences*, vol. 26, no. 4, pp. 636–646, 1969.

[4] S. Dhall, S. K. Pal, and K. Sharma, "Cryptanalysis of image encryption scheme based on a new 1D chaotic system," *Signal Processing*, vol. 146, pp. 22–32, 2018.

[5] R. Mathews, "On the derivation of a chaotic encryption algorithm," *Cryptolgia*, vol. 13, no. 1, pp. 29–42, 1989.

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

[7] E. Emad, "Watermarking 3D models using spectral mesh compression," *Signal Image and Video Processing*, vol. 3, p. 375, 2009.

[8] G. Chen, Y. Mao, and C. K. Chui, "A symmetric image encryption scheme based on 3D chaotic cat maps," *Chaos, Solitons & Fractals*, vol. 21, no. 3, pp. 749–761, 2004.

[9] Y. Mao, G. Chen, and S. Lian, "A novel fast image encryption scheme based on 3D chaotic Baker maps," *International Journal of Bifurcation and Chaos*, vol. 14, no. 10, pp. 3613–3624, 2004.

[10] Z. H. Guan, F. Huang, and W. Guan, "Chaos based image encryption algorithm," *Physics Letters A*, vol. 346, no. 1-3, pp. 153–157, 2005.

[11] V. Patidar, N. K. Pareek, G. Purohit, and K. K. Sud, "Modified substitution-diffusion image cipher using chaotic standard and logistic maps," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 10, pp. 2755–2765, 2010.

[12] X. Zhang and X. Wang, "Multiple image encryption algorithm based on mixed image element and chaos," *Computers and Electrical Engineering*, vol. 92, no. 6, 16 pages, 2017.

[13] Y. Luo, J. Yu, W. Lai, and L. Liu, "A novel chaotic image encryption algorithm based on improved baker map and logistic map," *Multimedia Tools and Applications*, vol. 78, no. 15, pp. 22023–22043, 2019.

[14] P. Ramasamy, V. Ranganathan, S. Kadry, R. Damaševičius, and T. Blažauskas, "An image encryption scheme based on block scrambling, modified zigzag transformation and key generation using enhanced logistic-tent map," *Entropy*, vol. 21, no. 7, p. 656, 2019.

[15] Z. Hua, F. Jin, B. Xu, and H. Huang, "2D logistic-sine-coupling map for image encryption," *Signal Processing*, vol. 149, pp. 148–161, 2018.

[16] X. Liao, Y. Yu, B. Li, Z. Li, and Z. Qin, "A new payload partition strategy in color image steganography," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 3, pp. 685–696, 2019.

[17] X. Liao, Z. Qin, and L. Ding, "Data embedding in digital images using critical functions," *Signal Process Image Commun*, vol. 58, pp. 146–156, 2017.

[18] K. A. Kumar Patro and B. Acharya, "An efficient colour image encryption scheme based on 1-D chaotic maps," *Journal of Information Security and Applications*, vol. 46, pp. 23–41, 2019.

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

[20] Z. Li, C. Peng, W. Tan, and L. Li, "An effective chaos-based image encryption scheme using imitating jigsaw method," *Complexity*, vol. 2021, Article ID 8824915, 18 pages, 2021.

[21] S. Xiao, Z. J. Yu, and YaS. Deng, "Design and analysis of a novel chaos-based image encryption algorithm via switch control mechanism," *Security and Communication Networks*, vol. 2020, Article ID 7913061, 12 pages, 2020.

[22] M. Gafsi, N. Abbassi, M. Ali Hajjaji, J. Malek, and A. Mtibaa, "Improved chaos-based cryptosystem for medical image encryption and decryption," *Scientific Programming*, vol. 2020, Article ID 6612390, 22 pages, 2020.

[23] K. A. K. Patro and B. Acharya, "An efficient dual-layer cross-coupled chaotic map security-based multi-image encryption system," *Nonlinear Dynamics*, vol. 10, 2021.

[24] M. García-Martínez, L. J. Ontañón-García, E. Campos-Cantón, and S. Čelikovský, "Hyperchaotic encryption based on multi-scroll piecewise linear systems," *Applied Mathematics and Computation*, vol. 270, pp. 413–424, 2015.

[25] E. C. Díaz-González, J.-A. López-Rentería, E. Campos-Cantón, and B. Aguirre-Hernández, "Maximal unstable dissipative interval to preserve multi-scroll attractors via multi-saturated functions," *Journal of Nonlinear Science*, vol. 26, no. 6, pp. 1833–1850, 2016.

[26] 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.

[27] L. S. Hill, "Cryptography in an algebraic alphabet," *The American Mathematical Monthly*, vol. 36, no. 6, pp. 306–312, 1929.

[28] L. S. Hill, "Concerning certain linear transformation apparatus of cryptography," *The American Mathematical Monthly*, vol. 38, no. 3, pp. 135–154, 1931.

[29] B. Acharya, G. S. Rath, S. K. Patra, and S. K. Panigrahy, "Novel methods of generating self-invertible matrix for hill cipher algorithm," *International Journal of Security*, vol. 1, no. 1, p. 14, 2007.

[30] S. M. Seyedzadeh, S. M. S. Moosavi, and S. Mirzakuchaki, "Using self-adaptive coupled piecewise nonlinear chaotic map for color image encryption scheme," in *Proceedings of the 2011 19th Iranian Conference on Electrical Engineering*, p. 1, Tehran, Iran, May 2011.

[31] X. Wei, L. Guo, Q. Zhang, J. Zhang, and S. Lian, "A novel color image encryption algorithm based on DNA sequence operation and hyper-chaotic system," *Journal of Systems and Software*, vol. 85, no. 2, pp. 290–299, 2012.

[32] C. Pak and L. Huang, "A new color image encryption using combination of the 1D chaotic map," *Signal Processing*, vol. 138, pp. 129–137, 2017.

[33] N. Chidambaram, P. Raj, K. Thenmozhi, and R. Amirtharajan, "Advanced framework for highly secure and cloud-based storage of colour images," *IET Image Processing*, vol. 14, no. 13, pp. 3143–3153, 2020.

[34] 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.

[35] K. A. K. Patro, B. Acharya, and V. Nath, "Various dimensional colour image encryption based on non-overlapping block-level diffusion operation," *Microsystem Technologies*, vol. 26, no. 5, pp. 1437–1448, 2020.

[36] R. Sivaraman, S. Rajagopalan, J. B. B. Rayappan, R. Amirtharajan, and R. Amirtharajan, "Ring oscillator as confusion - diffusion agent: a complete TRNG drove image security," *IET Image Processing*, vol. 14, no. 13, pp. 2987–2997, 2020.