Alanezi, A, Abd-El-Atty, B, Kolivand, H, Abd El-Latif, AA, Abd El-Rahiem, B, Sankar, S and S. Khalifa, H

Securing Digital Images through Simple Permutation-Substitution Mechanism in Cloud-Based Smart City Environment

http://researchonline.ljmu.ac.uk/id/eprint/14544/

Article

Hindawi
Security and Communication Networks
Volume 2021, Article ID 6615512, 17 pages
https://doi.org/10.1155/2021/6615512

Research Article

Securing Digital Images through Simple Permutation-Substitution Mechanism in Cloud-Based Smart City Environment

Ahmad Alanezi,[1] Bassem Abd-El-Atty ,[2,3] Hoshang Kolivand ,[1] Ahmed A. Abd El-Latif ,[2,3] Basma Abd El-Rahiem,[2,3] Syam Sankar,[4] and Hany S. Khalifa[5]

[1]*Department of Computer Science, Faculty of Engineering and Technology, Liverpool John Moores University (LJMU), Liverpool L3 3AF, UK*
[2]*Department of Mathematics and Computer Science, Faculty of Science, Menoufia University, Al Minufiyah 32511, Egypt*
[3]*Center of Excellence in Cybersecurity, Quantum Information Processing, and Artificial Intelligence, Menoufia University, Al Minufiyah 32511, Egypt*
[4]*Department of Computer Science and Engineering, NSS College of Engineering, Palakkad, Kerala, India*
[5]*Computer Science Department, Misr Higher Institute of Commerce and Computers, Mansoura, Egypt*

Correspondence should be addressed to Ahmed A. Abd El-Latif; a.rahiem@gmail.com

Received 17 October 2020; Revised 17 December 2020; Accepted 6 February 2021; Published 26 February 2021

Academic Editor: Sen Bai

Data security plays a significant role in data transfer in cloud-based smart cities. Chaotic maps are commonly used in designing modern cryptographic applications, in which one-dimensional (1D) chaotic systems are widely used due to their simple design and low computational complexity. However, 1D chaotic maps suffer from different kinds of attacks because of their chaotic discontinuous ranges and small key-space. To own the benefits of 1D chaotic maps and avoid their drawbacks, the cascading of two integrated 1D chaotic systems has been utilized. In this paper, we report an image cryptosystem for data transfer in cloud-based smart cities using the cascading of Logistic-Chebyshev and Logistic-Sine maps. Logistic-Sine map has been utilized to permute the plain image, and Logistic-Chebyshev map has been used to substitute the permuted image, while the cascading of both integrated maps has been utilized in performing XOR procedure on the substituted image. The security analyses of the suggested approach prove that the encryption mechanism has good efficiency as well as lower encryption time compared with other related algorithms.

1. Introduction

In smart cities environment, the data generated from various sources (smart city applications) are usually kept inside a cloud server and are manipulated by the concerned government officials and citizens of the city [1, 2]. The data of citizens include healthcare information, purchase behavior, weather conditions, environmental changes, and transport information. The majority of the data take the form of images. Image data concerning the day to day activities of people are extremely sensitive and critical. In order to save the data from exploitation by third parties, we need to build efficient encryption mechanisms so that they can be integrated with the cloud system for secure storage.

In the digital era of processing multimedia data by almost all of the electronic devices, technologists, researchers, and scientists are actively involved in the design and development of powerful cryptosystems. Since the images are the inevitable source of digital data in today's world, cryptosystem builders are focusing more on devising techniques that encipher the actual image information and in no way that the opponents must be able to disclose it. In their concrete terminology, images are nothing but a matrix of numbers. An encryption algorithm, through its reversible set of operations, conceals the actual pixel values. To achieve better security, we should mainly focus on four parameters: design of efficient confusion and diffusion strategy, reducing the correlation of neighboring pixels, enhancing the entropy value of the encrypted image, and huge key-space.

It is observed that, due to the high correlation amongst pixels of the image, popular enciphering mechanisms like Data Encryption Standard (DES) and Advanced Encryption Standard (AES) are not effectively suitable for enciphering images. The active area of research in designing encryption algorithms for images is chaos-based cryptography [3, 4]. It basically involves the processing of images with a sequence of random numbers produced by a chaotic system. Chaotic systems are mathematical functions with the property of being sensitive to primary values of parameters. The values of parameters or constants of the chaotic maps act as keys to the cryptosystem. Various researchers use maps like Lorentz map, Arnold-Tent map, Cat map, etc., to generate random numbers so that they can be applied on the image to change the original pixel values (substitution or diffusion) and to shift the position of actual pixel values (permutation or confusion). To enhance the security in all levels, the researchers use the concepts of DNA sequence operation [5], Cellular Automata [6], substitution-box [7, 8], finite-state machine [9], fractal sorting matrix [10], bit-level permutation [11], compressive sensing [12, 13], etc., along with chaos-based cryptography.

As we explained earlier, the permutation-substitution process decides the level of security attained by the final cipher image. It is considered as the two basic steps in encryption. Most of the researchers primarily focus on developing an efficient permutation strategy by devising a separate shifting algorithm that can effectively conceal the original image information. Traditional image permutation mechanisms like sort-based, Arnold-based, Baker-based, cyclic shift-based permutation mechanisms, etc., have disadvantages like weak permutation and high time complexity [14]. Based on these limitations, Wang et al. [14] presented a new permutation method to achieve low time complexity based on the combination of cyclic shift with sorting, and Hao et al. [15] presented a new permutation mechanism called "chaotic magic transform" using a two-dimensional chaotic system to achieve low time complexity and efficient permutation of image pixels. In substitution, the pixel value is getting changed, mostly because of the XOR operation between the set of random sequences produced by the chaotic map and the image matrix. Researchers have the maximum flexibility in deciding steps in the encryption algorithm. The selection of a particular step (permutation or substitution) in the algorithm is decided by three factors: the step must be reversible, yield an enhanced value of evaluation parameters, and have fast running speed.

Researchers in their new works clearly show the design of one-dimensional (1D) chaotic systems [14, 16, 17] and multidimensional chaotic maps [10–13] with an exclusive application in image encryption. Every work is focusing on how effectively we can improve various measures of security analysis. The 1D chaotic systems enjoy powerful benefits like being easy to design, having low computational complexity, having high-speed processing, and having simple structure. However, 1D maps have a weakness to several attacks because their initial values have a small key-space and chaotic discontinuous ranges [16].

Therefore, cascading systems are the solution to possess the benefits of iterating 1D chaotic maps and avoiding their drawbacks. Recently, Zhou et al. [18] proposed a new cascading system of two 1D systems (Tent, Sine, and Logistic) and presented its application in image encryption. In this work, we report a new image cryptosystem using the cascading of two integrated 1D chaotic systems (Logistic-Chebyshev and Logistic-Sine). In the suggested cipher approach, Logistic-Sine map is utilized to permute the plain image, and Logistic-Chebyshev map is used to substitute the permuted image. Cascading of both maps is used in performing XOR procedure on the substituted image. The experimental outcomes of the suggested approach prove that the encryption system has good efficiency and low running time for encryption compared to other related mechanisms.

The structure of our paper is as follows: the proposed framework for cloud-based smart city is provided in Section 2, while the elementary knowledge of the utilized chaotic integrated maps is delivered in Section 3. The presented image cryptosystem is given in Section 4, while the security analyses of the presented encryption mechanism are given in Section 5. As a final point, the conclusions are provided in Section 6.

2. Proposed Framework for Cloud-Based Smart City

Smart cities are an enhanced urban infrastructure in which citizens are offered high-quality life in a way of efficient delivery of various services such as transportation, e-governance, waste management, healthcare, education, and water supply. The information and communication technologies are effectively deployed to satisfy the desired level of service delivery. The smart city environment involves data reception from various sensors, devices, and people and is processed further to make final decisions. Large amounts of data are generated every day through various smart city applications. For the efficient storage and management of such data, cloud servers are deployed. Cloud computing allows us to access various services hosted remotely. Smart city applications make use of services cloud environment. Data security, authenticity, and integrity policies are also integrated along with the cloud platform for smooth data transmission and storage. The privacy of digital contents is of utmost importance [19, 20].

We intend to propose a cryptosystem that enciphers the digital image data captured from the smart city environment. Figure 1 shows the proposed framework for secure data transfer in cloud-based smart cities. The image data might represent medical information of patients, live traffic blocks and violation, climatic conditions, suspicious people, etc. The encrypted image data can be transferred and stored in interconnected cloud servers. The users at the other end

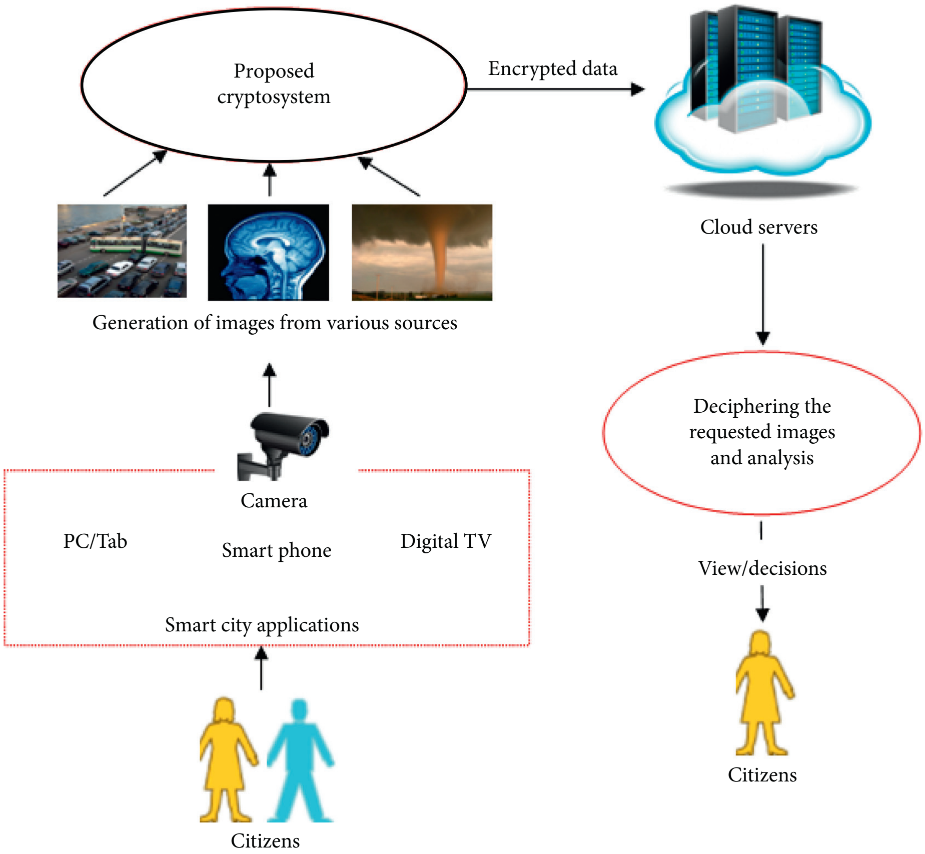


Figure 1: Framework outline of secure data transmission in cloud-based smart city.

can make use of the image data after applying the deciphering mechanism. Since the data are stored in an encrypted form at cloud servers, opponents must not be able to view and interpret the same. The following sections cover the concepts related to the proposed image cryptosystem.

3. Preliminary Knowledge

The proposed image cipher approach is based on cascading two integrated 1D chaotic systems: Logistic-Chebyshev and Logistic-Sine. The following subsections detail the two chaotic systems.

3.1. Logistic-Chebyshev Map

3.1. Logistic-Chebyshev Map. Logistic-Chebyshev map is an integration of two common 1D chaotic systems: Logistic and Chebyshev. It can be expressed as[7]

$$LC_{i+1} = \left[\alpha \times LC_i\left(1 - LC_i\right) + \frac{(4-\alpha)\cos\left(A \times \ \arccos\left(LC_i\right)\right)}{4}\right] \bmod 1, \tag{1}$$

where $\alpha \in (0,4)$ is the control parameter, $LC_0 \in (0,1)$ is the primary value of the system, and $A \in N$ is the degree of the chaotic map.

3.2. Logistic-Sine Map. Logistic-Sine map is an integration of two 1D chaotic maps: Logistic map and Sine map, which can be expressed as [16]

$$LS_{i+1} = \left(\beta\left(LS_i - LS_i^2\right) + (4-\beta)\frac{\sin\left(\pi \times LS_i\right)}{4}\right) \bmod 1, \tag{2}$$

where $\beta \in (0,4)$ is the control parameter and $LS_0 \in (0,1)$ is the original value.

4. Proposed Image Cipher Approach

In this part, we explain a new image cryptosystem using cascading Logistic-Chebyshev and Logistic-Sine maps. Logistic-Sine system is utilized to permute the plain image, and Logistic-Chebyshev map is used for substituting the permuted image, while the cascading of both integrated maps is utilized in performing XOR process on the substituted image. The architecture of the proposed approach is provided in Figure 2, whereas the encryption processes are provided in Algorithm 1.

5. Experimental Results

To estimate the performance of the presented encryption system, we used a laptop with Intel Core™ i5-2450M CPU 2.50 GHz and 6 GB RAM with preinstalled MATLAB software R2016b. We used standard test images from SIPI database [21] of dimension 512×512 as shown in Figure 3, which labeled as Boats, Bridge, Baboon, Sailboat, Airplane, and Peppers. The key parameters utilized to iterate Logistic-Chebyshev and Logistic-Sine maps are initialized as $LC_0 = 0.684$, $\alpha = 3.356$, $A = 152$, $LS_0 = 0.4794$, and $\beta = 3.8435$.

5.1. NIST SP 800-22 Test. To check the random characteristics of the generated sequence from cascading chaotic maps and the constructed encrypted images, we used NIST SP 800-22 tests. The crucial task of these tests is to measure the randomness property of a sequence and detect any non-random characteristics existing in the sequence. The outcome of each test generates a P-value in range [0, 1]. When the P-value is greater than the threshold value $\mu = 0.01$, this indicates that the sequence passes this test [22]. The outcomes of NIST SP 800-22 tests are stated in Table 1, where the two sequences of the cipher image Enc-Sailboat and its used key stream that was generated from chaotic maps passed all NIST SP 800-22 tests.

5.2. Time Efficiency. To verify the time effectiveness for the encryption process of our cryptosystem, Table 2 shows a simple comparison for encryption time for the proposed image cryptosystem with related cryptosystems for different sizes of images. Outcomes of encryption time for related works are given as reported in [10, 15, 16, 23]. The encryption time given in Table 2 confirms that our mechanism is superior to other ones in terms of time encryption.

5.3. Correlation Analysis. In plain images, per pixel is profoundly correlated with its neighboring pixels, and the value of correlation is imminent to 1 in all directions (horizontal, vertical, and diagonal). On the other hand, for the generated ciphered images using a well-designed image cryptosystem, the correlation values should be imminent to 0 [24]. For calculating correlation values of cipher images and their corresponding plain ones, we picked randomly 10^4 pairs of adjacent pixels in each direction.

$$V = \frac{\sum_{x=1}^{T} \left(P_x - \overline{P}\right)\left(C_x - \overline{C}\right)}{\sqrt{\sum_{x=1}^{T} \left(P_x - \overline{P}\right)^2 \sum_{x=1}^{T} \left(C_x - \overline{C}\right)^2}}, \tag{3}$$

where T denotes to the full number of neighboring pixel pairs in every direction, and P_x and C_x denote the values of neighboring pixels. The correlation values of the experimented datasets are provided in Tables 3 and 4, in which the correlation values of cipher images are very near to 0. The correlation distribution of neighboring pixels for greyscale Boats image before and after encryption is plotted in Figure 4, and the correlation distribution of color Sailboat image before and after encryption is plotted in Figures 5–7 . From the stated results in Tables 3 and 4, and the stated correlation distributions in Figures 4–7, we can conclude that our image cryptosystem is secure against correlation analysis.

5.4. Pixels Change Rate. To assess the plain image sensitivity to tiny bit changes, two measures are applied: Unified Average Changing Intensity (UACI) and Number of Pixels Change Rate (NPCR). The mathematical representations of NPCR and UACI can be declared as follows:

$$\mathrm{NPCR} = \frac{\sum_{x,y} \mathrm{Diff}(x, y)}{T} \times 100\%,$$

$$\mathrm{Diff}(x, y) = \begin{cases} 0, & \text{if } C1(x, y) = C2(x, y), \\ 1, & \text{if } C1(x, y) \neq C2(x, y), \end{cases} \tag{4}$$

$$\mathrm{UACI} = \frac{1}{T}\left(\sum_{x,y} \frac{\left|C1(x, y) - C2(x, y)\right|}{2^b - 1}\right) \times 100\%. \tag{5}$$

Here, $C1$, $C2$ are two encrypted images for one plain image with changes in one bit, T points to the full number of pixels used in the image, and b expresses the number of bits used to describe the pixel value. The NPCR and UACI outcomes of plain and cipher images are given in Table 5, which demonstrated that our image cryptosystem is highly sensitive to tiny pixel variations in the plain image.

5.5. Histogram Analysis. Histogram points to the frequency distribution of pixel values in the image. A well-designed cryptosystem should guarantee the uniformity of the histograms for different encrypted images. Figure 8 displays the histograms of greyscale images before and after the encryption process, and also Figure 9 displays the histograms of the plain and cipher Sailboat image, in which the histograms of plain images differ from each other and the histograms of the corresponding cipher images are uniform. However, we need a mathematical quantity analysis to check the histogram test; therefore, we perform Chi-square test (χ^2), which can be expressed as[25]

$$\chi^2 = \sum_{i=0}^{255} \frac{\left(f_i - s\right)^2}{s}, \tag{6}$$

Here, f_i denotes the frequency of the pixel value i, and s is the image dimension. By supposing that the significant level is $\lambda = 0.05$, then $\chi^2_\lambda(255) = 293.25$. For a given image, when the χ^2 value is smaller than $\chi^2_\lambda(255)$, this confirms the uniformity of the histogram for this image; otherwise, the image has nonuniform distribution. Tables 6 and 7 provide the results of χ^2 for the investigated dataset, in which the χ^2

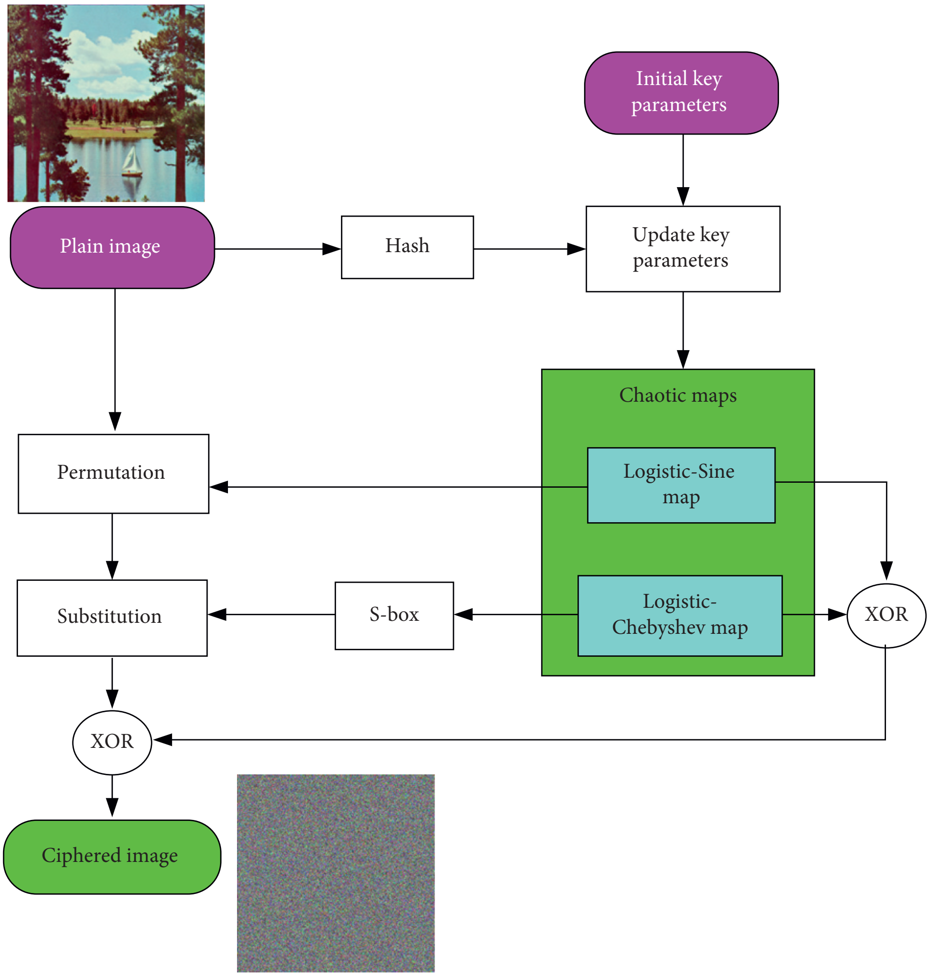


Figure 2: The architecture of the proposed image cryptosystem.

values for all cipher images are smaller than $\chi^2_\lambda(255)$. Consequently, the presented encryption algorithm can resist histogram analysis attacks.

5.6. Information Entropy Analysis. To compute the distribution of pixel values per level in the image, we employed the global entropy test, which can be stated as follows:

$$E(X) = \sum_{i=0}^{255} p(x_i) \log_2 \frac{1}{p(x_i)}, \qquad (7)$$

Here $p(x_i)$ denotes the probability of x_i. The probable values for a gray-scale image are 2^8, and then the ideal entropy value is equal to 8 bits. To assess the efficiency of the suggested image cryptosystem, the entropy values for the cipher image must be extremely nearby 8. Notwithstanding, the global entropy is ignoring to assess the true randomness for encrypted images. Consequently, local entropy can be estimated by the mean of global entropies for nonoverlapping blocks (1936-pixel per block). Table 8 displayed the values of global and local entropies for the plain images and their corresponding ciphered ones, in which all values of information entropy for encrypted images are really near 8 bits. Consequently, the suggested cryptosystem is protected against entropy attacks.

5.7. Contrast Analysis. To estimate the variation of local intensity that existed in an image, we employed the contrast test, which is a statistical measure and defined as given in [26]

$$\text{Con} = \sum_{x,y} |x - y|^2 p(x, y), \qquad (8)$$

Here, $p(x, y)$ denotes the number of gray-level cooccurrence matrices. For a given image, high contrast values denote that the image has significantly various gray levels, whereas lower values indicate constant gray levels. Contrast values of plain and cipher images are provided in Tables 9 and 10, in which all cipher images possess high contrast values.

5.8. Peak Signal-to-Noise Ratio Analysis. To measure the noise ratio between the plain and cipher images, we employed peak signal-to-noise ratio (PSNR) tool, which can be defined as[27]

```
**Input:** Plain image $(P)$
**Parameters:** $LS_0$, $\beta$, $LC_0$, $\alpha$, $A$//Used for iterating chaotic maps.
**Output:** Cipher image $(C)$ and decimal values $(H_1, H_2, H_3 \text{ and } H_4)$
$[m\ \ n\ \ c] = \text{size}\ (p)$//Get image dimension.
$Hb = hash(P)$//Compute the hash value $Hb$ for image $P$ using SHA-256 algorithm.
$H = u\text{int8}(Hb)$//Convert the 256 bit hash value to 32 integer values $h_1, h_2, \ldots, h_{32}$, where each integer composed of 8 bit.
$H_1 = (h_1 \oplus h_2 \oplus \cdots \oplus h_8)/256$;
$H_2 = (h_9 \oplus h_{10} \oplus \cdots \oplus h_{16})/128$;
$H_3 = (h_{17} \oplus h_{18} \oplus \cdots \oplus h_{24})/256$;
$H_4 = (h_{25} \oplus h_{26} \oplus \cdots \oplus h_{32})/128$;
//Update initial key parameters $(LS_0, \beta, LC_0, \alpha)$ using $H_1$, $H_2$, $H_3$, and $H_4$
$LC_0 = (LC_0 + H_1)/2$;
$\alpha = \alpha/2 + H_2$;
$LS_0 = (LS_0 + H_3)/2$;
$\beta = \beta/2 + H_4$;
$\{LC\}$ = Logistic-Chebyshev $(LC_0, \alpha, A, m \times n \times c)$//Using the updated key parameters $(LC_0, \alpha, A)$, operate Logistic-Chebyshev map
    for $m \times n \times c$ times to generate sequence $\{LC\}$, wherever the size of $P$ is $m \times n$ and $c$ denotes the number of color channels.
$\{LS\}$ = Logistic-Sine $(LS_0, \beta, m \times n \times c)$//Using the updated key parameters $(LS_0, \beta)$, operate Logistic-Sine system for $m \times n \times c$ times
    to create sequence $\{LS\}$.
$\{KC\} = \text{fix}(LC_i \times 10^{12} \bmod 256)$//Convert sequence $\{LC\}$ into integer values.
$\{KS\} = \text{fix}(LS_i \times 10^{12} \bmod 256)$;
$Per\text{Im} = \text{permutation}(P, \{\text{KS}\})$//Permute the input image $(P)$ using the sequence $\{KS\}$ and chaotic magic transform method
    presented in [15].
$S\text{box} = \text{unique}(\{KC\})$//Collect the first 256 unequal elements in the sequence $\{KC\}$ to construct the substitution-box (S-box).
//Substitution process.
for $I = 1{:}\ m$
    for $j = 1{:}\ n$
        for $k = 1{:}\ c$
            $\text{Sim}(i, j, k) = S\text{box}(Per\text{Im}(i, j, k) + 1)$;
        end
    end
end
Key = $KS \oplus KC$//Cascade both sequences ($\{KC\}$ and $\{KS\}$) to generate the key sequence Key.
$C = \text{Sim} \oplus \text{key}$//Cipher image
```

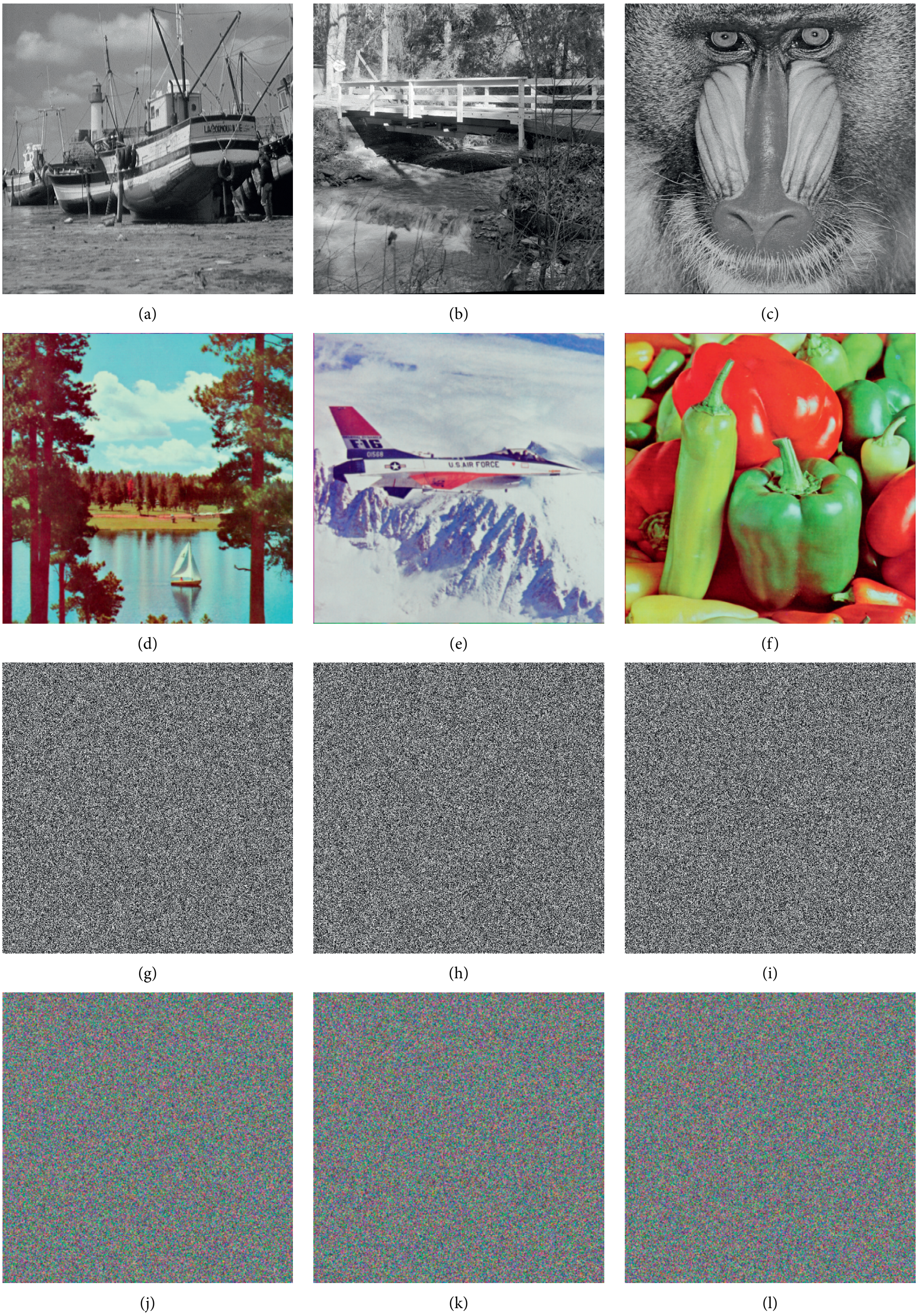
Algorithm 1: Encryption algorithm.

$$\text{PSNR}(P, C) = 20\log_{10}\left(\frac{\text{MAX}_P}{\sqrt{\text{MSE}}}\right),$$
$$\text{MSE} = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}[P(x, y) - C(x, y)]^2, \tag{9}$$

Here, MAX_P is the maximum pixel value of the plain image P, while C indicates its corresponding cipher image, and the dimensions of P and C are $M \times N$. Higher PSNR values denote that the cipher image is near to the plain image. Therefore, a well-designed encryption algorithm should have low PSNR values, which signify that the cipher image is significantly dissimilar from its corresponding plain image. The outcomes of PSNR and MSE values for the investigated dataset are provided in Table 11, in which the PSNR values are very low.

5.9. Key-Space and Key Sensitivity Analyses. The key-space referred to the various keys that can be applied in brute force attacks and must be large enough to resist those attacks. Our image cryptosystem uses key parameters $(LS_0, \beta.LC_0, \alpha, \text{ and } A)$ to operate chaotic maps during the encryption and decryption processes. By supposing that the computation precision of digital computers is 10^{-16}, the key-space for our cryptosystem is 10^{80}, which is large enough for any modern cryptographic mechanism.

Key sensitivity indicates that any slight modifications in the initial keys lead to significant variations in the outcome. To evaluate the key sensitivity of the presented cryptosystem, the encrypted Sailboat image is deciphered amidst tiny modifications in the primary keys. The outcomes of the key sensitivity for the presented mechanism are provided in Figure 10. Also, to evaluate the key sensitivity of our cryptosystem in quantity terms, we perform NPCR and UACI on decrypted Sailboat image with the correct key and other decrypted Sailboat images with tiny modifications in the initial keys in which the outcomes are stated in Table 12. From the results stated in Table 12 and Figure 10, our cryptosystem has high key sensitivity, in which any slight modifications in the initial keys lead to significant variations in the outcome.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

Figure 3: The first two rows display the used investigation images, whereas the last two rows display their ciphered images using our presented cryptosystem. (a) Boats. (b) Bridge. (c) Baboon. (d) Sailboat. (e) Airplane. (f) Peppers. (g) Enc-Boats. (h) Enc-Bridge. (i) Enc-Baboon. (j) Enc-Sailboat. (k) Enc-Airplane. (l) Enc- Peppers.

Table 1: Outcomes of NIST SP 800-22 tests for the cipher image Enc-Sailboat and its used key stream that was generated from chaotic maps.

Test name		*P*-value		Passed
		Enc-sailboat	Key stream	
Random excursions variant ($x = 1$)		0.758288	0.911716	√
Rank		0.277427	0.023295	√
Random excursions ($x = 1$)		0.759421	0.324551	√
Long runs of ones		0.538239	0.047194	√
Overlapping templates		0.309669	0.864874	√
Frequency		0.275713	0.305835	√
Linear complexity		0.299882	0.348444	√
Block-frequency		0.469785	0.646149	√
Runs		0.423022	0.456241	√
No overlapping templates		0.686946	0.311721	√
Universal statistical		0.943058	0.638841	√
Spectral DFT		0.912314	0.890517	√
Approximate entropy		0.358094	0.373851	√
Serial	Test 1	0.752991	0.278112	√
	Test 2	0.551133	0.465868	√
Cumulative sums	Reverse	0.388377	0.348202	√
	Forward	0.483105	0.282021	√

Table 2: Comparisons of encryption time (in seconds) for the proposed image cryptosystem with related cryptosystems for different sizes of images.

Encryption scheme	Image size		
	256 × 256	512 × 512	1024 × 1024
Our proposed method	0.0494	0.3033	1.0453
Ref. [10]	0.0779	0.3261	1.3146
Ref. [15]	0.0538	0.2338	1.1494
Ref. [16]	0.1789	0.6639	3.1426
Ref. [23]	0.0949	0.4010	1.9857

Table 3: Correlation coefficients of the experimented greyscale dataset.

Image	Direction		
	Hor.	Ver.	Dia.
Boats	0.9713	0.9367	0.9212
Enc-boats	−0.0011	0.0004	−0.0005
Bridge	0.9270	0.9397	0.8937
Enc-bridge	0.0015	0.0004	0.0011
Baboon	0.7623	0.8641	0.7274
Enc-baboon	−0.0002	−0.0001	0.0011

Table 4: Correlation coefficients of the tested color dataset.

Image	Direction								
	Hor.			Ver.			Dia.		
	Red	Green	Blue	Red	Green	Blue	Red	Green	Blue
Sailboat	0.9562	0.9701	0.9708	0.9565	0.9736	0.9735	0.9464	0.9584	0.9558
Enc-sailboat	−0.0002	−0.0003	−0.0002	0.0007	0.0002	−0.0012	−0.0006	−0.0014	0.0007
Airplane	0.9625	0.9710	0.9455	0.9721	0.9623	0.9639	0.9391	0.9406	0.9263
Enc-airplane	−0.0003	−0.0011	−0.0001	0.0002	−0.0001	−0.0001	0.0002	−0.0005	0.0005
Peppers	0.9682	0.9846	0.9689	0.9670	0.9832	0.9671	0.9619	0.9722	0.9518
Enc-peppers	−0.0012	−0.0009	0.0009	0.0003	−0.0008	−0.0009	0.0003	0.0003	−0.0004

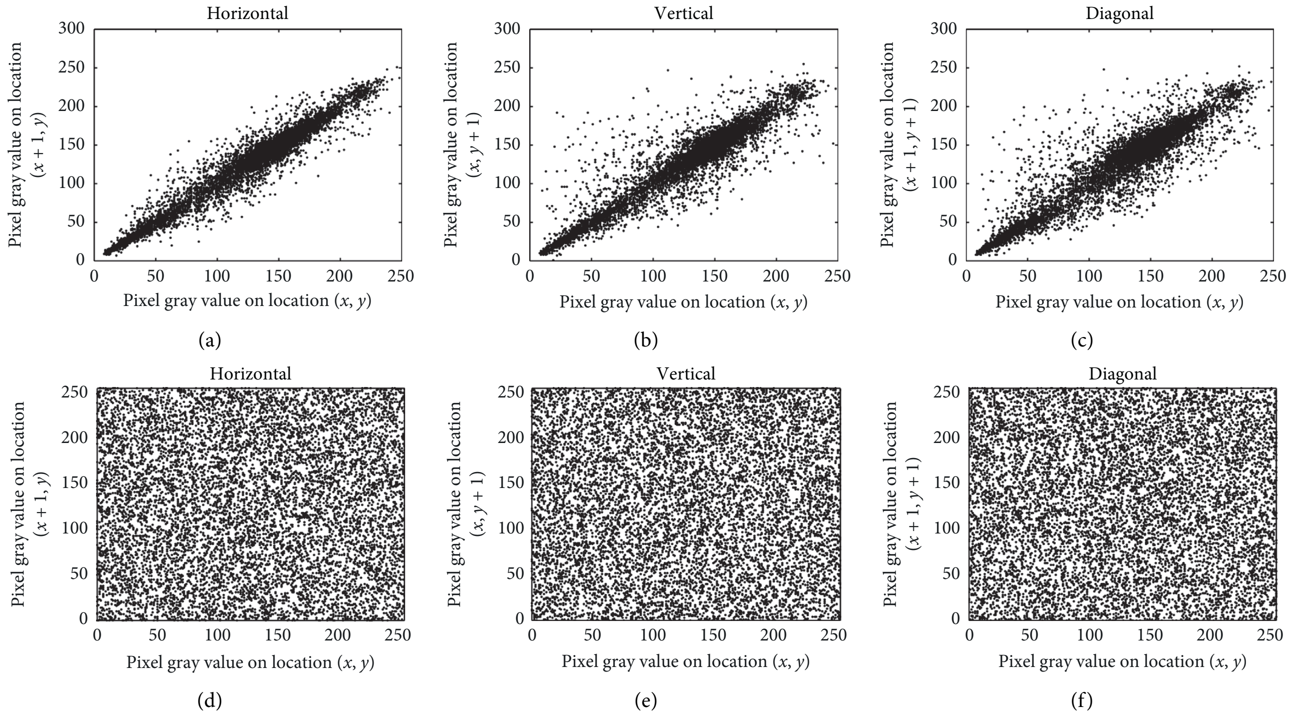


Figure 4: Correlation distribution of Boats image, where the correlation distribution of the plain image is stated in the first row, and the correlation distribution of the cipher image is stated in the last row.

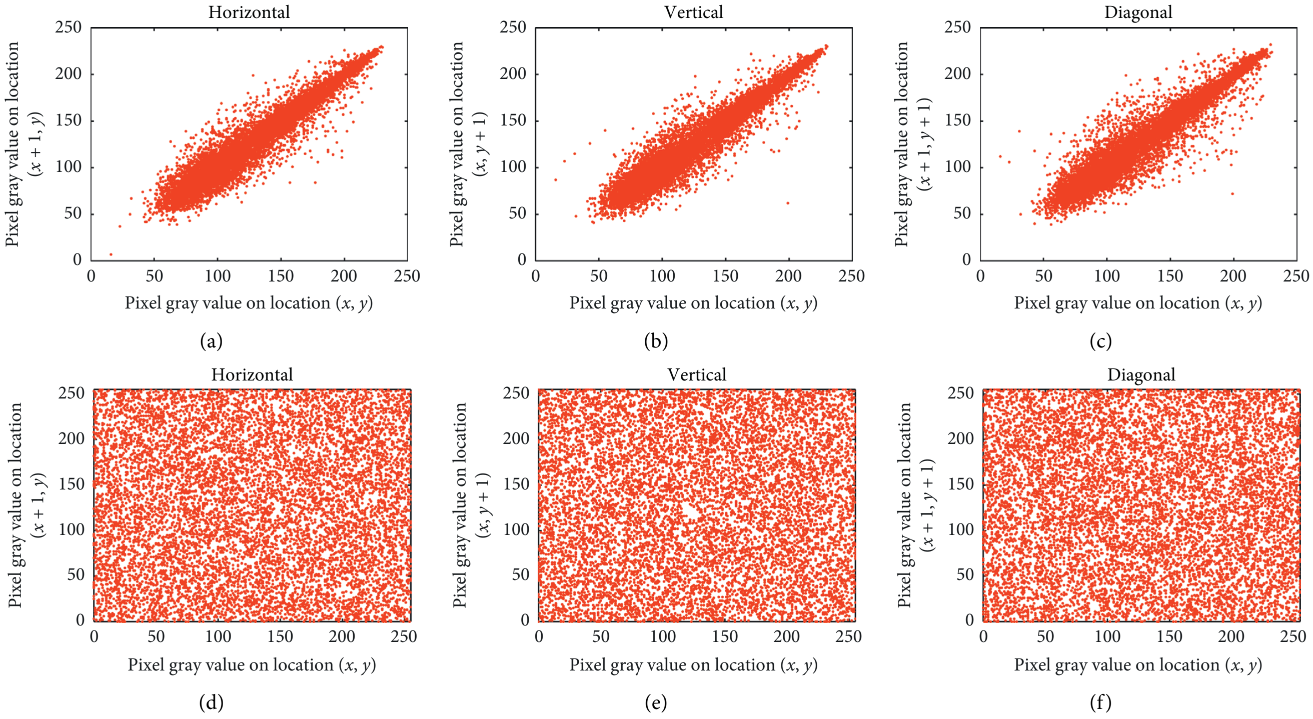


Figure 5: Red channel of Sailboat-Correlation distribution.

5.10. Classical Types of Attack. During the cryptanalyses of a cryptosystem, it is generally assumed that cryptanalysts have a complete understanding of the design of the cryptosystem and know everything regarding the cryptosystem except the values of initial key parameters. This is an obvious requirement in today's cryptosystems. There are four kinds of classic attacks: ciphertext only, known-plaintext, chosen-plaintext, and chosen-ciphertext. The chosen-plaintext

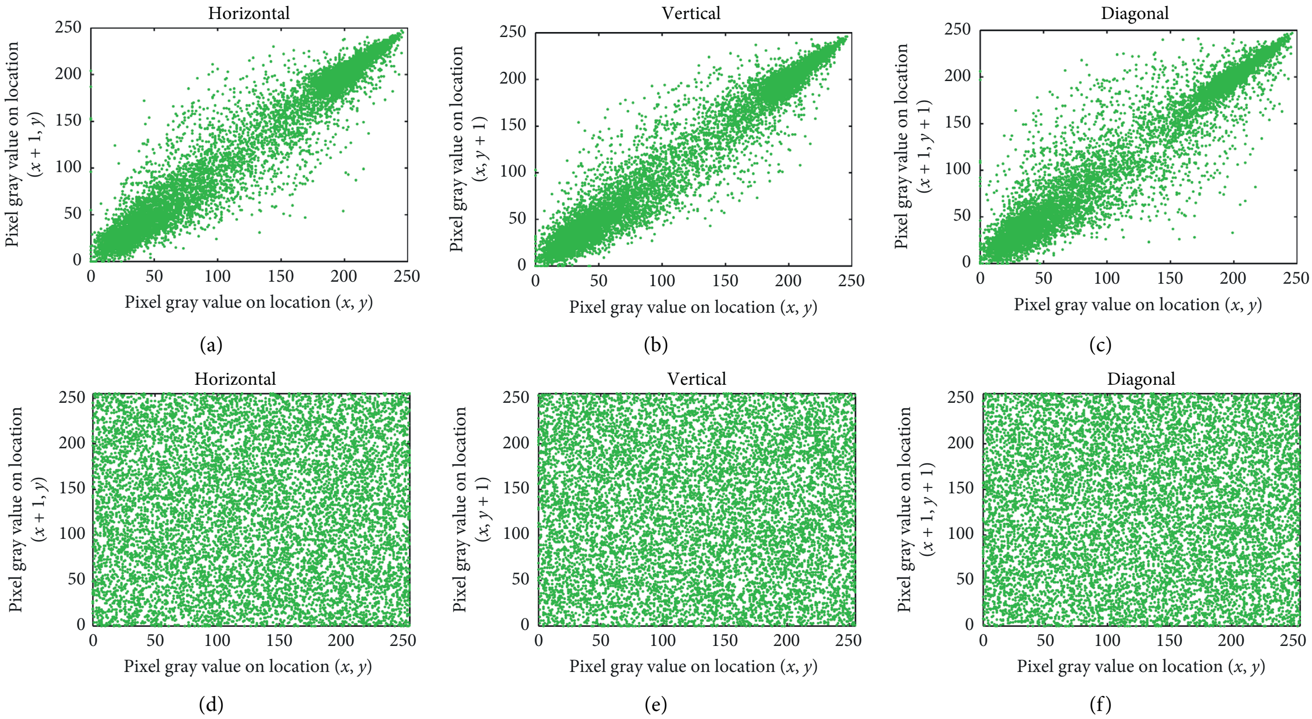


Figure 6: Green channel of Sailboat-Correlation distribution.

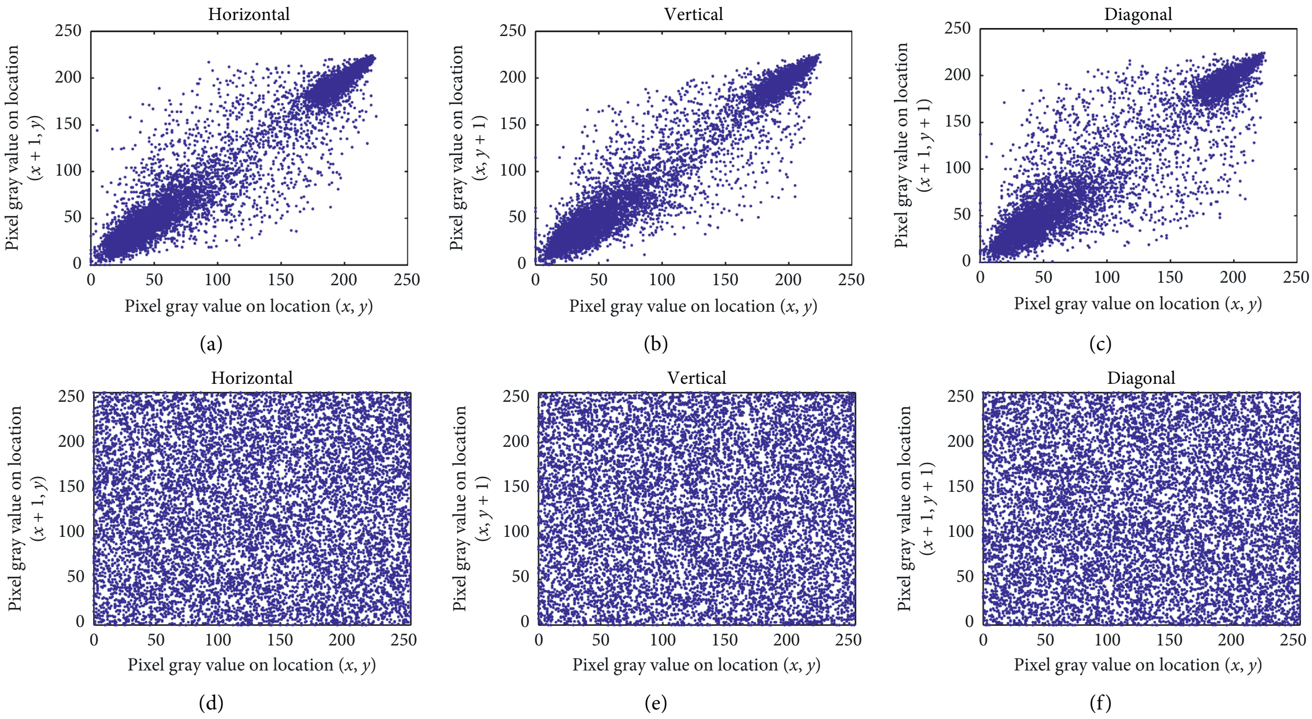
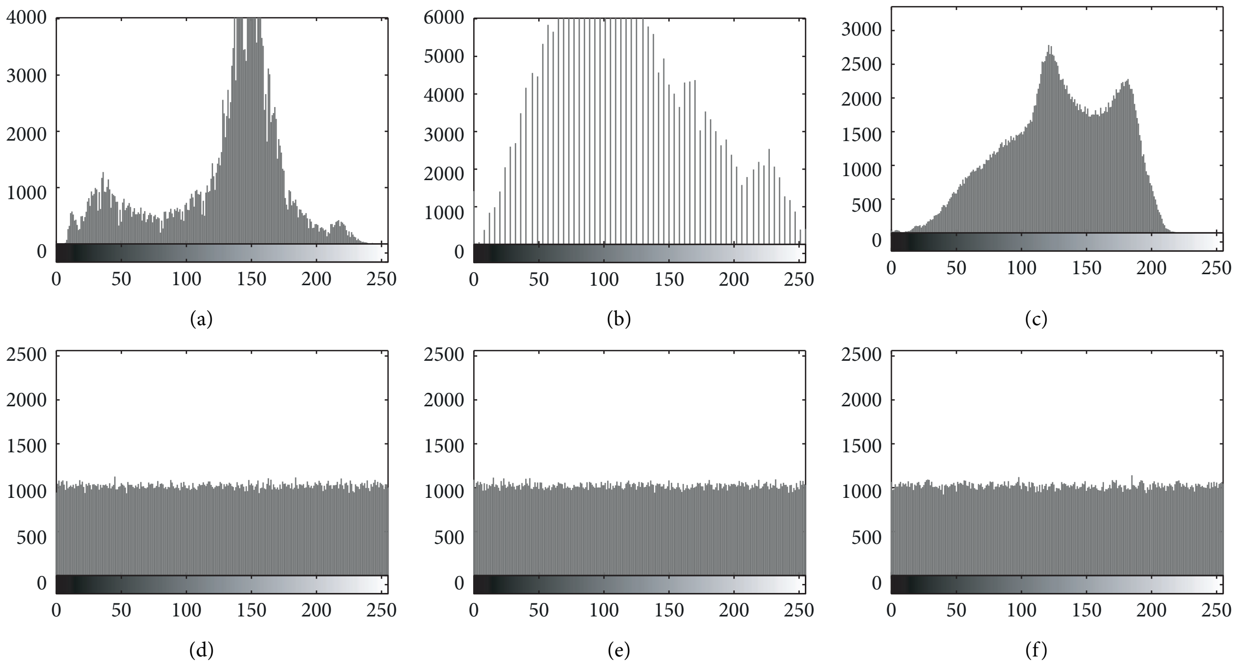


Figure 7: Blue channel of Sailboat-Correlation distribution.

attack is known to be the most powerful attack, in which the hacker has temporary access to the cryptosystem and can construct the ciphertext corresponding to a chosen-plaintext. If a cryptosystem has the capability to withstand the chosen-plaintext attack, it possesses the ability to withstand other types of attacks. The presented cryptosystem

Table 5: NPCR and UACI values of the experimented datasets.

Image	NPCR (%)	UACI (%)
Boats	99.61776	33.44965
Bridge	99.62539	33.46607
Baboon	99.62387	33.56153
Sailboat	99.62043	33.44339
Airplane	99.62234	33.45237
Peppers	99.62209	33.45786



(a)

(b)

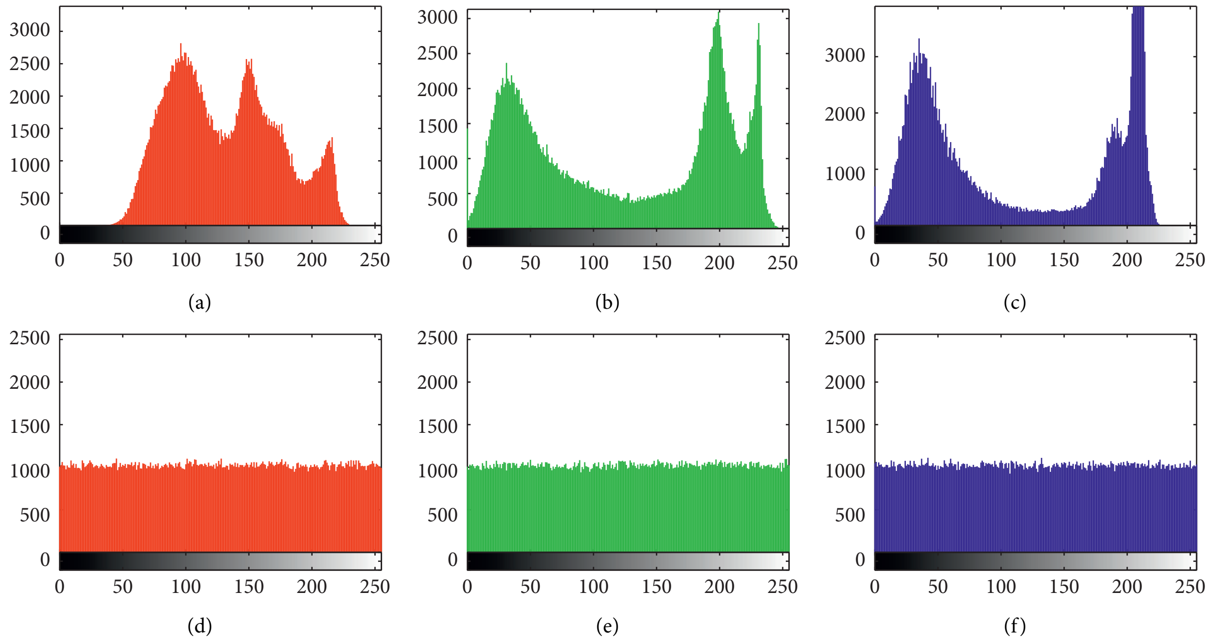
(c)

(d)

(e)

(f)

Figure 8: The histogram of the tested greyscale images, in which the encrypted images are totally having a uniform distribution. (a) Boats. (b) Bridge. (c) Baboon. (d) Enc-Boats. (e) Enc-Bridge. (f) Enc-Baboon.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 9: The histogram of Sailboat color image, in which the three channels of the cipher image are totally having a uniform distribution.

Table 6: χ^2 values of the experimented greyscale images.

Image	Chi-square value	Result
Boats	383969.687	Nonuniform
Bridge	1185618.347	Nonuniform
Baboon	187692.171	Nonuniform
Enc-boats	277.561	Uniform
Enc-bridge	263.324	Uniform
Enc-baboon	286.876	Uniform

Table 7: χ^2 values of the experimented color images.

Image	Chi-square value			Result
	R	G	B	
Sailboat	196697.306	130154.716	344571.537	Nonuniform
Airplane	678424.492	682495.382	1107858.005	Nonuniform
Peppers	213187.216	318382.929	491428.177	Nonuniform
Enc-sailboat	215.636	243.337	232.412	Uniform
Enc-airplane	204.193	269.281	287.061	Uniform
Enc-peppers	235.867	251.417	239.181	Uniform

Table 8: Global and local information entropies for the investigated dataset.

Image	Global entropy		Local entropy	
	Plain	Cipher	Plain	Cipher
Boats	7.19137	7.99923	6.10263	7.90249
Bridge	5.70556	7.99927	4.81525	7.90286
Baboon	7.35787	7.99918	6.66019	7.90322
Sailboat	7.76216	7.99976	6.07741	7.90136
Airplane	6.66391	7.99974	5.52864	7.90223
Peppers	7.66982	7.99976	6.04964	7.90145

Table 9: Contrast values of the experimented greyscale images.

Image	Original	Encrypted
Boats	0.37994	10.51092
Bridge	0.47895	10.49715
Baboon	0.61842	10.51323

Table 10: Contrast values of the experimented color images.

Image	Original			Encrypted		
	R	G	B	R	G	B
Sailboat	0.29432	0.48611	0.46158	10.48872	10.48024	10.50732
Airplane	0.18473	0.28502	0.13335	10.48935	10.52216	10.50711
Peppers	0.27514	0.30299	0.22137	10.50131	10.47429	10.53328

Table 11: PSNR and MSE values of experimented datasets.

Image	PSNR	MSE
Boats	9.29525	7.64812
Bridge	8.77761	8.61627
Baboon	8.64496	7.28999
Sailboat	7.87325	1.01174
Airplane	7.23595	1.03475
Peppers	7.44729	1.01103

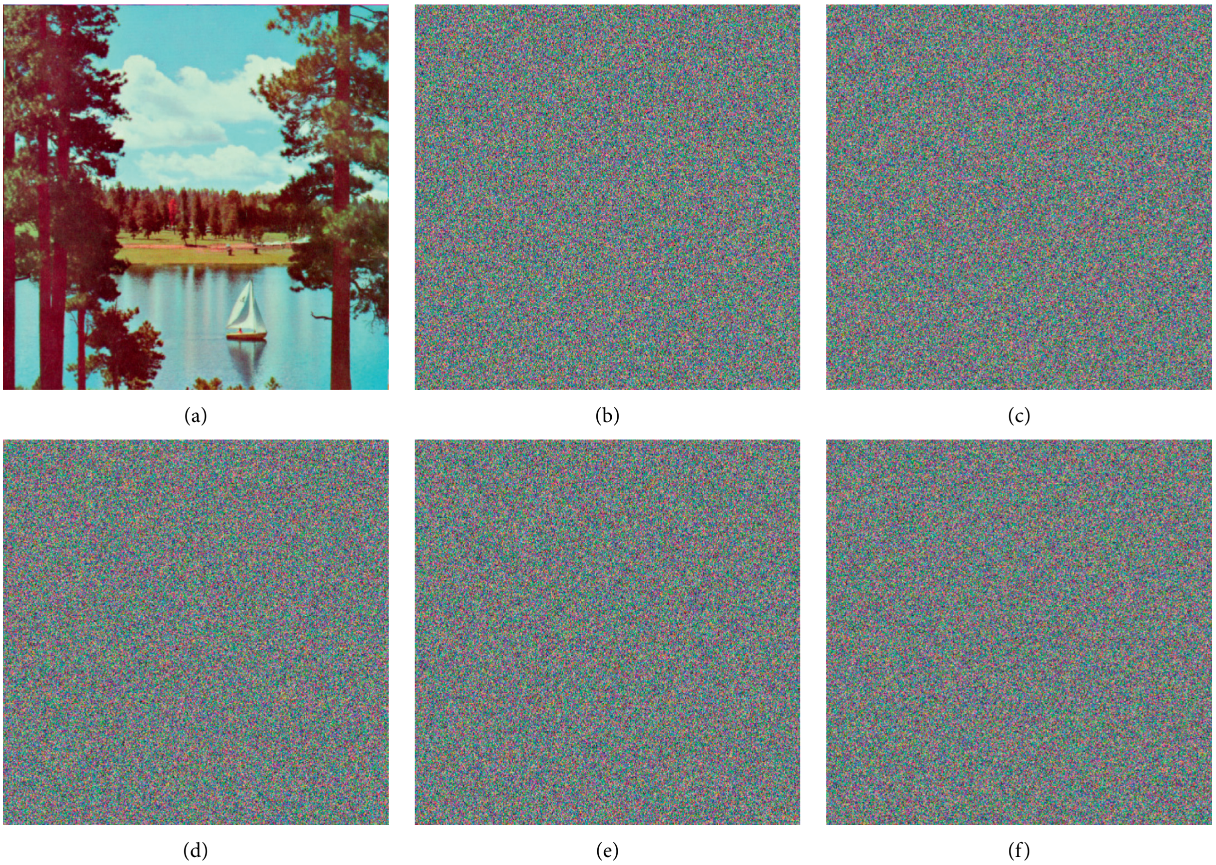


Figure 10: Key sensitivity of the presented encryption approach. (a) Correct key. (b) Correct key but $LC_0 = 0.684000000000001$. (c) Correct key but $\alpha = 3.356000000000001$. (d) Correct key but $A = 153$. (e) Correct key but $LS_0 = 0.479400000000001$. (f) Correct key but $\beta = 3.843500000000001$.

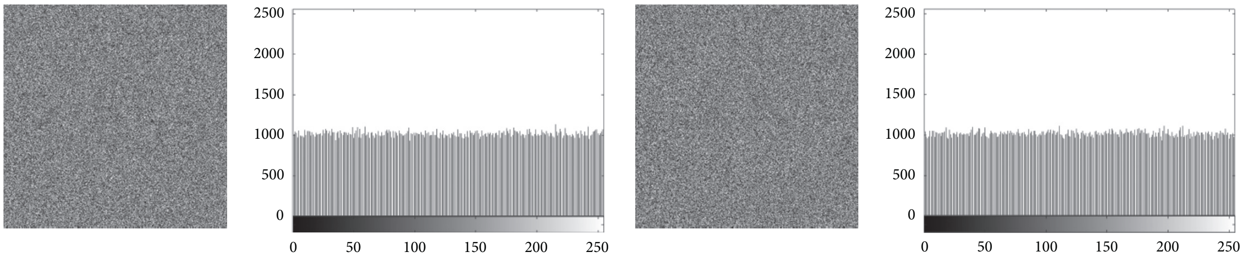
Table 12: NPCR and UACI of decrypted Sailboat image with the correct key and other decrypted Sailboat images with tiny modifications in the initial keys, as stated in Figure 10.

Image	NPCR (%)	UACI (%)
Figures 10(a) and 10(b)	99.604415	32.209367
Figures 10(a) and 10(c)	99.618912	32.198207
Figures 10(a) and 10(d)	99.613063	32.211601
Figures 10(a) and 10(e)	99.605052	32.217681
Figures 10(a) and 10(f)	99.600856	32.191464

is highly sensitive to the secret key ($LS_0, \beta.LC_0, \alpha$, and A). If there is any tiny change in one of the secret keys ($LS_0, \beta.LC_0, \alpha$, and A), then a significant variation is generated in the outcome. In addition, our cryptosystem employs the hash value of the plain image to update the initial key parameters; therefore, our cryptosystem depends not only on key parameters but also on the plain image. Cryptanalyst tries to obtain some valuable information regarding the secret key using full black and white images, due to their capability of disabling the role of permutation/substitution processes. The corresponding cipher images for black and white plain images and their corresponding histograms are given in Figure 11, in which no visual information can be obtained from these cipher images, and Table 13 provides some statistical analyses for these images. Consequently, our encryption approach has the ability to withstand the chosen-ciphertext and chosen-plaintext attacks.

5.11. Noise and Data Loss Attacks.

When data is transmitted over a communication channel, noise affects the information transmitted, and data may lose some of its parts. Subsequently, a well-designed encryption approach should have the capability of withstanding data loss and noise attacks. To assess the suggested cryptosystem against these attacks, we execute occlusion attacks by cutting out some parts of the cipher image or joining Salt & Pepper

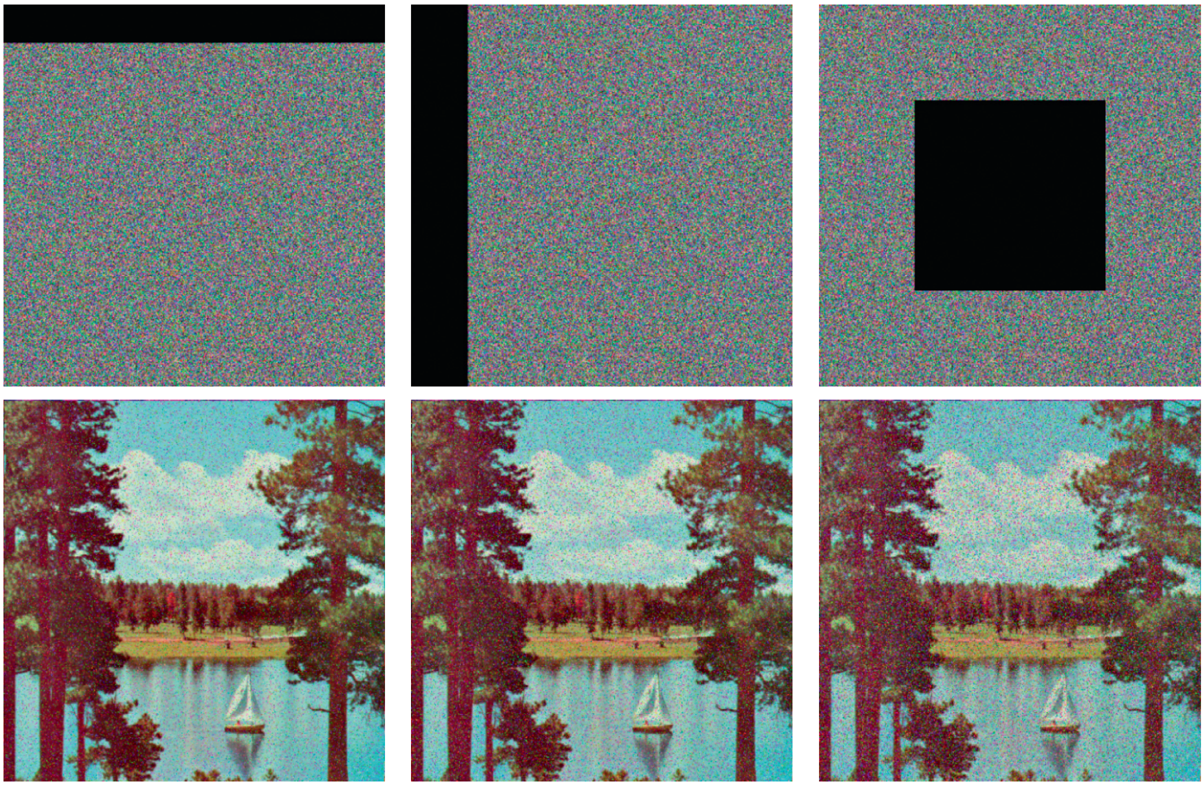


(a) (b) (c) (d)

Figure 11: Cipher images of full white and black images, and their corresponding histograms. (a) Enc-white. (b) Histogram of Enc-white. (c) Enc-black. (d) Histogram of Enc-black.

Table 13: Statistical examinations of the cipher full-white and full-black images.

Image	Chi value	Correlation			Entropy		Contrast
		Hor.	Ver.	Dia.	Global	Local	
Enc-white	279.8320	−0.0015	0.0004	0.0007	7.99923	7.9023	10.50954
Enc-black	280.1445	−0.0002	0.0001	0.0011	7.99922	7.9026	10.46016



(a) (b) (c)

Figure 12: Data loss attack, with the first row denoting the defective cipher images by cutting out some parts and the last row signifying the corresponding deciphered ones. (a) Cutting out 10%. (b) Cutting out 15%. (c) Cutting out 25%.

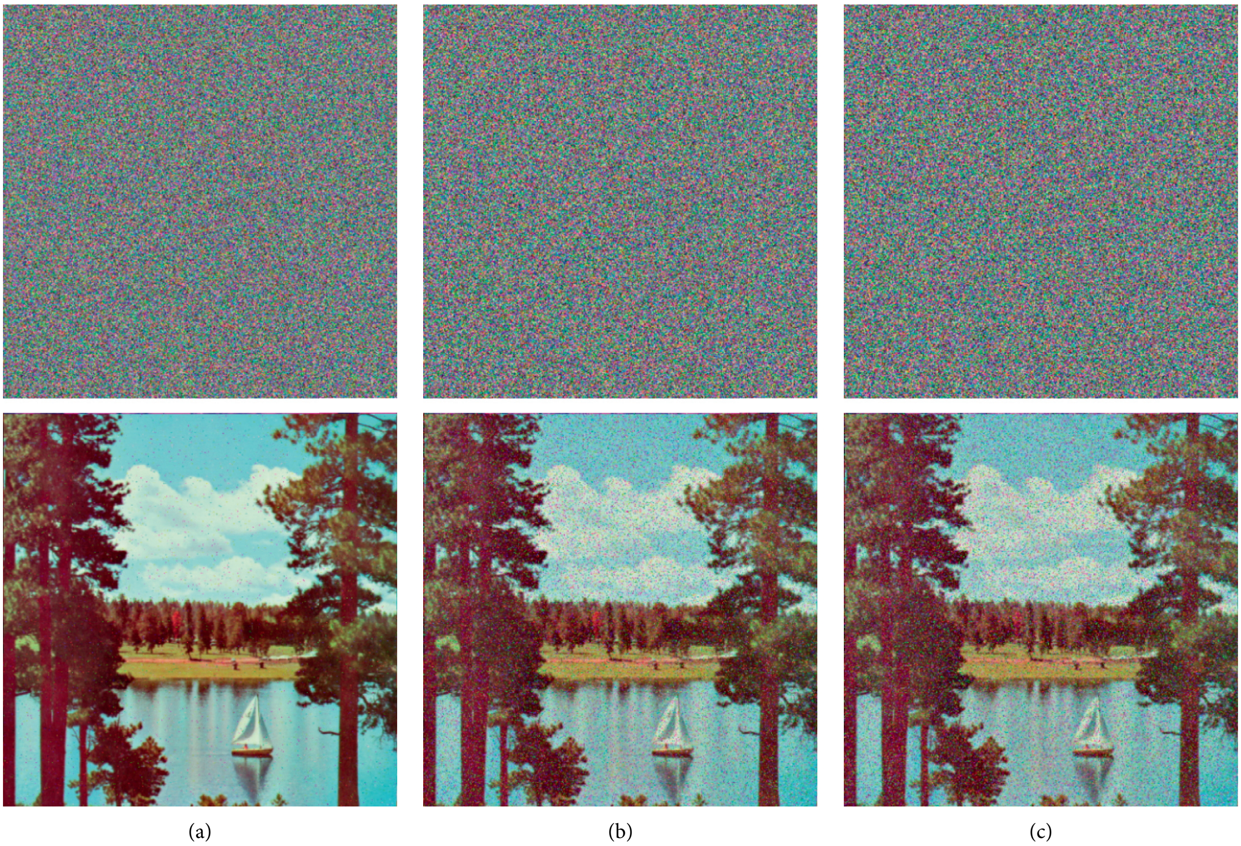


Figure 13: Noise attack, with the first row denoting the defective cipher images by varying Salt & Pepper noise density and the last row signifying the corresponding deciphered ones. (a) Density = 0.01. (b) Density = 0.15. (c) Density = 0.25.

Table 14: Comparison of our algorithm with other related cryptosystems in terms of average values of correlation, NPCR, UACI, local entropy, and global entropy.

Cryptosystem	Correlation			NPCR (%)	UACI (%)	Information entropy	
	Hor.	Ver.	Dia.			Global	Local
Our cryptosystem	−0.00016	0.00002	0.00023	99.62198	33.47181	7.99949	7.90227
Ref. [5]	0.00200	−0.00070	−0.00140	99.65000	33.48000	7.99700	—
Ref. [10]	0.00052	0.00033	0.00087	99.60960	33.45960	7.99930	7.90237
Ref. [11]	−0.00970	−0.00870	0.00650	99.60000	33.44000	7.99700	7.90217
Ref. [12]	−0.00074	0.00120	−0.00320	—	—	7.99830	—
Ref. [14]	0.00219	0.00169	0.00186	99.61100	33.47567	7.99929	7.90238
Ref. [25]	−0.00420	−0.00490	−0.00450	99.6101	33.52520	7.9995	7.90300
Ref. [26]	0.00180	−0.00161	0.00463	99.6225	33.59500	7.99301	—
Ref. [28]	0.00050	0.00170	−0.00250	99.60667	33.42667	7.99866	—

Table 15: Comparison of our cryptosystem with other related cryptosystems in terms of average values of Chi-square, contrast, and PSNR.

Cryptosystem	Chi-square	Contrast	PSNR
Our approach	250.51217	10.50209	8.21239
Ref. [11]	257.33667	—	—
Ref. [14]	249.42857	—	—
Ref. [25]	249.84440	—	—
Ref. [26]	—	10.43525	8.53790
Ref. [28]	256.75146	10.62060	8.41076

noise to it and then attempting to recover the secret image from the defective cipher image via the decryption procedure. Figures 12 and 13 show results of occlusion attacks, in which the original image is efficiently obtained after the decryption procedure.

5.12. Comparative Analysis. To confirm the effectiveness of our cryptosystem alongside other related approaches, Tables 14 and 15 present average values of correlation, NPCR, UACI, local information entropy, global information entropy, Chi-square, contrast, and PSNR of our cryptosystem with their average values reported in [5, 10–12, 14, 25, 26, 28]. The outcomes declared in Tables 2, 14, and 15 prove the effectiveness of the presented cryptosystem compared to other related approaches.

6. Conclusions

This paper has detailed a new cipher image mechanism for secure data transfer in cloud-based smart cities. The proposed encryption system is applicable to both color and greyscale images. The system is based on cascading two integrated 1D chaotic maps: Logistic-Chebyshev and Logistic-Sine. Logistic-Sine map is used to permute the plain image, and Logistic-Chebyshev map is used to substitute the permuted image, while the cascading of both integrated maps is used in performing XOR procedure on the substituted image. The experimental results of the suggested approach demonstrated the effectiveness of the presented cryptosystem. In the future, we aim to extend this work into designing a new visual cryptography mechanism for secure data transfer among Internet of Things devices.

Data Availability

The data used in the study can be accessed upon your request to the corresponding author.

Conflicts of Interest

The authors declare that there are no conflicts of interest for this paper and its contents.

Acknowledgments

This study was supported by Menoufia University.

References

[1] Q. Huang, L. Wang, and Y. Yang, “Secure and privacy-preserving data sharing and collaboration in mobile healthcare social networks of smart cities,” *Security and Communication Networks*, vol. 2017, Article ID 6426495, 2017.

[2] K. Zhang, J. Ni, K. Yang, X. Liang, J. Ren, and X. S. Shen, “Security and privacy in smart city applications: challenges and solutions,” *IEEE Communications Magazine*, vol. 55, no. 1, pp. 122–129, 2017.

[3] M. Amin and A. A. Abd El-Latif, “Efficient modified RC5 based on chaos adapted to image encryption,” *Journal of Electronic Imaging*, vol. 19, no. 1, Article ID 013012, 2010.

[4] A. Belazi, A. A. Abd El-Latif, R. Rhouma, and S. Belghith, “Selective image encryption scheme based on DWT, AES S-box and chaotic permutation,” in *2015 International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 606–610, IEEE, Dubrovnik, Croatia, August 2015.

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

[6] T. J. Zhang, I. M. Manhrawy, A. A. Abdo, A. A. Abd El-Latif, and R. Rhouma, “Cryptanalysis of elementary cellular automata based image encryption,” *Advanced Materials Research*, vol. 981, pp. 372–375, 2014.

[7] B. Abd-El-Atty, M. Amin, A. Abd-El-Latif, H. Ugail, and I. Mehmood, “An efficient cryptosystem based on the Logistic-Chebyshev map,” in *Proceedings of the 2019 13th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, pp. 1–6, IEEE, Island of Ulkulhas, Maldives, August 2019.

[8] K. M. Ali and M. Khan, “Application based construction and optimization of substitution boxes over 2D mixed chaotic maps,” *International Journal of Theoretical Physics*, vol. 58, no. 9, pp. 3091–3117, 2019.

[9] H. M. Waseem and M. Khan, “A new approach to digital content privacy using quantum spin and finite-state machine,” *Applied Physics B*, vol. 125, no. 2, p. 27, 2019.

[10] Y. Xian and X. Wang, “Fractal sorting matrix and its application on chaotic image encryption,” *Information Sciences*, vol. 547, pp. 1154–1169, 2021.

[11] Z.-H. Gan, X.-L. Chai, D.-J. Han, and Y.-R. Chen, “A chaotic image encryption algorithm based on 3-D bit-plane permutation,” *Neural Computing and Applications*, vol. 31, no. 11, pp. 7111–7130, 2019.

[12] X. Chai, J. Bi, Z. Gan, X. Liu, Y. Zhang, and Y. Chen, “Color image compression and encryption scheme based on compressive sensing and double random encryption strategy,” *Signal Processing*, vol. 176, Article ID 107684, 2020.

[13] X. Chai, H. Wu, Z. Gan, Y. Zhang, and Y. Chen, “Hiding cipher-images generated by 2-D compressive sensing with a multi-embedding strategy,” *Signal Processing*, vol. 171, Article ID 107525, 2020.

[14] X. Wang, L. Feng, and H. Zhao, “Fast image encryption algorithm based on parallel computing system,” *Information Sciences*, vol. 486, pp. 340–358, 2019.

[15] Z. Hua, Y. Zhou, C.-M. Pun, and C. L. P. Chen, “2D Sine logistic modulation map for image encryption,” *Information Sciences*, vol. 297, pp. 80–94, 2015.

[16] Y. Zhou, L. Bao, and C. L. P. Chen, “A new 1D chaotic system for image encryption,” *Signal Processing*, vol. 97, pp. 172–182, 2014.

[17] L. Li, B. Abd-El-Atty, A. A. Abd El-Latif, and A. Ghoneim, “Quantum color image encryption based on multiple discrete chaotic systems,” in *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 555–559, IEEE, Prague, Czech Republic, September 2017.

[18] Y. Zhou, Z. Hua, C. M. Pun, and C. P. Chen, “Cascade chaotic system with applications,” *IEEE Transactions on Cybernetics*, vol. 45, no. 9, pp. 2001–2012, 2014.

[19] S. Shahzadi, B. Khaliq, M. Rizwan, and F. Ahmad, “Security of cloud computing using adaptive neural Fuzzy inference system,” *Security and Communication Networks*, vol. 2020, Article ID 5352108, 2020.

[20] I. A. Awan, M. Shiraz, M. U. Hashmi, Q. Shaheen, R. Akhtar, and A. Ditta, "Secure framework enhancing AES algorithm in cloud computing," *Security and Communication Networks*, vol. 2020, Article ID 8863345, 2020.

[21] 2020 http://sipi.usc.edu/database/database.php?volume=miscSIPI image database–Misc.

[22] A. A. A. EL-Latif, B. Abd-El-Atty, and S. E. Venegas-Andraca, "Controlled alternate quantum walk-based pseudo-random number generator and its application to quantum color image encryption," *Physica A: Statistical Mechanics and Its Applications*, vol. 547, Article ID 123869, 2020.

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

[24] B. Abd-El-Atty, A. A. Abd El-Latif, and S. E. Venegas-Andraca, "An encryption protocol for NEQR images based on one-particle quantum walks on a circle," *Quantum Information Processing*, vol. 18, no. 9, p. 272, 2019.

[25] 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.

[26] J. Ahmad and S. O. Hwang, "A secure image encryption scheme based on chaotic maps and affine transformation," *Multimedia Tools and Applications*, vol. 75, no. 21, pp. 13951–13976, 2016.

[27] B. Abd-El-Atty, A. M. Iliyasu, H. Alaskar, A. A. Abd El-Latif, and A. Ahmed, "A robust quasi-quantum walks-based steganography protocol for secure transmission of images on cloud-based E-healthcare platforms," *Sensors*, vol. 20, no. 11, p. 3108, 2020.

[28] S. Askar, A. Karawia, A. Al-Khedhairi, and F. Al-Ammar, "An algorithm of image encryption using logistic and two-dimensional chaotic economic maps," *Entropy*, vol. 21, no. 1, p. 44, 2019.