

Received August 13, 2020, accepted August 18, 2020, date of publication August 24, 2020, date of current version September 3, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3019216

# Robust Image Encryption With Scanning Technology, the El-Gamal Algorithm and Chaos Theory

SURA F. YOUSIF<sup>1</sup>, (Member, IEEE), ALI J. ABBOUD<sup>2</sup>, AND HUSSEIN Y. RADHI<sup>2</sup>

<sup>1</sup>Department of Chemical Engineering, College of Engineering, University of Diyala, Diyala 00964, Iraq

<sup>2</sup>Department of Computer Engineering, College of Engineering, University of Diyala, Diyala 00964, Iraq

Corresponding author: Ali J. Abboud (ali.j.abboud@gmail.com)

**ABSTRACT** Digital images are the most frequently used signals to convey information in the internet era. The security of these images is the primary concern in rapidly changing networked environments. In this research, we present a novel approach to secure images by integrating a scanning technique, the El-Gamal public key cryptosystem and chaotic systems. In brief, zigzag and spiral scanning are used first to construct a permuted image. Then, the El-Gamal encryption algorithm is exploited to encrypt the permuted image. Finally, Lorenz and Rössler chaotic sequences are utilized to scramble the pixel locations in the confusion and diffusion stages. This last step that mixes two stages can fortify the entire security performance and enlarge the key size. Exhaustive analysis has been carried out on the SIPI (signal and image processing institute) dataset to assess the efficiency and security of the proposed method. Numerical and visual results indicate the capability of the proposed image cryptosystem to protect images against several known attacks. In addition, the comparative analysis results indicate that the proposed approach outperforms the compared approaches in terms of the visual quality and security criteria.


**INDEX TERMS** Image security, scanning, chaotic, Lorenz and Rössler sequences, El-Gamal algorithm.

## I. INTRODUCTION

Digital images are widely utilized in a wide spectrum of applicable services. The images are transmitted through public communication channels and the internet [1]. However, exchanging images in this way increases the possibility of threats by unauthorized persons changing or stealing these images [2]. The most effective tool to prevent such adversary threats to digital images is cryptographic encryption/decryption algorithms [3], [4]. These tools are used to scramble images during storage, transmission and processing. Symmetric and asymmetric cryptosystems are the two basic types of these algorithms. Symmetric encryption schemes are also called shared key cryptosystems, since they need only a single key to do encryption and decryption operations. The symmetric algorithms are classified as stream and block ciphers. The RC4 algorithm is an example of a stream cipher, whilst the Data Encryption Standard (DES) and Advanced Encryption Standard (AES) algorithms are examples of block ciphers [5], [6]. On the other hand,

asymmetric cryptosystems have two different keys that are applied in the encryption and decryption phases. The first key is the encryption key (or public key), which is utilized by the sender to encrypt plain images, while the decryption key (or private key) is utilized by the recipient to decrypt scrambled images. Examples of asymmetric cryptography are the El-Gamal and RSA algorithms. Adopting earlier encryption algorithms has a large impact on the computation time and power because these algorithms need complex processing operations to perform permutation, substitution and key scheduling. Hence, it is necessary to develop hybrid security algorithms that combine encryption algorithms and biometrics to obtain more secure and efficient algorithms [7], [8].

Recently, chaotic systems have been used heavily to develop robust cryptographic algorithms [9]. These systems have proved their ability to create very strong defences against different kinds of attacks. In addition, the systems provide a good balance of the efficiency, security and speed, which makes them the best candidate to secure digital images. Chaotic systems possess several interesting characteristics, including erratic behaviour, boundedness, deterministic nature, high sensibility to system parameters/initial

The associate editor coordinating the review of this manuscript and approving it for publication was Md. Asikuzzaman .

values, non-periodicity, transitivity, simplicity and pseudo randomness. All these nonlinear properties make chaotic systems effective methods to protect digital images in various kinds of environments. However, applying chaotic systems in the cryptographic domain faces some difficulties. For example, some cryptosystems employ low dimensional chaotic maps in their security algorithms. These maps are characterized by their simple construction and high velocity of producing chaotic series. On the other hand, low dimensionality maps possess short periodicity, which degrades their dynamic chaotic characteristics within the finite precision execution of a computer. Consequently, the opponent can easily obtain the secret keys of the chaotic system. Furthermore, chaotic systems with low dimensionality possess small key sizes, which results in weakened security. Hence, adopting chaotic systems with high dimensionality significantly improves the security by increasing the nonlinearity. Hence, this kind of chaotic system has been widely utilized in image security [10]–[12].

Different cryptosystems have been proposed for image security. The researchers in [13] utilized a phase retrieval technology, the fractional Mellin algorithm, to protect digital images. The strength of this approach is using spatial filters in the Fourier transform domain, namely, radial Hilbert and toroidal plate masks. In addition, the author in [14] presented a colour image security scheme based on chaotic systems. High randomness in the data and a large key space for this scheme are obtained by applying chaotic systems with multiple parameters. In addition, other researchers [15] partitioned digital images into four sections by adopting the quaternary principle in order to encode every section individually. Then, the DNA process and chaotic Chen system are sequentially applied to drastically modify the input image information. In [16], the authors proposed a secure scheme that relies upon chaotic sequence and comprehensive sensing technology. A greyscale image is shuffled via zigzag coding after converting it using the wavelet transform. Then, the image is compressed and encrypted via a skew tent map and embedded inside the transported image. In [17], they compressed and encrypted an input colour image by executing an enhanced cat map. After that step, the image was encrypted using the El-Gamal cryptography algorithm and diffused by the 3D Lorenz system. Another hybrid approach [18] is proposed by executing the rules and operations of the DNA algorithm and the Henon-Sine chaotic map to secure digital images that were sent throughout the internet. Furthermore, other authors [19] introduced a new three-dimensional chaotic map by merging a logistic map with a piecewise map. Then, this new chaotic map is utilized to encrypt colour images. Additionally, the researchers in [20] merged compression and cryptographic techniques by utilizing chaotic maps, the Kd-tree and hierarchical wavelet trees. Moreover, there is an interesting approach in [21] that adopted the zigzag scan methodology and cellular automata to confuse the coefficients of a greyscale image. Then, a chaotic system is applied to perceive and compress the shuffled image to obtain a final

secure image. In another direction [22], the hash value of an image is used as the seed value for Rossler and piece-wise linear chaos systems. Then, chaotic systems are employed to shuffle and diffuse the pixels of an image. [23] Introduced a novel approach to encrypt and shuffle a colour image by applying a new 1D chaotic map called the coupled sine map. Furthermore, [24] suggested an encryption approach that utilized scrambling and diffusion operations. The scrambling process is achieved by integrating Tent and staged Logistic maps combined with a composite map while the diffusion process is achieved by applying a Hopfield neural network. A fast image encryption mechanism is presented in [25] that is based on a new chaotic lattice model. The new chaotic map in this scheme is exploited to generate a secret key to scramble and diffuse the image pixels. The final image cipher has a good ability to resist various kinds of attacks. In another paper, a colour image is divided into  $(4 \times 4)$  parts, and then each part is further divided into  $(16 \times 16)$  blocks [26]. Secret keys are generated in this technique by employing a new 3D chaotic map, and then these keys are permuted with the blocks to obtain the encrypted image. In [27], a five-dimensional hyper chaotic system is implemented to produce pseudo random sequences. The initial values of the algorithm are generated via secret keys and a hash function. Image pixels are swapped to perform the permutation process whereas the diffusion process is performed by adopting cyclic shift technology. In addition, the researchers in [28] proposed an approach to join the spiral scan, the random partition of the overlapped blocks of an image and (Henon and Lü) chaotic systems to secure an image and then spiral scanning was adopted for pixel scrambling after splitting the image into overlapped blocks. After that step, an XOR process between the scrambled image elements and a private matrix generated from a Lü map is calculated to create the encrypted image. Moreover, there are other approaches that use the device hardware capabilities to design an efficient chaotic systems. For instance, the authors in [29] proposed a novel chaotic sequences generator based on large precision arithmetic. This approach is to encrypt digital images by using chaos cryptosystems leads to improve the system security and also increases the key space size dramatically. Another research by the same authors in the similar direction [30] is used to enhance the randomness of 5D chaotic system by utilizing PIC microcontroller. The generated chaotic map is the main tool to encrypt grayscale digital images sent through wireless telecommunication channels. Also, there is an attractive approach in [31] that integrates a DNA encoding scheme, hyper chaotic maps and dynamic filtering to protect digital images. The objective of all earlier mentioned approaches is to produce unrecognizable images. However, the efficiency and security of these approaches need to be carefully considered. Relying on this assumption, a new image security scheme to encrypt grey and colour images is proposed. This scheme is based on using scanning technology, the El-Gamal public key cryptosystem and chaos theory. The major contributions of this work are summarized as follows: (1) utilizing

scan technique, El-Gamal algorithm and chaotic systems to build a powerful color image encryption mechanism (2) the usage of image scanning technique into two different directions has improved significantly the scrambling process of input image pixels. (3) Precise usage of high dimensional chaotic maps to increase the key space size and to enhance image security method. (4) confusion and diffusion stages are merged into one stage to resist several known attacks.

The remainder of this paper is organized as follows. The background is described in Section II, the proposed methodology is explained in Section III, the image security simulation results are analysed in Section IV, the image security analysis is illustrated in Section V, and the comparative analysis results are presented in Section VI. Finally, Section VII is devoted to the conclusion.

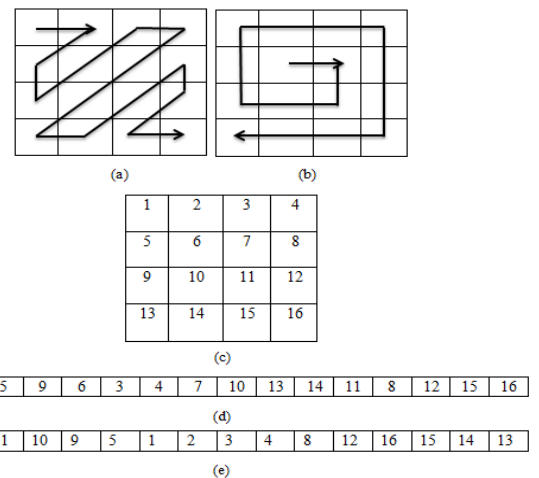
**II. BACKGROUND**

In this section, we elaborate the main components of our approach that consist of scanning technology, the El-Gamal public key cryptosystem and chaos theory as follows.

**A. SCANNING ARRANGMENTS**

In general, all naturalistic images are characterized by possessing the robust relationships among their adjacent pixels such that the prediction of any pixel value may be reasonably easy from its neighbouring pixel values. To improve the entropy value and reduce the tight relations among image pixels, scan methodology is adopted to shuffle the pixel positions in the input image. A scan word refers to various scanning methods for two dimensional images. For an  $(n \times n)$  image, the number of generated scan pathways is  $(n \times n)!$  when utilizing the scanning technique. The scan pathway for an ordinary image merely represents an order such that every pixel in the image is visited exactly once. The encryption process also determines the proper scanning pathways. The set of scan pathways in the encryption operation should be kept confidential because they form the cryptographic key. The achieved security level is high, and it is difficult to extract the secret key by employing existing calculation techniques. Scan technology is used in various applications such as data hiding, compression and encryption [32].

Zigzag and spiral scanning are the most popular methods to convert a two-dimensional matrix-like  $(m, n)$  image into a  $(1, m \times n)$  one dimensional vector. By utilizing these two technologies, the relationships among adjacent pixels in the input image are demolished. The adjacent pixels in the two-dimensional image are very dispersed and detached into the one-dimensional vector [32], [33]. Zigzag and spiral pathways are employed in this approach to scramble the locations of the pixels in the plain image. Using this method, the powerful relationships among image pixels can be disturbed, which results in a stronger cryptographic algorithm. For the zigzag pathway, the starting point is the pixel at the position of  $(1, 1)$ , which means that the navigation begins from the first pixel and proceeds in the same direction until it finishes at the last pixel in the lower right. For a spiral pathway, the starting



**FIGURE 1.** Zigzag and spiral pathways for the shuffling process: (a) Zigzag path with the starting pixel at  $(1, 1)$ , (b) Spiral path with the starting pixel at  $(2, 2)$ , (c) Original matrix, (d) generated vector from (a), and (e) generated vector from (b).

point is the pixel at the position of  $(2, 2)$ , which indicates that the traversal starts from that pixel and moves in the same direction until it ends at the last pixel in the lower left. An example of a  $4 \times 4$  matrix is shown in Fig. 1 (c) and the generated vectors by employing zigzag and spiral pathways (Fig. 1 (a) and Fig. 1 (b), respectively) are clarified in Fig. 1 (d) and Fig. 1 (e), respectively. Different starting points in the matrix can give different permutation impacts. Therefore, applying zigzag and spiral scans with these two different starting points can ameliorate the degree of scrambling degree, which consequently increases the algorithm’s security.

**B. EI-GAMAL CRYPTOGRAPHIC ALGORITHM**

El-Gamal is an asymmetric key cryptosystem invented in 1985 by Taher El-Gamal. This algorithm represents an alternate method for the RSA public key cipher [34]. The major difference between the El-Gamal and RSA algorithms is that RSA security relies upon the difficulty of factorizing large prime numbers whereas El-Gamal security relies on the difficulty of calculating the discrete logarithm modulus of large prime numbers. The discrete logarithm issue is a notably difficult problem in mathematics because it depends mainly on conjecture to obtain all the potential solutions for it. Thus, breaking this cryptographic system is almost unattainable or it requires an extremely long time. The main advantage of the El-Gamal technique is that the same plaintext message results in a different ciphertext message every time it is encrypted.

Key production, encryption and decryption are the three phases of the El-Gamal algorithm that are briefly described as follows.

**C. KEY PRODUCTION**

The key production procedure in the El-Gamal algorithm is described below [34].

1. An integer number  $g$  and a prime number  $p$  are chosen randomly such that  $g$  is a root of  $p - 1$ .
2. Another random number  $x$  is selected randomly from the interval  $[1, \dots, p-2]$ , where  $x$  represents the secret key.
3.  $y$  is calculated as follows:  $y = g^x \text{ mod } p$ , where  $y$  denotes to the public key.

The confidential key of the El-Gamal cryptosystem is represented by  $x$  while the public key involves the triple ( $p, g$  and  $y$ ).

#### D. ENCRYPTION PHASE

The encryption procedure in the El-Gamal algorithm is described below [34].

1. The public key ( $p, g$  and  $y$ ) is obtained from the transmitter.
2. A random integer number  $k$  is selected, where  $k$  is in the interval  $[1, \dots, p - 2]$ .
3. A cipher text message composed of the pair  $(c_1, c_2)$  is computed as follows:  $c_1 = g^k \text{ mod } p$ ,  $c_2 = m \times y^k \text{ mod } p$ , where  $m$  symbolizes that the secret message needs to be encrypted. Then,  $(c_1, c_2)$  is transferred to the receiver.

#### E. DECRYPTION PHASE

The encrypted message  $(c_1, c_2)$  along with the secret key  $x$  and the prime number  $p$  are utilized to retrieve the plain message  $m$  by computing the following:  $m = c_2 / (c_1)^x \text{ mod } p$  [34].

#### F. CHAOTIC SYSTEMS

Chaos phenomenon is a clear reality that appears in dynamical nonlinear systems. Chaotic systems are characterized by possessing complicated structures and are highly sensitive to the initial and control parameters. Therefore, it is natural to upgrade image security technology via chaotic ciphers. Within the construction of the presented security approach, the Lorenz system and Rössler system are employed to secure digital images. The description of these systems is given below.

#### G. LORENZ SYSTEM

This system is one of the most famous chaotic systems that was established by the scientist Lorenz in 1963. The Lorenz system is described via three-dimensional independent equations:

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

where  $a, b$  and  $c$  denote the control parameters and  $x_0, y_0$  and  $z_0$  represent the initial states. To be in a chaotic state, the typical values of the system parameters should be 10, 8/3, and 28, respectively, for  $a, b$  and  $c$  [35].

#### H. RÖSSLER SYSTEM

A Rössler system is a three-dimensional chaotic generator that was designed by Rössler in 1970. The simple state equations of Rössler are given as follows:

$$\begin{aligned} \dot{x} &= -y - z \\ \dot{y} &= x + ay \\ \dot{z} &= b + z(x-c) \end{aligned} \quad (2)$$

In the above equations,  $x_0, y_0$  and  $z_0$  are the initial seeds; and  $a, b$  and  $c$  are the system constants. Rössler is considered a nonlinear ergodic system when the values of  $a, b$  and  $c$  are set as 0.2, 0.2 and 5.7, respectively [36].

#### III. PROPOSED METHODOLOGY

We elaborate our proposed methodology to secure digital images in this section. It is composed of scan technologies, the El-Gamal public key algorithm and chaotic systems. There are three phases in this approach, which are image partitioning, image scanning (or scrambling) and image encryption/decryption. In the first phase, the image is vertically partitioned into two equal blocks. Next, in the second phase, the pixels in the left image block are scrambled by adopting the zigzag pathway (Fig. 1 (a)) whereas the pixels in the right image block are scrambled by adopting the spiral pathway (Fig. 1 (b)). After that step, the resultant vectors from these two pathways are merged into one matrix to construct a newly modified image (the details of this matrix will be explained later).

Finally, the El-Gamal algorithm is applied to the modified image in order to produce a secure digital image. To improve the system security and increase the entropy (by minimizing the correlation among image pixels), chaotic systems are exploited to execute primordial cryptographic processes (i.e., confusion and diffusion). A three dimensional Lorenz chaotic system is applied on the encrypted image obtained from the third phase to achieve the confusion operation by randomly varying the pixel locations. After that step, a three-dimensional Rössler system is employed to conduct the diffusion operation by changing the image pixel values to eventually obtain the ciphered image. In addition, we have to mention that the secret keys consist of information about constant parameters, the initial states of the chaotic systems and the secret keys of the El-Gamal cryptosystem. The block diagram of the proposed approach is illustrated in Fig. 2.

#### A. ENCRYPTION PROCEDURE FOR GRAYSCALE IMAGE

The details of the grayscale image encryption operation are detailed below as follows.

**Step1:** The original image  $A(m, m)$  is vertically separated into two equal blocks to get  $A_1(m, m/2)$  and  $A_2(m, m/2)$ .

**Step2:** The pixel arrangement within the left image block  $A_1(m, m/2)$  is shuffled using zigzag scanning (Fig. 1 (a)) to get the resultant vector  $B_1(1, m \times m/2)$ .



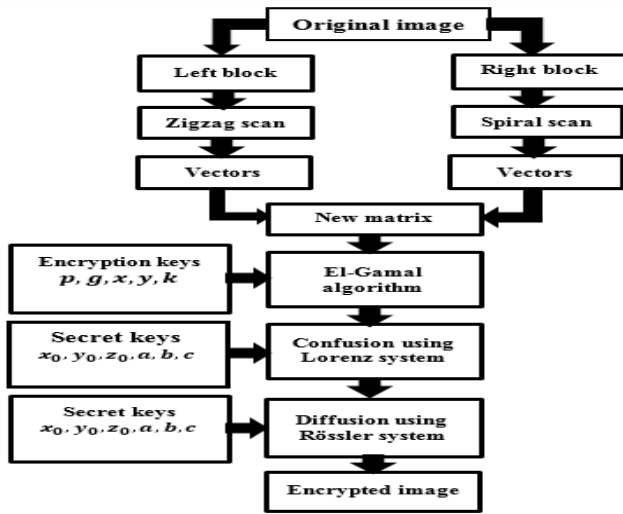


FIGURE 2. Block diagram of the proposed approach.

**Step3:** The pixel arrangement within the right image block  $A_2(m, m/2)$  is shuffled using spiral scanning (Fig. 1 (b)) to get the resultant vector  $B_2(1, m \times m/2)$ .

**Step4:** The obtained vectors  $B_1(1, m \times m/2)$  and  $B_2(1, m \times m/2)$  are reshaped to the same size to acquire  $C_1(m, m/2)$  and  $C_2(m, m/2)$ , respectively.

**Step5:** To increase the permutation degree,  $C_1(m, m/2)$  and  $C_2(m, m/2)$  are integrated into one matrix as follows. Suppose that  $C_1$  and  $C_2$  are the matrices shown in Fig. 3 (a) and Fig. 3 (b) respectively, each with a size of  $(8 \times 4)$ . To form matrix  $D$  in Fig. 3 (c), the first column of  $C_1$  represents the first column of  $D$ , the first column of  $C_2$  represents the second column of  $D$  and so on. In other words, the columns of  $C_1$  represent the odd columns in  $D(1, 3, 5, \dots)$  and the columns of  $C_2$  represent the even columns in  $D(2, 4, 6, \dots)$ . After that, matrix  $D$  is flipped from bottom to top to get the matrix  $D_1(8, 8)$ . This matrix ( $D_1$ ) corresponds to the newly modified matrix  $D_1(m, m)$ .

**Step6:** The El-Gamal algorithm is applied to the image matrix  $D_1$  after setting the parameters  $p, g, y,$  and  $k$  to get the encrypted pair  $(E_1, E_2)$  as follows:

$$E_1 = g^k \text{ mod } p \quad (3)$$

$$E_2(i, j) = D_1(i, j) \times y^k \text{ mod } p \quad (4)$$

**Step7:** Three chaotic sequences  $x, y$  and  $z$  are created via a Lorenz system according to (1) by utilizing its initial and system parameters values, where each sequence has a size of  $(1, m \times m)$ .

**Step8:** Reshape  $x, y$  and  $z$  with the same size of  $E_2$  to acquire  $x_1, y_1$  and  $z_1$ , respectively. Then,  $x_1$  is used to scramble  $E_2$ 's pixels as follows:

$$[l_x, d_x] = \text{sort}(x_1) \quad (5)$$

1	9	17	25
2	10	18	26
3	11	19	27
4	12	20	28
5	13	21	29
6	14	22	30
7	15	23	31
8	16	24	32

(a)

33	41	49	57
34	42	50	58
35	43	51	59
36	44	52	60
37	45	53	61
38	46	54	62
39	47	55	63
40	48	56	64

(b)

1	33	9	41	17	49	25	57
2	34	10	42	18	50	26	58
3	35	11	43	19	51	27	59
4	36	12	44	20	52	28	60
5	37	13	45	21	53	29	61
6	38	14	46	22	54	30	62
7	39	15	47	23	55	31	63
8	40	16	48	24	56	32	64

(c)

8	40	16	48	24	56	32	64
7	39	15	47	23	55	31	63
6	38	14	46	22	54	30	62
5	37	13	45	21	53	29	61
4	36	12	44	20	52	28	60
3	35	11	43	19	51	27	59
2	34	10	42	18	50	26	58
1	33	9	41	17	49	25	57

(d)

FIGURE 3. Details of the new modified image: (a) matrix  $C_1$ , (b) matrix  $C_2$ , (c) matrix  $D$  (d) modified matrix  $D_1$ .

$$F(i, j) = E_2(l_x(i), l_x(j)) \quad (6)$$

The random numbers produced by  $x_1$  are sorted in ascending order using (5).  $d_x$  symbolizes the new sequence generated after sorting  $x_1$  in ascending order while  $l_x$  denotes to index value of  $d_x$ . Based on the index  $l_x$ , the pixels in  $E_2$  are permuted in rows ( $i$ ) and columns ( $j$ ) according to (6) to get the confused image  $F(m, m)$ .

**Step9:** Three chaotic sequences  $x', y'$  and  $z'$  are created via the Rössler system according to (2) by utilizing its initial and system parameters values, and each sequence has a size of  $(1, m \times m)$ .

**Step10:** Reshape  $x', y'$  and  $z'$  to the same size of  $F$  to acquire  $x'_1, y'_1$  and  $z'_1$ , respectively. Then,  $x'_1$  is used to diffuse the confused matrix  $F$ 's pixels by applying the bitwise XOR process as follows:

$$G(i, j) = \text{bitxor}(G(i, j - 1), F(i, j), x'_1(i, j)) \quad (7)$$

where  $G$  represents the diffused image or the eventual encrypted image.

## B. ENCRYPTION PROCEDURE FOR COLOUR IMAGE

The details of the colour image encryption operation are given below as follows.

**Step1:** The original image  $P(m, m \times 3)$  is divided into three channels as  $R(m, m), G(m, m)$  and  $B(m, m)$ , respectively.

**Step2:** The first channel  $R(m, m)$  is vertically separated into two equal blocks to get  $R_1(m, m/2)$  and  $R_2(m, m/2)$ .

**Step3:** Repeat Step 2 on the second channel  $G(m, m)$  to get the matrices  $G_1(m, m/2)$  and  $G_2(m, m/2)$ .

**Step4:** Repeat Step 2 on the third channel  $B(m, m)$  to get the matrices  $B_1(m, m/2)$  and  $B_2(m, m/2)$ .

**Step5:** The pixel arrangement within the left image block  $R_1(m, m/2)$  is shuffled using zigzag scanning (Fig. 1 (a)) to get the resultant vector  $S_1(1, m \times m/2)$ .

- Step 6:** The pixel arrangement within the right image block  $R_2(m, m/2)$  is shuffled using spiral scanning (Fig. 1 (b)) to get the resultant vector  $S_2(1, m \times m/2)$ .
- Step 7:** Repeat Steps 5 and 6 for matrices  $G_1(m, m/2)$  and  $G_2(m, m/2)$ , respectively, to get the resultant vectors  $H_1(1, m \times m/2)$  and  $H_2(1, m \times m/2)$ , respectively.
- Step 8:** Repeat Steps 5 and 6 for matrices  $B_1(m, m/2)$  and  $B_2(m, m/2)$ , respectively, to get the resultant vectors  $T_1(1, m \times m/2)$  and  $T_2(1, m \times m/2)$ , respectively.
- Step 9:** The obtained vectors  $S_1(1, m \times m/2)$  and  $S_2(1, m \times m/2)$  are reshaped to the same size to acquire  $K_1(m, m/2)$  and  $K_2(m, m/2)$ , respectively.
- Step 10:** Repeat Step 9 on the obtained vectors  $H_1(1, m \times m/2)$  and  $H_2(1, m \times m/2)$  to acquire  $L_1(m, m/2)$  and  $L_2(m, m/2)$ .
- Step 11:** Repeat Step 9 for the obtained vectors  $T_1(1, m \times m/2)$  and  $T_2(1, m \times m/2)$  to acquire  $J_1(m, m/2)$  and  $J_2(m, m/2)$ , respectively.
- Step 12:** Repeat Step 5 from Section A for the matrices  $[K_1(m, m/2)$  and  $K_2(m, m/2)]$ ,  $[L_1(m, m/2)$  and  $L_2(m, m/2)]$ , and  $[J_1(m, m/2)$  and  $J_2(m, m/2)]$  to get the newly modified matrices  $D_1(m, m)$ ,  $D_2(m, m)$ , and  $D_3(m, m)$ , respectively.
- Step 13:** The El-Gamal algorithm is applied to matrices  $D_1$ ,  $D_2$  and  $D_3$ , respectively, after setting the parameters  $p$ ,  $g$ ,  $y$ , and  $k$  to get the encrypted pairs  $(Q_1, Q_2)$ ,  $(Q_1, Q_3)$  and  $(Q_1, Q_4)$ , respectively, as follows:

$$Q_1 = g^k \text{ mod } p \quad (8)$$

$$Q_2(i, j) = D_1(i, j) \times y^k \text{ mod } p \quad (9)$$

$$Q_3(i, j) = D_2(i, j) \times y^k \text{ mod } p \quad (10)$$

$$Q_4(i, j) = D_3(i, j) \times y^k \text{ mod } p \quad (11)$$

- Step 14:** Three chaotic sequences  $X$ ,  $Y$  and  $Z$  are created via the Lorenz system according to (1) by utilizing the initial and system parameter values, and each sequence has a size of  $(1, m \times m)$ .
- Step 15:** Reshape  $X$ ,  $Y$  and  $Z$  to the same size of  $Q_2$  to acquire  $X_1$ ,  $Y_1$  and  $Z_1$ , respectively. Then,  $X_1$  is used to scramble the pixels in  $Q_2$ ,  $Q_3$  and  $Q_4$  as follows:

$$[l_x, d_x] = \text{sort}(X_1) \quad (12)$$

$$F_1(i, j) = Q_2(l_x(i), l_x(j)) \quad (13)$$

$$F_2(i, j) = Q_3(l_x(i), l_x(j)) \quad (14)$$

$$F_3(i, j) = Q_4(l_x(i), l_x(j)) \quad (15)$$

The pixels in  $Q_2$ ,  $Q_3$  and  $Q_4$  are permuted in rows ( $i$ ) and columns ( $j$ ) according to (13), (14) and (15), respectively, to get the confused images  $F_1(m, m)$ ,  $F_2(m, m)$  and  $F_3(m, m)$  respectively.

**Step 16:** Three chaotic sequences  $X'$ ,  $Y'$  and  $Z'$  are created via the Rössler system according to (2) by utilizing its initial and system parameters values, and each sequence has a size of  $(1, m \times m)$ .

**Step 17:** Reshape  $X'$ ,  $Y'$  and  $Z'$  to the same size of  $F_1$  to acquire  $X'_1$ ,  $Y'_1$  and  $Z'_1$ , respectively. Then,  $X'_1$  is used to diffuse the confused matrices  $F_1$ ,  $F_2$  and  $F_3$  pixels by applying matrices  $F_1$ ,  $F_2$  and  $F_3$  pixels by applying the bitwise XOR process as follows:

$$W_1(i, j) = \text{bitxor}(W_1(i, j-1), F_1(i, j), X'_1(i, j)) \quad (16)$$

$$W_2(i, j) = \text{bitxor}(W_2(i, j-1), F_2(i, j), X'_1(i, j)) \quad (17)$$

$$W_3(i, j) = \text{bitxor}(W_3(i, j-1), F_3(i, j), X'_1(i, j)) \quad (18)$$

where  $W_1$ ,  $W_2$  and  $W_3$  represent the diffused images.

**Step 18:** Concatenate the three matrices  $W_1$ ,  $W_2$  and  $W_3$  to get the final encrypted image  $W$  with a size of  $(m, m * 3)$ .

### C. IMAGE DECRYPTION PROCEDURE

The image decryption procedure is the opposite operation of the encryption procedure. The above encryption steps (in sections A and B) can be applied in inverse order to recover a plain image. The decryption procedure begins from the encrypted images  $G$  or  $W$  (in sections A and B) and finishes up with the input images  $A$  or  $P$ . Note that without knowing the secret keys of the Lorenz, Rössler, and El-Gamal systems or the secret pathways of the scan techniques, it is very difficult to recover the original image.

### IV. IMAGE SECURITY SIMULATION RESULTS

Many experiments have been conducted to validate the robustness, efficiency, and security of the proposed approach. To produce valid results, standard images are used in these experiments. The SIPI image database contains this set of digital images. This database includes four groups of images: Textures (64 images), Aerials (38 images), Miscellaneous (39 images) and Sequences (69 images). Our experiments were performed on these groups of images as follows: all the images in the first group, the first 20 images in the second group, 24 images for the third group, and finally the first 34 images in the fourth group. The results of these experiments will be shown separately later as curves due to space limitations. A sample of these images (grey and colour) that are sized  $512 \times 512$  from the SIPI database is shown in Fig. 4.

The MATLAB 2013a simulation software is employed in this research to implement image cryptosystem under Windows 7 operating system environment. In addition, a personal computer (laptop) with an Intel (R) Core (TM) i3 2.4 GHz CPU and 4 GB of installed memory (RAM), is used to execute the scientific experiments. The secret parameters utilized in our experiments are set as follows.

For the El-Gamal algorithm,  $p = 17$ ,  $g = 6$ ,  $x = 5$ ,  $y = 7$ , and  $k = 10$ .



**FIGURE 4.** Test images: (a) Cameraman, (b) Lena, (c) Baboon, (d) Goldhill, (e) Lake, (f) Elaine, (g) Colour Lena, (h) Colour Baboon, (i) Peppers, (j) Airplane, (k) Splash, and (l) Barbara.

For the Lorenz system,  $x_0 = 0.5389$ ,  $y_0 = -0.3946$ ,  $z_0 = 0.7142$ ,  $a = 10$ ,  $b = 28$ , and  $c = 8/3$ .

For the Rössler system,  $x_0 = -0.8913$ ,  $y_0 = 0.6256$ ,  $z_0 = -0.1978$ ,  $a = 0.2$ ,  $b = 0.2$ , and  $c = 5.7$ .

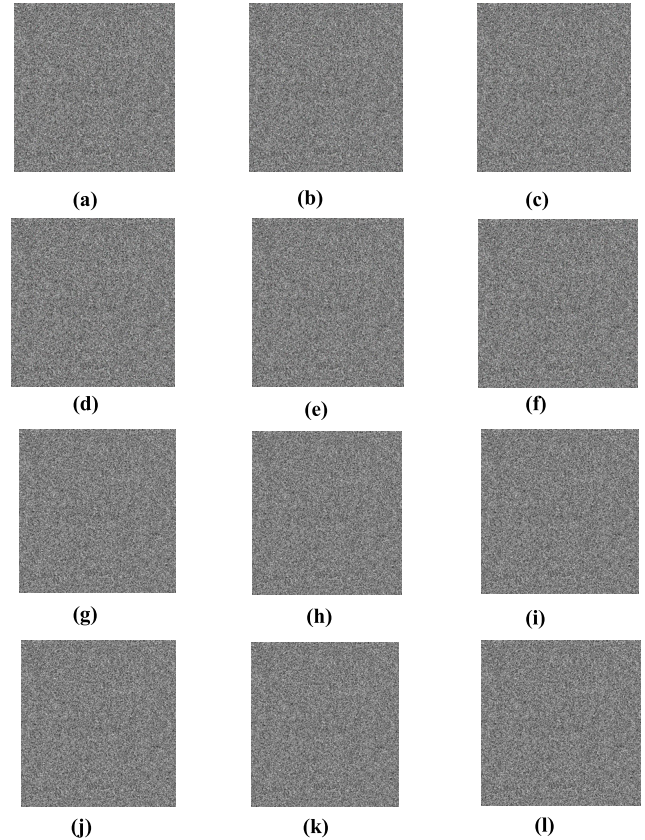
The encrypted and decrypted versions of all images in Fig. 4 are exhibited in Figs. 5 and 6, respectively. It is evident from the encryption outcomes in Fig. 5 that the grey and colour information within the input images is effectively concealed. In addition, the images that were reconstructed in Fig. 6 by employing the right secret keys are identical to the original input images. This implies that the suggested image security approach can effectively encrypt different types of grey and colour images and safeguard their confidential data. Furthermore, the original images can be readily recovered when the encrypted images are transferred to their authorized recipients.

## V. IMAGE SECURITY ANALYSIS

This section is devoted to measuring and analysing the security of the image encryption/decryption approach. Statistical, differential and key exhaustive search analyses are used to investigate the viability and effectiveness of our proposed approach as follows.

### A. HISTOGRAM ANALYSIS

An image histogram is a significant statistical analysis tool. It is a schematic method that exhibits the distribution of image pixels at every grey level intensity. A good cipher should



**FIGURE 5.** Image encryption results: (a) Cameraman, (b) Lena, (c) Baboon, (d) Goldhill, (e) Lake, (f) Elaine, (g) Colour Lena, (h) Colour Baboon, (i) Peppers, (j) Airplane, (k) Splash, and (l) Barbara.

provide a flat and uniform histogram to prevent statistical attacks [37], [38]. Fig. 7 shows the histograms of the unencrypted and encrypted Cameraman and Lena colour images (RGB channels). We can notice from Figs. 7 (a) and (c)-(e) that the histograms of the unencrypted images contain many peaks whereas the histograms of the encrypted images in Figs. 7 (b) and (f)-(h) contain pixel values at the extremist points of the histogram. It can be noticed that information leakage via a statistical attack is difficult on encrypted image. Hence, the proposed method can successfully prevent this type of attack. Chi-square analysis can also be used to measure the histogram uniformity. The chi-square definition is described as follows [27], [38]:

$$X^2 = \sum_{i=1}^{256} \frac{(o_i - e_i)^2}{e_i} \tag{19}$$

where  $o_i$  and  $e_i$  denote the observed and expected frequencies, respectively. The ideal theoretical value for  $X^2$  is 293.25 for 255 degrees of freedom and a significance level of 0.05. The results of the chi-square analysis for the sample images (Fig. 4) are given in Table 1. All outcomes produced by the presented scheme for the encrypted images in this table are smaller than the ideal value, which indicates the uniformity of the histogram. Hence, our proposed method passed the  $X^2$  test.





**FIGURE 6.** Image decryption results: (a) Cameraman, (b) Lena, (c) Baboon, (d) Goldhill, (e) Lake, (f) Elaine, (g) Color Lena, (h) Color Baboon, (i) Peppers, (j) Airplane, (k) Splash and (l) Barbara.

**B. IMAGE ENTROPY ANALYSIS**

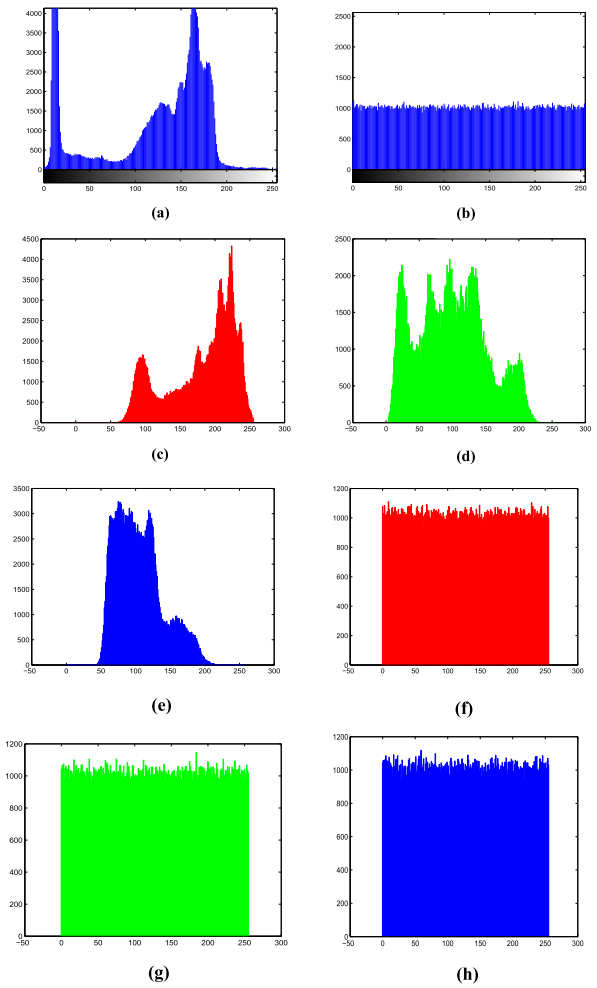
It is an important measure to test image randomness which can be calculated as follows [39], [40]:

$$H(X) = - \sum_{i=0}^n P(x_i) \log_2 P(x_i) \tag{20}$$

where  $X$  points to a random variable, which is defined as  $X = \{x_0, x_1, x_2, \dots, x_n\}$ ;  $P$  represents the probability of the symbol  $x_i$  and  $n$  denotes the image size. The theoretical ideal entropy value should be equal to 8. Larger entropy values imply a high secured cryptographic method. The outcomes of the computed entropies for the sample images and their encrypted versions obtained via our method are reported in Table 1. The entropies of all the encrypted images are close to 8, which shows that the suggested approach generates perfectly random output images; thereby, it possesses strong immunity to entropy analysis.

However, the global entropy (*Eq.(20)*) has some weaknesses such as low efficiency, inaccuracy and inconsistency. Local entropy analysis is used to overcome these weaknesses and to measure the encrypted image randomness over local non-overlapping blocks. The local image entropy is calculated as follows:

$$\overline{H_{(k, T_B)}(S)} = \sum_{i=1}^k \frac{H(S_i)}{k} \tag{21}$$



**FIGURE 7.** Histograms of plain and encrypted images: (a) plain Cameraman, (b) encrypted Cameraman, (c) and (f) plain and encrypted color Lena in R channel, (d) and (g) plain and encrypted color Lena in G channel and (e) and (h) plain and encrypted color Lena in B channel.

where  $S_1, S_2, \dots, S_k$  refer to the non-overlapping blocks, which are randomly selected from the encrypted image with  $T_B$  pixels and  $H(S_i)$  is the global entropy of  $S_i$ . When  $k$  and  $T_B$  are equal to 30 and 1936, respectively, the local entropy should be in the interval  $[7.901901305, 7.903037329]$  to meet the security requirements [9], [20], [22], [25]. Table 1 lists the test outcomes of the local entropy produced by our scheme for the sample images. The local entropy scores in this table for all encrypted images are greater than the optimal interval; hence, the proposed cryptosystem has high randomness to tolerate the local Shannon entropy test.

**C. CORRELATION ANALYSIS**

Correlation analysis is used to quantify the degree of association among the image pixel values. Typically, adjacent pixels in a normal image are characterized by possessing strong relationships. Effective cryptographic methods should lessen this relation in the encrypted image to prevent any pixel relation analysis attack [41], [42]. In general, a smaller correlation



TABLE 1. Results of the quality of the measurements between the plain and encrypted images.

Image	Plain Image entropy	Encrypted Image Entropy (Global)	Encrypted Image Entropy (Local)	MSE	PSNR (dB)	SSIM	$\chi^2$
Cameraman	7.0482	7.9994	7.9058	9379.2	8.4091	0.0308	230.2715
Lena	7.4455	7.9993	7.9062	7772.2	9.2254	0.0353	242.4023
Baboon	7.3585	7.9993	7.9070	7269.3	9.5159	0.0308	244.6406
Goldhill	4.5028	7.9993	7.9060	8137.9	9.0257	0.0361	237.2266
Lake	7.4570	7.9993	7.9064	9548.4	8.3315	0.0304	262.0215
Elaine	7.4950	7.9993	7.9068	7637.1	9.3015	0.0378	261.7559
Color Lena	7.7503	7.9997	7.9066	8955.3	8.6100	0.0345	236.4258
Color Baboon	7.7624	7.9997	7.9069	8636.7	8.7673	0.0292	244.1855
Peppers	7.6698	7.9997	7.9067	10158	8.0628	0.0283	265.4063
Airplane	6.6639	7.9995	7.9064	10381	7.9683	0.0344	268.5254
Splash	7.2428	7.9997	7.9065	11238	7.6238	0.0317	257.9094
Barbara	7.6986	7.9996	7.9068	8911.4	8.6314	0.0286	248.1178

TABLE 2. Results of the correlation coefficients of the sample images and their encrypted versions in three directions.

Image	Horizontal		Vertical		Diagonal	
	Plain	Encrypted	Plain	Encrypted	Plain	Encrypted
Cameraman	0.9828	-0.0097	0.9898	-0.0021	0.9723	-0.0080
Lena	0.9750	-0.0034	0.9848	-0.0072	0.9580	-0.0064
Baboon	0.8555	-0.0026	0.7756	-0.0062	0.7418	-0.0015
Goldhill	0.9704	0.0021	0.9708	-0.0117	0.9504	0.0042
Lake	0.9755	0.0074	0.9737	-0.0060	0.9568	-0.0144
Elaine	0.9926	-0.0255	0.9941	0.0071	0.9851	-0.0234
Color Lena	0.9818	-0.0021	0.9903	-0.0030	0.9698	-0.0177
Color Baboon	0.9214	-0.0081	0.8663	-0.0011	0.8510	-0.0065
Peppers	0.9694	-0.0096	0.9668	-0.0094	0.9622	0.0039
Airplane	0.9727	-0.0206	0.9517	-0.0086	0.9362	-0.0099
Splash	0.9922	0.0061	0.9968	-0.0238	0.9883	-0.0104
Barbara	0.9305	0.0069	0.9728	-0.0187	0.9126	-0.0216

between neighbouring pixels in the encrypted image means a stronger cryptographic algorithm. The correlation coefficient ( $r_{xy}$ ) can be calculated in any specific direction (vertical, horizontal or diagonal) by randomly choosing several pairs of adjoining pixels.  $r_{xy}$  is defined as in the following formulas:

$$E(x) = \frac{1}{N} \sum_{i=1}^N x_i \tag{22}$$

$$D(x) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))^2 \tag{23}$$

$$cov(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))(y_i - E(y)) \tag{24}$$

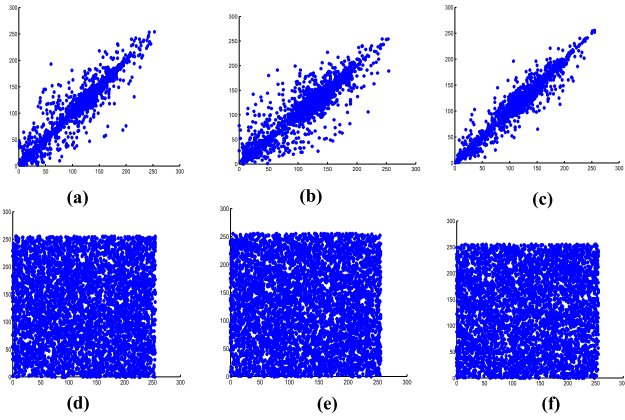
$$r_{xy} = \frac{cov(x, y)}{\sqrt{D(x)} \times \sqrt{D(y)}} \tag{25}$$

where  $x_i$  and  $y_i$  form the pair of adjacent pixels in the image; and  $D$ ,  $E$ , and  $cov$  symbolize the variance, mean, and covariance, respectively. Five thousand (5000) neighbouring pixel pairs are randomly chosen in the experiments from the same position of the output and input images to compute the correlation coefficients along the horizontal, vertical and diagonal directions to verify the algorithm’s performance.

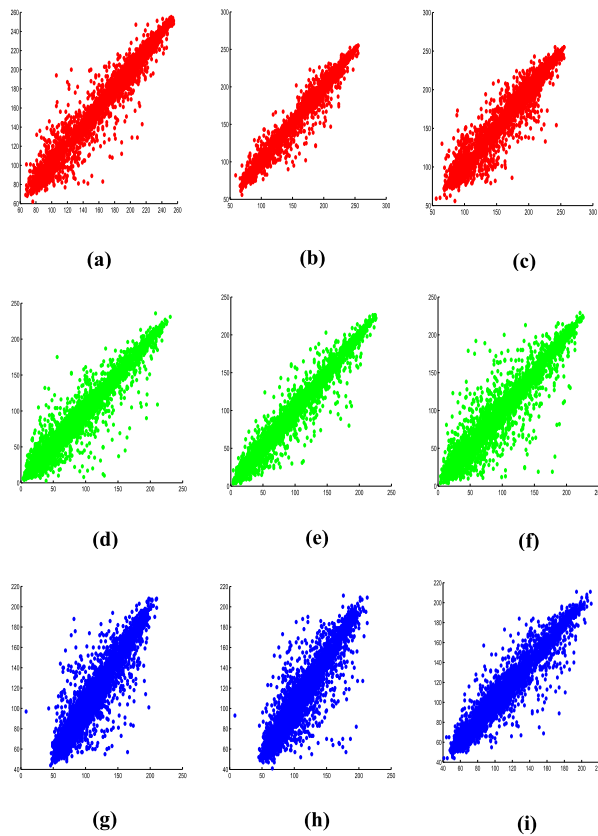
The calculated correlation coefficients for unencrypted and encrypted images are shown in Table 2. It can be found from the numerical results in Table 2 that the correlation values of unencrypted images are approximately one while those of the encrypted images are approximately zero. In addition, the visual results of the neighbouring pixel distributions in the horizontal, vertical and diagonal directions of the Cameraman image and Lena colour image (in RGB channels) are presented in Figs. (8-10). The visual comparison of the results reveals that the pixel couples in the unencrypted image are grouped along the diagonal direction, which points to strong correlation among pixel pairs, whilst the pixel couples in the encrypted image are equally distributed along the whole range of grey level values. The numerical and visual results in Table 2 and Figs. (8-10) imply that the relation among adjacent pixel pairs in the plain image can be effectively broken in the encrypted image by the proposed image encryption approach.

**D. PEAK SIGNAL TO NOISE RATIO (PSNR)**

The peak signal to noise ratio is a common metric used to quantify the quality of an image in terms of the mean squared error (MSE). This metric measures the amount of



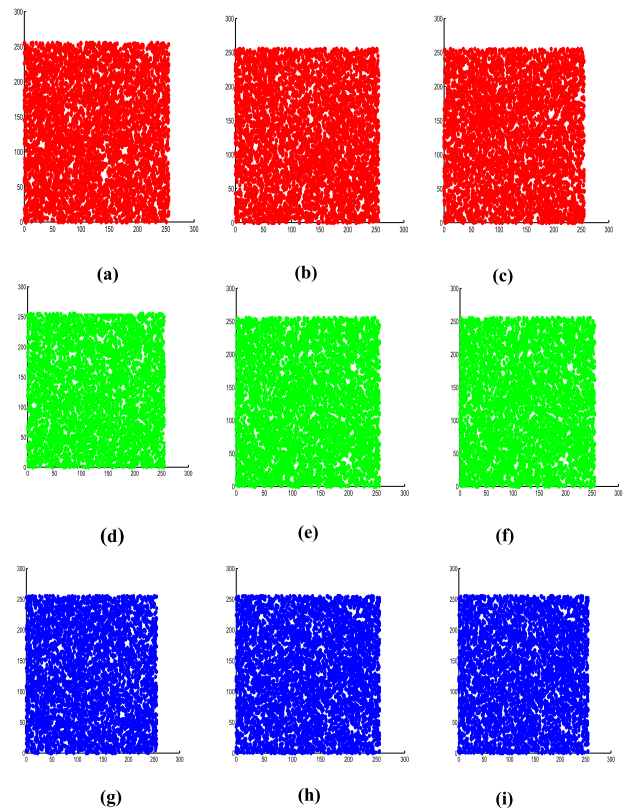
**FIGURE 8.** Distribution of neighbouring pixels for the correlation analysis for the Cameraman image in the following directions: (a) and (d) Horizontal direction for plain and encrypted images, (b) and (e) Vertical direction for plain and encrypted images, and (c) and (f) Diagonal direction for plain and encrypted images.



**FIGURE 9.** Distribution of neighbouring pixels for the correlation analysis for the plain Lena colour image in the following direction: (a) Horizontal, (b) vertical, and (c) diagonal directions in the R channel; (d) Horizontal, (e) vertical, and (f) diagonal directions in the G channel; and (g) Horizontal, (h) vertical, and (i) diagonal directions in the B channel.

error difference between plain and encrypted images as follows [43], [44]:

$$MSE = \frac{1}{N \times N} \sum_{i=1}^N \sum_{j=1}^N [X(i, j) - Y(i, j)]^2 \quad (26)$$



**FIGURE 10.** Distribution of neighbouring pixels for the correlation analysis for the encrypted Lena colour image in the following directions: (a) Horizontal, (b) vertical, and (c) diagonal directions in the R channel; (d) Horizontal, (e) vertical, and (f) diagonal directions in the G channel; and (g) Horizontal, (h) vertical, and (i) diagonal directions in the B channel.

$$PSNR = 10 \times \log_{10} \left[ \frac{255 \times 255}{MSE} \right] \quad (27)$$

where  $X$  and  $Y$  denote the encrypted and unencrypted images, respectively; and  $N \times N$  indicates the image size. A high MSE and a low PSNR mean a greater difference between the output and input images. Table 1 illustrates the PSNRs and MSEs of the images tested by our proposed approach. It is clear from Table 1 that the MSEs are notably large, whereas the PSNRs are notably small, for each tested image. These results prove the high distortion in the encrypted image, which means that the proposed technique possesses good performance in terms of image diffusion and confusion.

### E. STRUCTURAL SIMILARITY INDEX MEASURE (SSIM)

SSIM is a popular measure used for computing the visual difference between the plain and encrypted images. The SSIM value between two images  $I_1$  and  $I_2$  is computed as follows [19], [26], [40]:

$$SSIM(I_1, I_2) = \frac{(2\mu_{I_1}\mu_{I_2} + \alpha)(2\sigma_{I_1I_2} + \beta)}{(\mu_{I_1}^2 + \mu_{I_2}^2 + \alpha)(\sigma_{I_1}^2 + \sigma_{I_2}^2 + \beta)} \quad (28)$$

TABLE 3. Results for NPCR.

Image (512 × 512)	NPCR (%)	Critical values for NPCR		
		$N_{0.05} = 99.5893\%$	$N_{0.01} = 99.5810\%$	$N_{0.001} = 99.5717\%$
Cameraman	99.61	Pass	Pass	Pass
Lena	99.61	Pass	Pass	Pass
Baboon	99.61	Pass	Pass	Pass
Goldhill	99.61	Pass	Pass	Pass
Lake	99.61	Pass	Pass	Pass
Elaine	99.61	Pass	Pass	Pass
Colour Lena	99.60	Pass	Pass	Pass
Colour Baboon	99.60	Pass	Pass	Pass
Peppers	99.61	Pass	Pass	Pass
Airplane	99.61	Pass	Pass	Pass
Splash	99.61	Pass	Pass	Pass
Barbara	99.61	Pass	Pass	Pass

TABLE 4. Results for the UACI.

Image (512 × 512)	UACI (%)	Critical values of UACI		
		$U_{0.05}^- = 33.3730\%$	$U_{0.05}^- = 33.3730\%$	$U_{0.05}^- = 33.3730\%$
		$U_{0.05}^+ = 33.5541\%$	$U_{0.05}^+ = 33.5541\%$	$U_{0.05}^+ = 33.5541\%$
Cameraman	33.3740	Pass	Pass	Pass
Lena	33.4902	Pass	Pass	Pass
Baboon	33.5480	Pass	Pass	Pass
Goldhill	33.0875	Fail	Fail	Fail
Lake	35.5319	Fail	Fail	Fail
Elaine	32.2328	Fail	Fail	Fail
Color Lena	33.4573	Pass	Pass	Pass
Color Baboon	33.3875	Pass	Pass	Pass
Peppers	33.5503	Pass	Pass	Pass
Airplane	36.9962	Fail	Fail	Fail
Splash	33.4687	Pass	Pass	Pass
Barbara	33.4006	Pass	Pass	Pass

where  $\mu_{I_1}$  and  $\mu_{I_2}$  refer to the averages of  $I_1$  and  $I_2$ , respectively;  $\sigma_{I_1 I_2}$  is the covariance between  $I_1$  and  $I_2$ ;  $\alpha$  and  $\beta$  represent the constants; and  $\sigma_{I_1}^2$  and  $\sigma_{I_2}^2$  are the variances of  $I_1$  and  $I_2$ , respectively. The lower the SSIM is, the better the encryption algorithm is. The outcomes of the SSIM for the test images are illustrated in Table 1. It is evident from the reported scores in Table 1 that the introduced mechanism can destroy the similarity between the original test images and their corresponding encrypted images.

**F. DIFFERENTIAL ANALYSIS**

The best known method to weaken a cryptographic algorithm is the chosen-plaintext attack using differential analysis. Generally, the attacker initially produces two plain images that differ by one bit only. Next, an encryption scheme is implemented on both images by employing the same keys. After that, a comparison is made in order to monitor the differences between the two generated encrypted images. The relation between the original and encrypted images may be a helpful way to determine the secret keys. A differential attack may be impractical and not robust if a one-bit variation of the original image impacts each bit in the corresponding encrypted image. Two common indices called the Unified Average Changing Intensity (UACI) and the Number of Pixels Change Rate (NPCR) are often utilized in practice by researchers to

quantify a system’s ability to resist a differential analysis. The UACI and NPCR are calculated as follows:

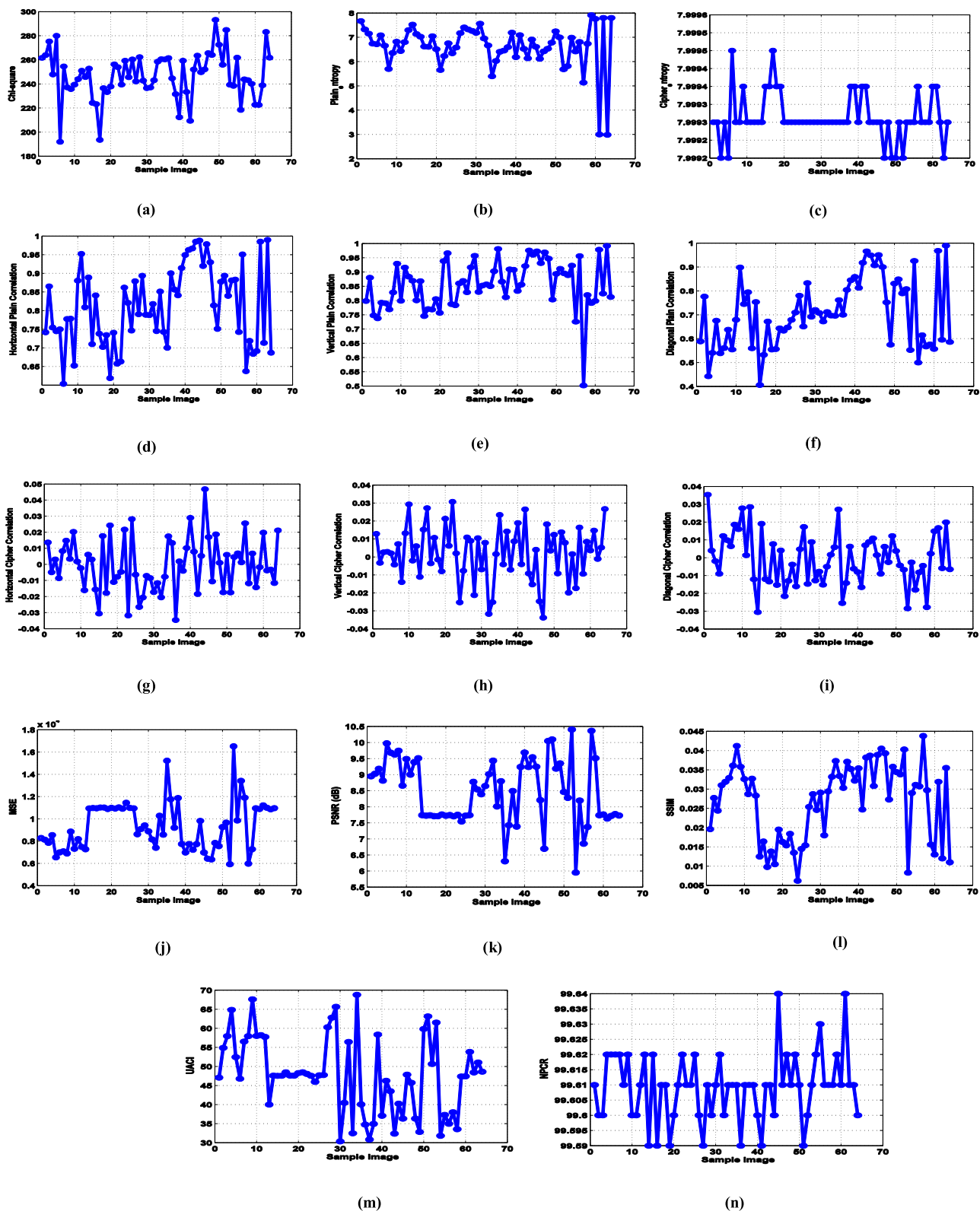
$$NPCR = \frac{\sum_{i,j} D(i,j)}{N \times N} \times 100\% \tag{29}$$

$$UACI = \frac{1}{N \times N} \left[ \sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{255} \right] \times 100\%$$

$$D(i) = \begin{cases} 0 & \text{if } C_1(i,j) = C_2(i,j) \\ 1 & \text{if } C_1(i,j) \neq C_2(i,j) \end{cases} \tag{30}$$

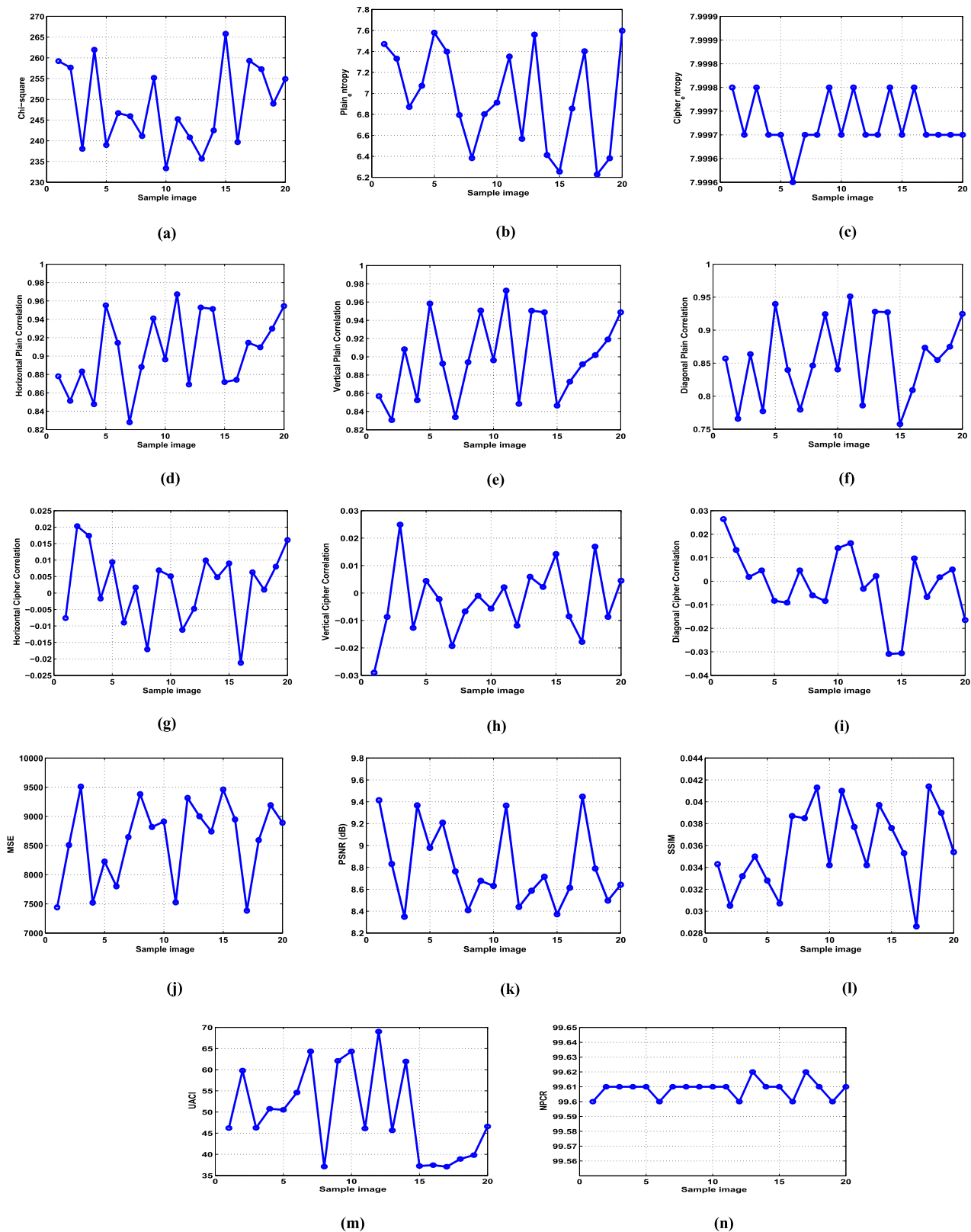
where  $C_1$  symbolizes the first encrypted image while  $C_2$  symbolizes the second encrypted image after modifying one pixel in the unencrypted input image [45]–[48].

The critical values of the NPCR index are shown in Table3. The values of this analysis are calculated for different kinds of tested images such that each image size is (512 × 512) with the significance levels of 0.05 (99.5893%), 0.01 (99.5810%) and 0.001 (99.5717%) respectively [49]. It is evident from the experimental results in Table 3 that the the values of NPCR for all cases are larger than critical values which implies that the presented proposed chaotic cryptosystem passed all required NPCR analysis tests successfully. The critical values of the UACI index are shown in the Table 4. The values of this analysis are calculated for different kinds of tested images such that each image size is (512 × 512) with

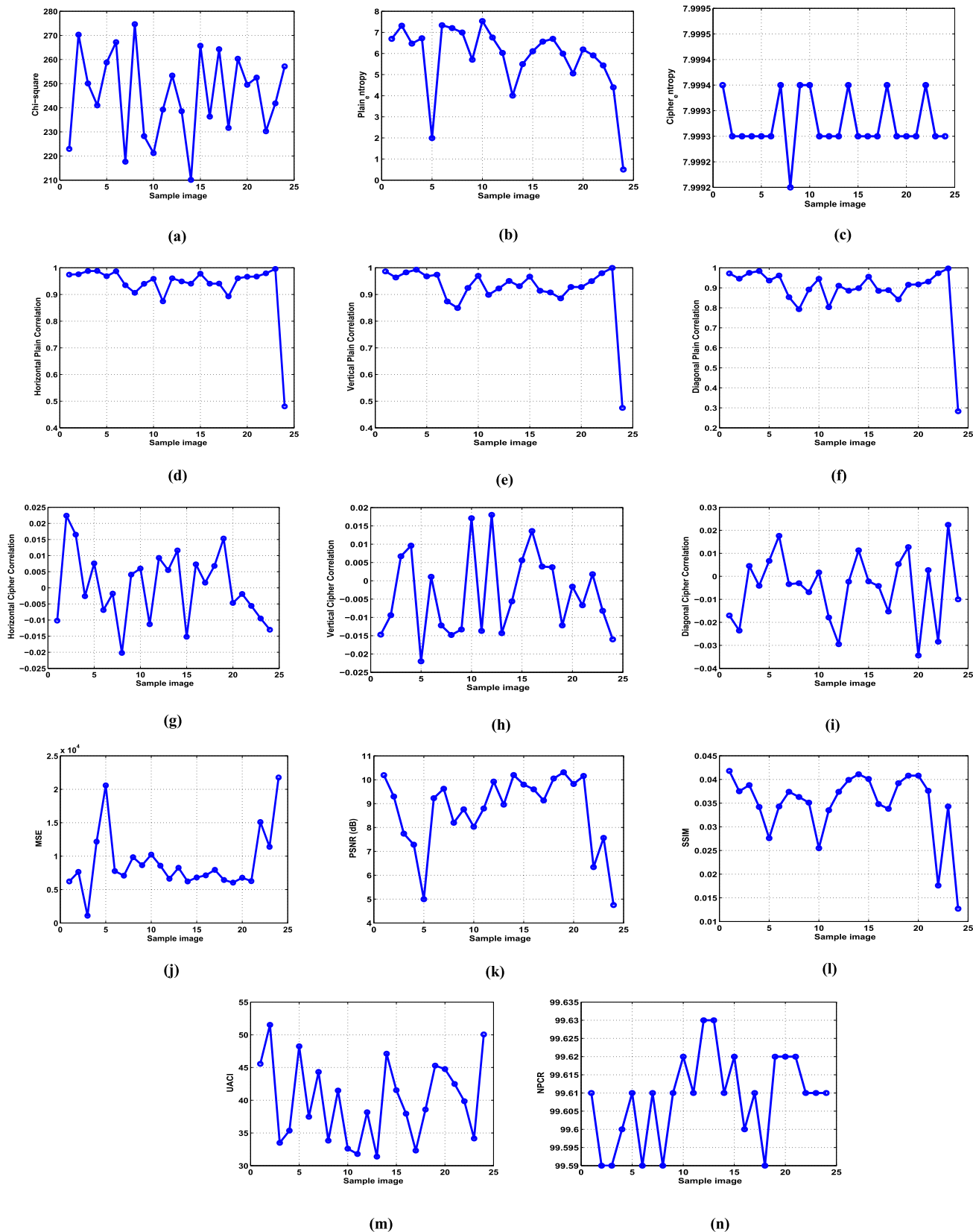


**FIGURE 11.** Results of the quality measures for textures group in terms of the (a) Chi-square; (b) Global plain image entropy; (c) Global encrypted image entropy; Correlation in the (d) horizontal, (e) vertical, and (f) diagonal directions of the plain image; Correlation in the (g) horizontal, (h) vertical, and (i) diagonal directions of the encrypted image; (j) MSE; (k) PSNR; (l) SSIM; (m) UACI; and (n) NPCR.

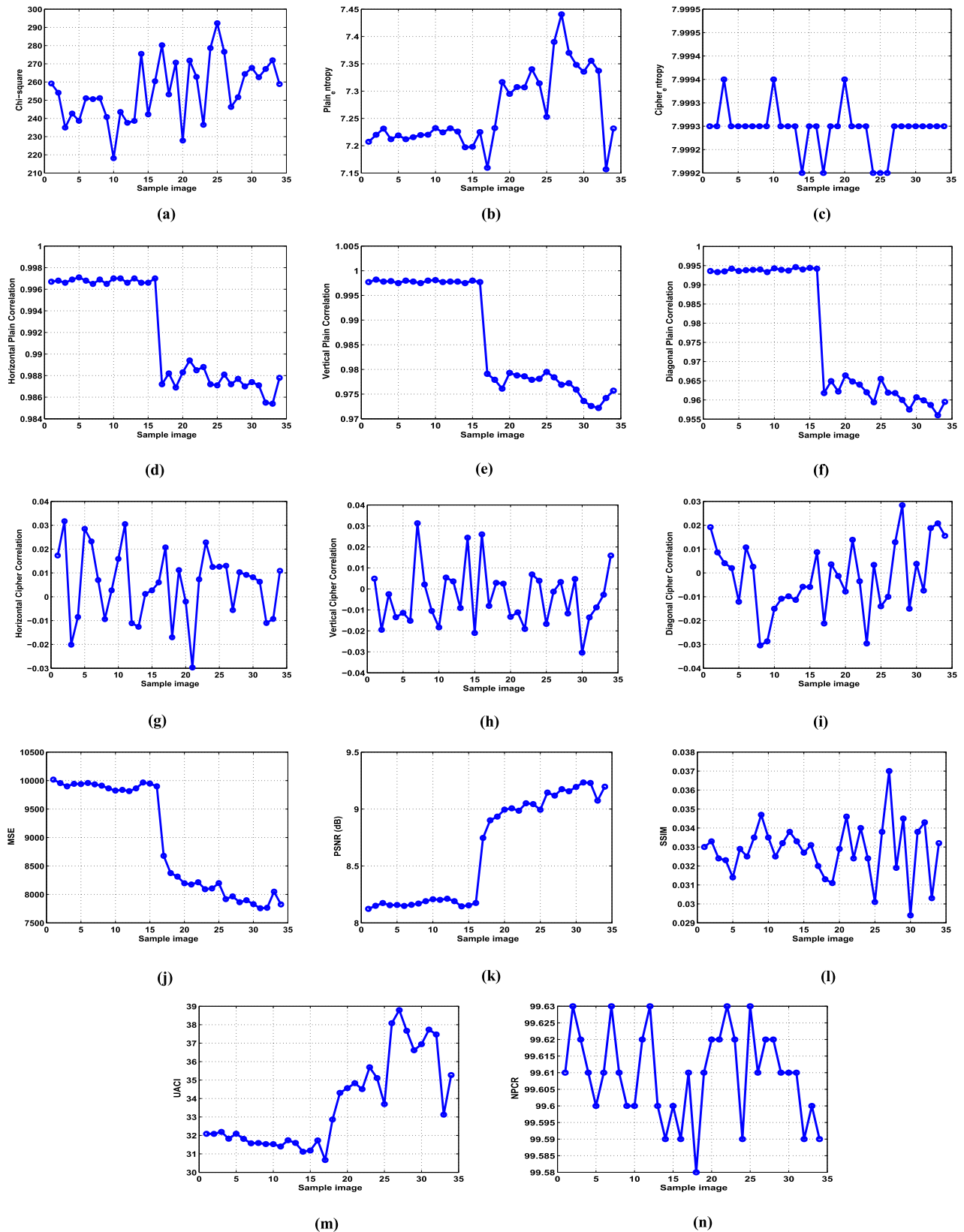




**FIGURE 12.** Results of the quality measures for the aerials group in terms of the (a) Chi-square; (b) Global plain image entropy; (c) Global encrypted image entropy; Correlation in the (d) horizontal, (e) vertical, and (f) diagonal directions of the plain image; Correlation in the (g) horizontal, (h) vertical, and (i) diagonal directions of the cipher image; (j) MSE; (k) PSNR; (l) SSIM; (m) UACI; and (n) NPCR.



**FIGURE 13.** Results of the quality measures for the miscellaneous group in terms of the (a) Chi-square; (b) Global plain image entropy; (c) Global encrypted image entropy; Correlation in the (d) horizontal, (e) vertical, and (f) diagonal directions of the plain image; Correlation in the (g) horizontal, (h) vertical, and (i) diagonal directions of the cipher image; (j) MSE; (k) PSNR; (l) SSIM; (m) UACI; and (n) NPCR.



**FIGURE 14.** Results of the quality measures for the sequence group in terms of the (a) Chi-square; (b) Global plain image entropy; (c) Global encrypted image entropy; Correlation in the (d) horizontal, (e) vertical, and (f) diagonal directions of the plain image; Correlation in the (g) horizontal, (h) vertical, and (i) diagonal directions of the cipher image; (j) MSE; (k) PSNR; (l) SSIM; (m) UACI; and (n) NPCR.

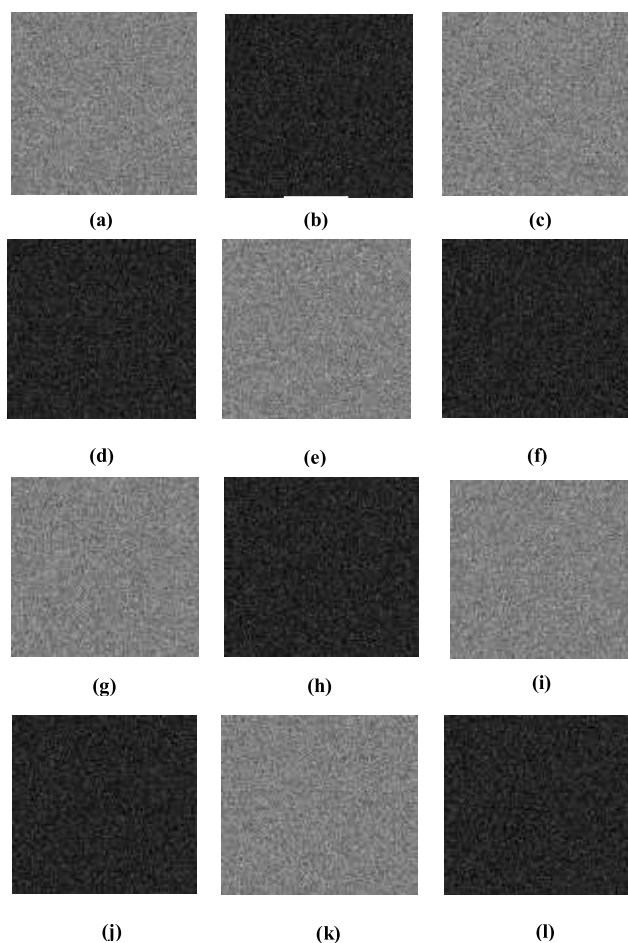
the significance levels 0.05 (33.3730%, 33.5541%), 0.01 (33.3445%, 33.5826%) and 0.001 (33.3115%, 33.6156%) respectively [49]. It is obvious from the UACI values in this table that index values within the range of critical values except for Goldhill, Lake, Elaine and Airplane images. Hence, this observation indicates that the presented approach possesses a powerful diffusion mechanism and it is also has high sensitivity to the tiny changes in the plaintext such that can prevent differential analysis attacks.

The encryption results of the four image groups in the SIPI database (section IV) in terms of the chi-square analysis, global entropy, and correlation analysis in three directions; the MSE; the PSNR; the SSIM; the UACI and the NPCR are clarified in Figs. (11-14).

The chi-square test results are shown in the Figs. 11 (a), 12 (a), 13 (a) and 14 (a) respectively. We have noticed from these experimental results that the test scores are smaller than their ideal value (293.25). Such observation implies that the proposed approach passed the  $X^2$  test for the four image groups in SIPI image dataset. Next, the global entropy test results are shown in the Figs. 11 (b), 12 (b), 13 (b) and 14 (b) respectively. The scores are generally less than 8, while these scores are very close to 8 in the encrypted images as shown in the Figs. 11 (c), 12 (c), 13 (c) and 14 (c) respectively. This means that the presented scheme produces encrypted image with excellent randomness and passed global entropy test for the four SIPI image groups. Next, for the correlation analysis in horizontal, vertical and diagonal directions in Figs. 11 (d, e, f), 12 (d, e, f), 13 (d, e, f) and 14 (d, e, f) respectively, the correlation value is close to 1 due to the strong relationship between pixels in the input image along the three directions. However, these values fluctuate between positive and negative values (or in other word close to 0) in the output image along three directions as illustrated in Figs. 11 (g, h, i), 12 (g, h, i), 13 (g, h, i) and 14 (g, h, i) respectively. Such observation means that the suggested algorithm can lessen the correlation between the image pixels and also it pass the correlation coefficient test for the all images in the SIPI dataset. For the MSE test in Figs. 11 (j), 12 (j), 13 (j) and 14 (j) respectively, the outcome values are very large. In contrary to the PSNR outcomes in Figs. 11 (k), 12 (k), 13 (k) and 14 (k) respectively, are very low. Such opposite behavior between MSE and PSNR demonstrates that the difference between input and output image is extremely high. It is also proves that proposed approach produces encrypted images with high distortion for the different kinds of images in the SIPI dataset to defeat differential attack and increase dissimilarity between plain and encrypted images.

**G. KEY SPACE ANALYSIS**

Key space analysis represents the group of all possible keys that can be utilized in an image security system. A good encryption method should possess considerable key space in order to possess enough security to resist an exhaustive attack. Broadly, the key space size should be greater than  $2^{100}$  [21], [50], [51]. The secret key in the presented



**FIGURE 15.** Results of the key sensibility test in encryption: (a) encrypted image for Key 1, (b) difference between Fig. 5 (b) and (a), (c) encrypted image for Key 2, (d) difference between Fig. 5 (b) and (c), (e) encrypted image for Key 3, (f) difference between Fig. 5 (b) and (e), (g) encrypted image for Key 4, (h) difference between Fig. 5 (b) and (g), (i) encrypted image for Key 5, (j) difference image Fig. 5 (b) and (i), (k) encrypted image for Key 6, and (l) difference between Fig. 5 (b) and (k).

work basically contains the confidential key  $x$  of the El-Gamal mechanism; the initial parameter values of the Lorenz system that are utilized in the permutation phase of  $x_0, y_0, z_0, a, b, \text{ and } c$ ; and the initial parameters values of the Rössler system that are utilized in the diffusion phase of  $x_0, y_0, z_0, a, b, \text{ and } c$ . Then, the total number of these keys is 13 and their size is  $10^{13}$ . If every key has a calculation accuracy of  $10^{-15}$  in a computer, then the key space will reach  $10^{13} \times 10^{15} = 10^{195} \approx 2^{648}$  bits. Moreover, if the zigzag and spiral scanning pathway technologies are considered as secret keys, then the inclusive key space of the introduced method is much greater than  $2^{100}$ , which is big enough to prevent an exhaustive attack.

**H. KEY SENSITIVITY ANALYSIS**

Key sensitivity analysis guarantees that no information can be uncovered about a plain image if there is a tenuous modification in the secret keys. This implies that a minor variation in the encryption and/or decryption keys should



TABLE 5. Results of the key sensibility test in the encryption operation for the grey Lena image.

Modulated parameter by adding $\Delta$	Obtained key	UACI (%) Between plain and encrypted images	NPCR (%) Between plain and encrypted images	MSE Between plain and encrypted images	PSNR (dB) Between plain and encrypted images	SSIM Between plain and encrypted images	$r_{xy}$ between two encrypted images
(x) of El-Gamal algorithm	Key 1	45.5305	99.63	7764.9	9.2294	0.0354	0.0030
( $x_0$ ) of Lorenz system	Key 2	44.9817	99.57	7780.2	9.2209	0.0360	-0.0009657
(a) of Lorenz system	Key 3	46.4017	99.58	7753.2	9.2360	0.0358	0.0036
(c) of Lorenz system	Key 4	45.6451	99.62	7771.3	9.2259	0.0352	-0.0010
( $z_0$ ) of Rössler system	Key 5	45.0963	99.64	7787.0	9.2171	0.0364	-0.0018
(b) of Rössler system	Key 6	46.0171	99.61	7789.1	9.2159	0.0369	-0.0039

yield a large deformity in the encrypted and unencrypted images [40], [52]. The key sensibility affects the cryptographic operation by employing slightly modified keys to encrypt the same input image. The Grey Lena image (Fig. 4 (b)) is utilized as the input image and its encrypted version obtained via the right secret keys is illustrated in Fig. 5 (b). Now, one of these keys (for instance, x of the El-Gamal algorithm) is modulated with a variation ( $\Delta = \lfloor(10)\wedge(-14)$ ) whilst maintaining the other parameters constant. Next, this modulated key with the original keys is used to encrypt the input image to get the encrypted image, as presented in Fig. 15 (a), whereas Figs. 15 (c), 15 (e), 15 (g), 15 (i) and 15 (k) clarify the subsequent images obtained by applying other wrong secret keys. To visually observe the variation in the resultant encrypted images, the differences between the right encrypted image (Fig. 5 (b)) and those consequent encrypted images are explained in Figs. 15 (b), 15 (d), 15 (f), 15 (h), 15 (j) and 15 (l). It can be concluded from the results in Fig. 15 that the encrypted images have a large variation, even if a trivial change occurs to one of the secret keys. We can prove the immunity of our proposed approach to the attacks on the key by observing the difference between the original and encrypted images in terms of the UACI, NPCR, MSE, PSNR and SSIM, as shown below.

From the values in Tables 1, 3 and 4, it is obvious that a small changing key leads to more than 99% of the pixels being altered in the encrypted image. Moreover, the correlation between the original encrypted image and the wrong encrypted images are tabulated in Table 5. The correlation scores in each case of Table 5 are near to zero, which demonstrates that the encrypted images are extremely different from each other. Fig. 15 and Table 5 prove the high sensitivity of the cryptosystem to the ciphering keys. The key sensibility has an impact on the decryption operation, and it is verified by decrypting the encrypted image using one altered key and keeping the others constant. The same modified secret keys in the encryption operation are employed to decrypt the encrypted Grey image Baboon (as in Fig. 5(c)) and the restored images are shown in Fig. 16. Clearly, the decryption outcomes in this figure with incorrect keys yield noisy images and no beneficial information about the input image can be revealed from them. Table 6 shows the UACI, NPCR, MSE, PSNR, SSIM and correlation values between

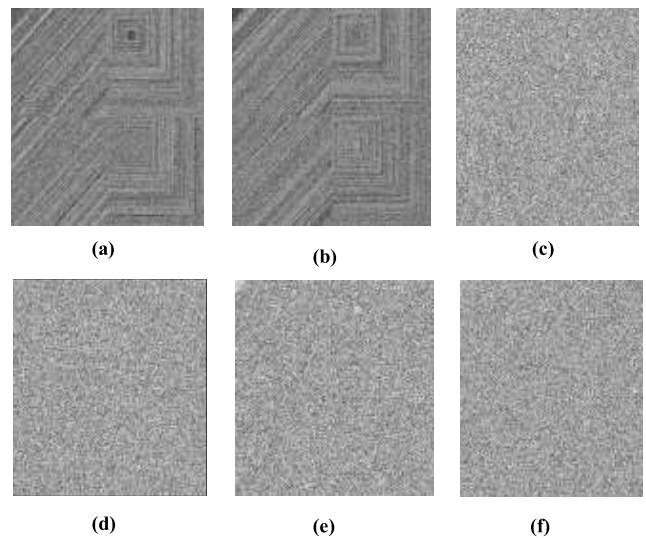


FIGURE 16. Results of the key sensibility test in decryption: (a) decrypted image for Key 1, (b) decrypted image for Key 2, (c) decrypted image for Key 3, (d) decrypted image for Key 4, (e) decrypted image for Key 5, and (f) decrypted image for Key 6.

the plain and decrypted images. The encryption/decryption keys are acquired from slightly altered in the original keys. Table 6 shows that the MSEs are high while the PSNR, SSIM and correlation values are so low. Additionally, the UACIs and NPCRs are extremely large and near their ideal or critical values, which indicates that the difference between plain and decrypted images is more than 99%. Fig. 16 and Table 6 confirm the high sensibility of the proposed approach to the decryption keys.

Based on the above discussion, we can say that our approach is very sensible to any change in the confidential keys, which makes it able to withstand an exhaustive attack.

I. TIME AND COMPLEXITY ANALYSIS

Time is an important feature for real-time applications to assess the efficiency of an encryption algorithm [38]. The computer configuration that is used to implement this algorithm was mentioned in section IV. First, the encryption process consists of four stages: the scanning (zigzag and spiral), El-Gamal cryptosystem, confusion and diffusion processes. Second, the time is recorded for each stage and the overall

TABLE 6. Results of the key sensibility test in the decryption operation for the grey Baboon image.

Obtained key	UACI (%) between plain and decrypted images	NPCR (%) between plain and decrypted images	MSE between plain and decrypted images	PSNR (dB) between plain and decrypted images	SSIM between plain and decrypted images	$r_{xy}$ between plain and decrypted images
Key 1	46.8241	99.95	16681	5.9085	0.0670	-0.0105
Key 2	42.3033	99.60	2897.3	13.5108	0.0663	0.0508
Key 3	46.2077	99.59	2930.9	13.4607	0.0668	0.0236
Key 4	40.8447	99.57	2974.0	13.3975	0.0686	-0.0022
Key 5	47.1439	99.61	12589	7.1310	0.0230	-0.0020
Key 6	47.1217	99.64	12506	7.1598	0.0209	0.0028

encryption time is also computed for all the tested images (shown in Fig. 4). Figs. 17 (a) and 17 (b) present the overall encryption times of the grey and colour images with different sizes (from  $128 \times 128$  to  $1024 \times 1024$ ) in Figs. 4 (a-f) and Figs. 4 (g-l). It can be noticed from these figures that if the image size is increased, the encryption time is also increased for all tested images. For instance, the encryption time for the  $128 \times 128$  Cameraman image increases from 0.0794s to 6.4376s if image size increased to  $1024 \times 1024$ . Furthermore, we found that the scanning stage took the longest time in comparison with the El-Gamal algorithm encryption stage with the least processing time. For example, the execution time for the scanning operation is 0.041959s while the execution time for the El-Gamal operation is 0.000897s for  $128 \times 128$  Cameraman image. Fig. 17 (c) shows the total encryption time for different sized Cameraman image. According to this figure, the execution time for each stage of the encryption process increases gradually with the image size such that the execution time of these operations increase from 0.018734s (for an image size of  $128 \times 128$ ) to 1.231179s (for an image size of  $1024 \times 1024$ ). In addition, the colour image takes much longer time than the grey image because it consists of three channels: Red, Green and Blue. Fig. 17 (d) clarifies the differences between the total encryption time for the grey and colour Lena images with different sizes. It is clear from the visual comparison between the two curves in Fig. 17 (d) that the grey image takes less time than the colour image. For the  $256 \times 256$  grey Lena, the total encryption time is 0.2378s, whereas for the  $256 \times 256$  colour Lena, the total encryption time is 0.7858s. Based on the time analysis in Fig. 17, it can be concluded that the total encryption time of our proposed approach for the sample tested images is small. Hence, this approach is a promising candidate to encrypt images with different sizes.

J. RANDOMNESS ANALYSIS

The random numbers produced by a chaotic system should be analysed with several tests to check their randomness before usage. There are common standard tests used to achieve this purpose such as the TestU01, Diehard and NIST tests.

The NIST test package consists of fifteen tests in total recommended via NIST US. All these statistical tests are executed by utilizing 100 series with the stream of length

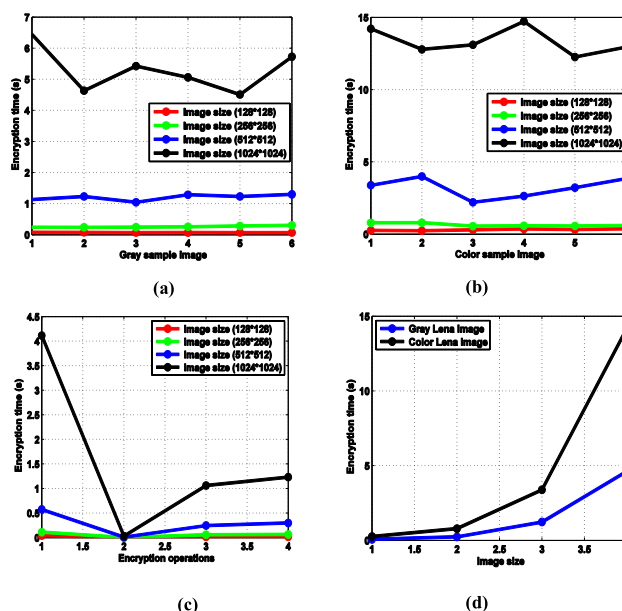


FIGURE 17. Results of time complexity analysis (a) total encryption time of grey images with different sizes, (b) total encryption time of color images with different sizes, (c) encryption time at each operation for Cameraman with different sizes and (d) total encryption time comparison between grey and colour Lena images with different sizes.

( $i=1000000$ bits) for each chaotic system used in this work. The NIST tests are based on P-value. If the obtained P-value  $\geq \alpha$  in each test, where  $\alpha$  refers to the significance level (or the decision rule) and  $\alpha = 0.01$ , then the generated binary sequence is random and it passes the test with a confidence of ninety-nine percent. Otherwise, if  $P\text{-value} < \alpha$ , then the produced sequence is not random and it fails the test with a confidence of ninety-nine percent [25], [31], [41]. In addition, if the success proportion is larger than 96%, then the produced sequence passes the 15 NIST tests. Table 7 is used to present the outcomes of the NIST tests for the binary sequences used in our approach. These sequences are produced by the two chaotic systems mentioned earlier in the presented method. This table reveals that the produced sequences via this cryptosystem are random and pass the NIST tests.

TestU01 is also performed in order to test the randomness level of the chaotic systems employed in the proposed

**TABLE 7. Results of the NIST test for the sequences produced with the used chaotic systems.**

Test name	Sequence of Lorenz system			Sequence of Rössler system		
	P-value	Success proportion	State	P-value	Success proportion	State
Frequency	0.8003	100%	Pass	0.7131	100%	Pass
Block frequency ( $m = 128$ )	0.5508	100%	Pass	0.8629	100%	Pass
Runs	0.7893	100%	Pass	0.5652	100%	Pass
Longest run ( $M = 1000, N = 100$ )	0.9478	100%	Pass	0.8470	100%	Pass
Rank	0.8911	100%	Pass	0.4487	100%	Pass
FFT	0.2329	100%	Pass	0.0435	100%	Pass
Non-overlapping template ( $m = 9, B = 001000110$ )	0.6034	100%	Pass	0.4923	100%	Pass
Overlapping template ( $m = 9, B = 110000100$ )	0.9245	100%	Pass	0.8690	100%	Pass
Universal ( $L = 5, Q = 1000, K = 145000$ )	0.4881	100%	Pass	0.4560	100%	Pass
Linear complexity ( $M = 1000$ )	0.9648	100%	Pass	0.6162	100%	Pass
Serial ( $M = 16$ )	0.9994	100%	Pass	0.4857	100%	Pass
Approximate entropy ( $m = 10$ )	0.9708	100%	Pass	0.5128	100%	Pass
Cumulative sums (forward)	0.9898	100%	Pass	0.8473	100%	Pass
Random excursions ( $x = -4$ )	0.9765	100%	Pass	0.9202	100%	Pass
Random excursions variant ( $x = -9$ )	0.7664	100%	Pass	0.8551	100%	Pass

**TABLE 8. Results of the TestU01 for the sequences produced with the used chaotic systems.**

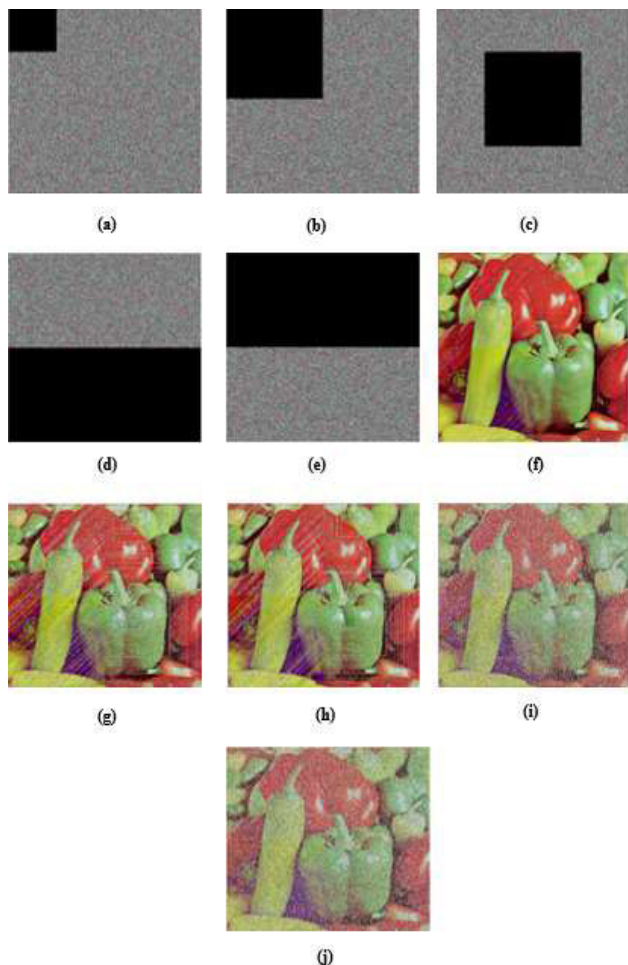
Battery	Sequence of Lorenz system			Sequence of Rössler system		
	Number of statistics	Success proportion	State	Number of statistics	Success proportion	State
Small crush	13	98%	Pass	11	97%	Pass
Crush	107	98%	Pass	101	97%	Pass
Big crush	120	99%	Pass	115	98%	Pass
Rabbit	38	98%	Pass	37	98%	Pass
Alphabit	15	97%	Pass	16	99%	Pass
Block Alphabit	100	98%	Pass	101	98%	Pass

method. TestU01 contains six tests. The first three tests are utilized for sequences of random numbers in  $U(0, 1)$ , these tests are Small crush, Crush, and Big crush. The other three tests are used for bit or binary sequences, these tests are Rabbit, Alphabit, and Block alphabit. Crush employs 96 statistical tests, Big crush employs 106 tests, while Rabbit and Alphabit use 38 and 17 various tests respectively. 1000000 bits are utilized in the presented work for each generated sequence in each test whilst the other parameters are selected automatically in each test as a function of the available number of generated bits. Block alphabit test utilizes the tests of Alphabit repeatedly in a binary file, the bits in this file are reordered then stored in blocks with various sizes, for example, the block size can be 2 or 4 or 8 or 16 bits [53]. The results obtained by implementing TestU01 on the binary sequences generated by the chaotic systems used in this approach are reported in Table 8. It is clear from the TestU01 scores in this table that the success proportions for all the six tests are larger than 96% which demonstrates that

the generated sequences obtained via Lorenz and Rössler systems are totally random and pass the statistical TestU01. We conclude from Tables 7 and 8 that the produced numbers for each sequence of the chaotic system used in this algorithm are random and can pass each test in the NIST and TestU01 successfully.

**K. IMMUNITY TO ATTACKS ANALYSIS**

This subsection is devoted to explaining the immunity of the proposed approach to specific kinds of important attacks. Occlusion is the first attack on the images, which greatly affects the decrypted image at the receptor [27], [28], [44]. Different sized blocks are removed from the Peppers encrypted image (Fig. 5 (i)) in order to create several attacked encrypted images, as exemplified in Figs. 18 (a-e). Then, these attacked images are decrypted to obtain images as represented in Figs. 18 (f-j). Additionally, we have calculated the MSEs, PSNRs, SSIMs and correlation coefficients between the unencrypted and encrypted images, as



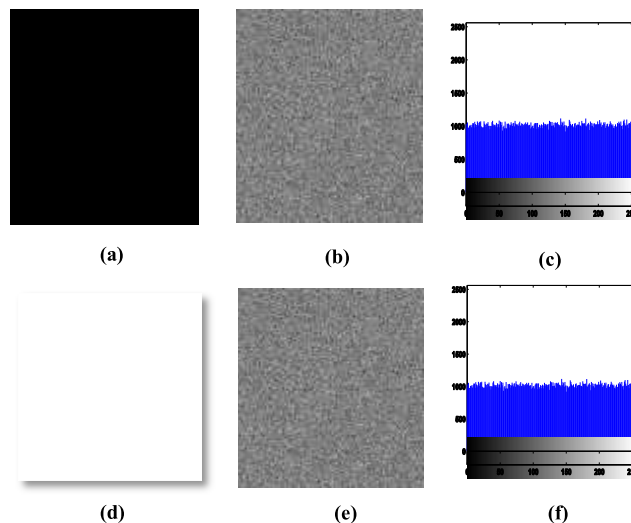
**FIGURE 18.** Robustness against occlusion attack (a) encrypted image with 1/8 data loss, (b) encrypted image with 1/4 data loss in the top left corner, (c) encrypted image with 1/4 data loss in the middle section, (d) encrypted image with 1/2 data loss in the bottom, (e) encrypted image with 1/2 data loss in the top, (f) restored image of (a), (g) restored image of (b), (h) restored image of (c), (i) restored image of (d), and (j) restored image of (e).

**TABLE 9.** MSEs, PSNRs, SSIMs and correlations between the input and restored images in presence of an occlusion attack.

Cropping size	MSE	PSNR (dB)	SSIM	$r_{xy}$
1/8 data loss	1265.2	17.1090	0.5352	0.7904
1/4 data loss in the top left corner	4401.9	11.6944	0.2479	0.5245
1/4 data loss in the middle section	4456.3	11.6410	0.2527	0.5207
1/2 data loss in the top	8804.9	8.6836	0.1284	0.3007
1/2 data loss in the bottom	8807.6	8.6822	0.1280	0.2984

shown in Table 9. Fig. 18 and Table 9 exhibit that the relation between the occlusion size and the quality of decrypted image is monotonically decreasing. It means that if the size of the occlusion increased, then the quality of the decrypted image decreased.

For example, when the percentage of occlusion is 1/8, then the MSE, PSNR, SSIM and  $r_{xy}$  values are 1265.2,



**FIGURE 19.** Results of chosen/known plaintext attack (a) chosen black image, (b) encrypted black image, (c) histogram of encrypted black image, (d) Chosen white image, (e) encrypted white image, and (f) histogram of encrypted white image.

17.1090 dB, 0.5352 and 0.7904, respectively. However, these metrics become 8807.6, 8.6822 dB, 0.1280 and 0.2984, respectively, when the occlusion becomes 1/2. Additionally, different cropping locations yield different results. As an example, cropping 1/2 in the top of the encrypted image gives 8804.9, 8.6836 dB, 0.1284 and 0.3007, respectively, for the MSE, PSNR, SSIM and  $r_{xy}$ , while cropping 1/2 in the bottom of the encrypted image yields 8807.6, 8.6822 dB, 0.1280, and 0.2984, respectively, for the MSE, PSNR, SSIM and  $r_{xy}$ .

Chosen plaintext and known plaintext image attacks are serious attacks in cryptography. Usually, the opponent in these attacks chooses a special plain image, for instance, a white image or black image, to attack the image security schemes by eliminating confusion and diffusion functions to obtain the secret keys [9], [24], [54]. Those two attacks are applied on black and white input images as shown in Fig. 19 (a) and Fig. 19 (d), respectively. Additionally, the corresponding encrypted versions with their histograms are illustrated in Figs. 19 (b), (e), (c) and (f), respectively. In addition, Table 10 is used to present the entropy, MSE, PSNR, SSIM, UACI, NPCR, and correlation coefficient scores for the encrypted black and white images. We can notice from Fig. 19 and Table 10 that the obtained encrypted images are noisy and the distributions of their histograms are reasonably uniform. All the metric values in Table 10 prove that the proposed approach can successfully resist these two kinds of attacks. Noise is the most influential attack on encrypted images. Examples of these noises are Poisson, salt and pepper, and Gaussian, which have serious effects on the security of digital images [22]. If the information of an encrypted image is distorted by noise, then retrieving the decrypted image is notably difficult. To sum up, we can say that our proposed approach have good immunity against



TABLE 10. Results of the quality metrics for encrypted black and white images.

Image	Encrypted Image Global entropy	Encrypted Image Local entropy	MSE	PSNR (dB)	UACI (%)	SSIM	NPCR (%)	Correlation coefficient		
								H	V	D
Black	7.9993	7.9061	8804.9	8.6836	45.5125	0.0369	99.62	-0.0110	0.0202	0.0091
White	7.9993	7.9061	21446	4.8173	49.5629	0.00072	99.62	0.0192	0.0029	0.0062

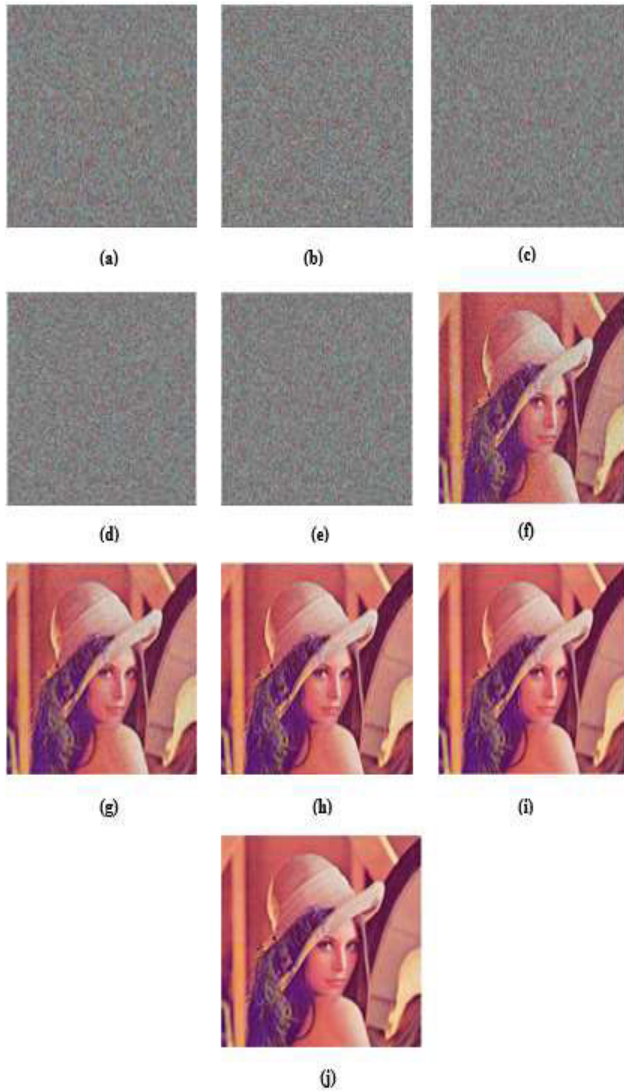


FIGURE 20. Encrypted and decrypted images under Gaussian noise attacks of different levels (a) encrypted image with 0.2 noise density, (b) encrypted image with 0.1 noise density, (c) encrypted image with 0.05 noise density, (d) encrypted image with 0.02 noise density, (e) encrypted image with 0.01 noise density, (f) decrypted image of a, (g) decrypted image of b, (h) decrypted image of c, (i) decrypted image of d, and (j) decrypted image of e.

several known attacks to the information security community and industry professionals.

In our experiments, Gaussian and salt and pepper noises are added to the encrypted image with various densities in order to analyse the capability of the proposed

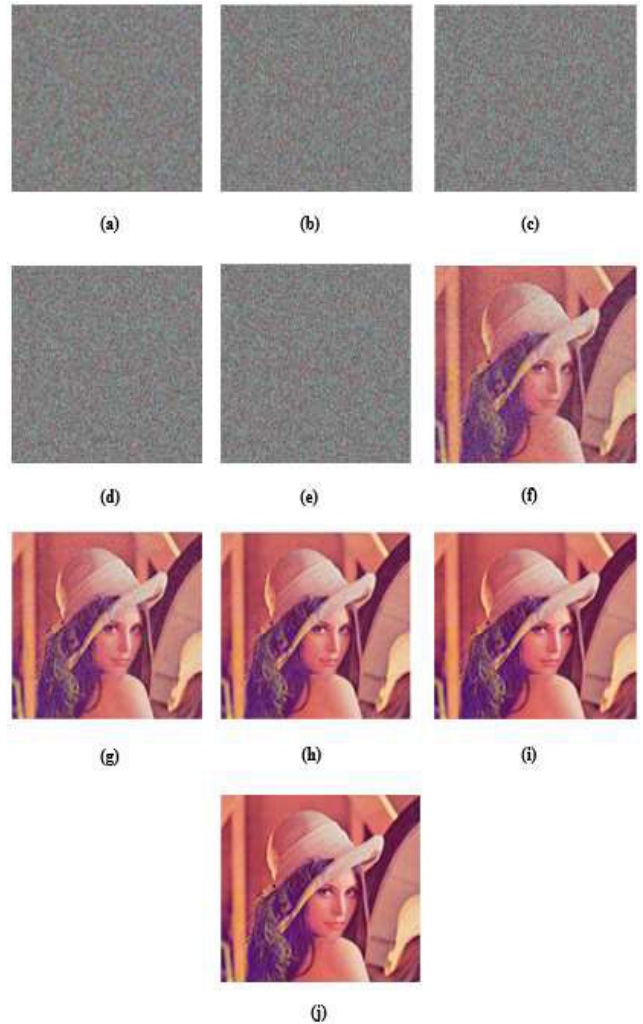


FIGURE 21. Encrypted and decrypted images under salt and pepper noise attack of different levels (a) encrypted image with 0.2 noise density, (b) encrypted image with 0.1 noise density, (c) encrypted image with 0.05 noise density, (d) encrypted image with 0.02 noise density, (e) encrypted image with 0.01 noise density, (f) decrypted image of a, (g) decrypted image of b, (h) decrypted image of c, (i) decrypted image of d, and (j) decrypted image of e.

method to resist a noise attack. The Colour Lena test image (Fig. 4 (g)) is employed in the experiment to study the impact of Gaussian and salt and pepper noises with various intensities (0.2, 0.1, 0.05, 0.02 and 0.01) on the encrypted images and their decrypted noisy results, as shown in Figs. 20 and 21, respectively. Furthermore, Table. 11 and Fig. 22 are used to show the MSEs, PSNRs, SSIMs and correlations between the

**TABLE 11.** Results of the MSEs, PSNRs, SSIMs, and correlations between restored and plain images under Gaussian, and salt and pepper noise attacks for the Lena colour image.

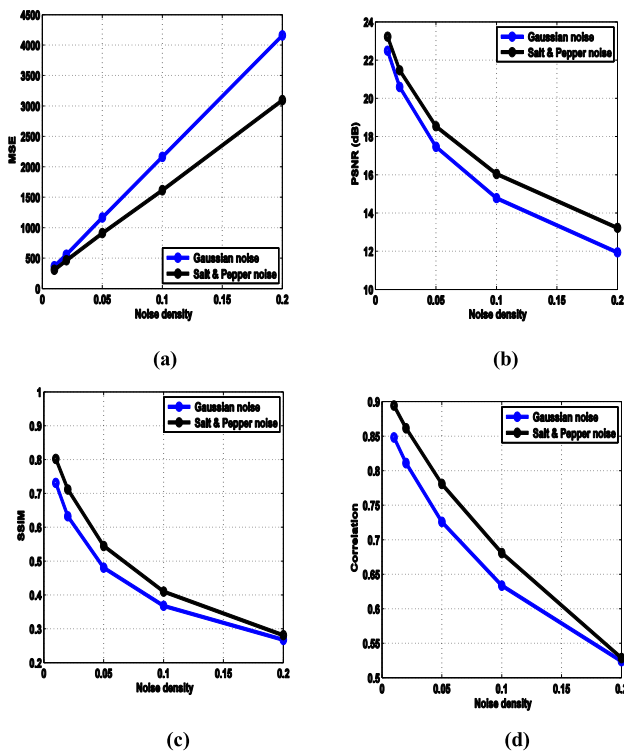
Gaussian noise				
Noise density	MSE	PSNR (dB)	SSIM	$r_{xy}$
0.2	4160.4	11.9395	0.2674	0.5241
0.1	2163.9	14.7785	0.3677	0.6336
0.05	1166.7	17.4650	0.4804	0.7259
0.02	556.7380	20.5970	0.6323	0.8110
0.01	366.5483	22.4895	0.7310	0.8483
Salt and pepper noise				
Noise density	MSE	PSNR (dB)	SSIM	$r_{xy}$
0.2	3095.8	13.2230	0.2812	0.5287
0.1	1617.4	16.0427	0.4101	0.6807
0.05	910.4740	18.5381	0.5449	0.7808
0.02	464.4735	21.4612	0.7117	0.8615
0.01	309.4399	23.2250	0.8022	0.8943

**TABLE 12.** Comparison results of MSEs, PSNRs and SSIMs for the presented method and similar approaches.

Image	Algorithm	MSE	PSNR (dB)	SSIM
Colour Lena	Ref. [19]	39.101	8.6728	0.0341
	Ref. [23]	8940.46	8.6170	0.0100
	Ref. [26]	-	8.583804	0.021284
	Ref. [50]	5812.68	10.2354	-
	Proposed	8955.3	8.6100	0.0345
Colour Baboon	Ref. [23]	8628.31	8.7715	0.0090
	Ref. [26]	-	8.781472	0.019615
	Proposed	8636.7	8.7673	0.0292
Peppers	Ref. [23]	10128.01	8.0756	0.0091
	Ref. [26]	-	8.079685	0.019063
	Proposed	10158	8.0628	0.0283
Airplane	Ref. [23]	10351.70	7.9807	0.0098
	Ref. [26]	-	7.957823	0.019587
	Proposed	10381	7.9683	0.0344
Splash	Ref. [23]	11238.79	7.6236	0.0089
	Ref. [26]	-	7.550227	0.020432
	Proposed	11238	7.6238	0.0317
Barbara	Ref. [23]	8526.60	8.8230	0.0089
	Ref. [26]	-	8.740601	0.018851
	Proposed	8911.4	8.6314	0.0286

**TABLE 13.** Comparison results of the key space for the presented method and similar approaches.

Algorithm	Key space
Ref. [2]	$2^{233}$
Ref. [9]	$10^{98}$
Ref. [15]	$2^{128}$
Ref. [17]	$10^{117}$
Ref. [18]	$10^{112}$
Ref. [19]	$2^{189}$
Ref. [21]	$2^{232}$
Ref. [23]	$2^{186}$
Ref. [25]	$2^{440}$
Ref. [26]	$2^{558}$
Ref. [29]	$2^{2041}$
Ref. [30]	$2^{249}$
Ref. [31]	$2^{384}$
Ref. [35]	$2^{428}$
Ref. [38]	$10^{56}$
Ref. [39]	$10^{160}$
Ref. [43]	$2^{138}$
Ref. [44]	$10^{90}$
Ref. [45]	$2^{256}$
Ref. [46]	$2^{273}$
Ref. [48]	$2^{256}$
Ref. [50]	$10^{70}$
Ref. [51]	$2^{256}$
Proposed	$2^{648}$



**FIGURE 22.** Results of the (a) MSE, (b) PSNR, (c) SSIM, (d) correlation between restored and plain images under Gaussian and, salt and pepper noise attacks for the Lena color image.

reconstructed and input images. From these results, it can be seen that the visual quality of the noisy decrypted images gradually increases as the noise intensity in the encrypted image decreases. For example, when the density of Gaussian noise is 0.2, then the MSE, PSNR, SSIM and correlation scores are 4160.4, 11.9395 dB, 0.2674 and 0.5241, respectively. Then, these scores become 366.5483, 22.4895 dB, 0.7310 and 0.8483, respectively, when the noise density is 0.01. On the other hand, when the salt and pepper noise

density is 0.2, then the MSE, PSNR, SSIM and correlation values are 3095.8, 13.2230 dB, 0.2812 and 0.5287, respectively. These measures then become 309.4399, 23.2250 dB, 0.8022 and 0.8943, respectively, when the noise density is 0.01. The above results show that our proposed approach can keep the basic information in the recovered image distinguishable in the presence of several kinds of noise with different intensities.

TABLE 14. Comparison results of the entropy, UACI, NPCR and correlation in the presented method and similar approaches.

Image	Algorithm	Global Entropy	UACI (%)	NPCR (%)	Correlation coefficient		
					H	V	D
Cameraman	Ref. [6]	7.9968	33.6015	99.6529	0.0001	-0.0015	-0.0001
	Ref. [18]	7.9993	33.5574	99.6082	0.0014	0.0002	0.0035
	Proposed	7.9994	33.3740	99.61	-0.0097	-0.0021	-0.0080
Gray Lena	Ref. [6]	7.9971	33.6548	99.6135	-0.0009	-0.0101	-0.0023
	Ref. [9]	7.9993	33.43	99.58	-0.0139	0.0177	0.0006
	Ref. [15]	7.9971	33.5053	99.5712	0.0015	0.0018	0.0018
	Ref. [18]	7.9994	33.5079	99.6002	0.0032	0.0016	0.0023
	Ref. [20]	7.99642	33.23	99.47	0.00051	-0.0043	0.00031
	Ref. [25]	7.999303	33.4330	99.6009	0.0027	0.0003	0.0012
	Ref. [28]	7.9992	33.39	99.60	-0.0685	0.0857	0.0059
	Ref. [31]	7.9993	33.4524	99.7990	-0.00032225	-0.00063715	-0.0071
	Ref. [35]	7.9993100	33.4436	99.6107	0.0013	0.0008	0.0066
	Ref. [42]	7.9993	33.4500	99.6200	-0.0045	-0.0001	0.0053
	Ref. [44]	7.9993	33.4500	99.6200	0.0044	0.0151	0.0012
	Ref. [48]	7.9993	33.4600	99.6200	-0.0285	0.0014	0.0013
Ref. [51]	-	-	-	0.0034	-0.0009	-0.0002	
Proposed	7.9993	33.4902	99.61	-0.0034	-0.0072	-0.0064	
Gray Baboon	Ref. [9]	7.9994	33.41	99.63	-0.0106	0.0036	0.0180
	Ref. [18]	7.9992	33.5281	99.5903	0.0029	0.0033	0.0062
	Ref. [20]	7.99645	33.21	99.36	-0.00245	-0.00314	0.00137
	Ref. [25]	-	33.4538	99.601	0.00005	0.0007	0.0051
	Ref. [35]	7.9992644	33.4308	99.6210	0.0050	0.0015	0.0016
	Ref. [42]	7.9993	33.4300	99.6100	-0.0108	0.0087	0.0058
	Ref. [44]	7.9993	33.4100	99.6300	-0.0058	0.0131	0.0030
	Ref. [48]	7.9993	33.4200	99.6100	0.0013	-0.0281	0.0128
	Ref. [50]	7.9993	33.4300	99.6100	-	-	-
	Proposed	7.9993	33.5480	99.61	-0.0026	-0.0062	-0.0015
Goldhill	Ref. [42]	7.9994	33.4100	99.6100	-0.0096	-0.0092	-0.0014
	Ref. [44]	7.9993	33.4700	99.6000	-0.0232	-0.0026	-0.0198
	Proposed	7.9993	33.0875	99.61	0.0021	-0.0117	0.0042
Lake	Ref. [20]	7.99681	33.18	99.46	0.00874	0.00812	0.00038
	Proposed	7.9993	35.5319	99.61	0.0074	-0.0060	-0.0144
Color Lena	Ref. [1]	7.999772	33.447	99.619	0.0137	0.0024	-0.0035
	Ref. [19]	-	33.45	99.61	0.000187	0.000592	-0.000736
	Ref. [23]	7.9994	33.4488	99.6037	-0.0043	-0.0007	0.0030
	Ref. [24]	7.9993	-	-	-0.0034	0.00193	-0.0134
	Ref. [26]	7.9992	33.3354	99.8241	0.0004	0.0003	-0.0005
	Ref. [29]	7.99980	33.4397	99.6086	-	-	-
	Ref. [35]	-	33.2404	99.6249	-	-	-
Proposed	7.9997	33.4573	99.60	-0.0021	-0.0030	-0.0177	
Color Baboon	Ref. [1]	7.999778	33.457	99.609	-0.0192	-0.0151	0.0138
	Ref. [23]	7.9993	33.4470	99.6147	-0.0084	-0.0015	-0.0024
	Ref. [26]	7.9991	33.3452	99.7885	-0.0009	-0.0007	0.0006
	Ref. [35]	-	33.3056	99.7091	-	-	-
Proposed	7.9997	33.3875	99.60	-0.0081	-0.0011	-0.0065	
Peppers	Ref. [1]	7.999788	33.466	99.608	0.0012	0.0059	-0.0046
	Ref. [23]	7.9994	33.4600	99.6040	0.0006	0.0065	-0.0011
	Ref. [24]	7.9994	-	-	-0.00083	-0.00006	0.00293
	Ref. [26]	7.9989	33.3321	99.8446	-0.0011	0.0013	0.0015
	Ref. [35]	-	33.3247	99.7065	-	-	-
Proposed	7.9997	33.5503	99.61	-0.0096	-0.0094	0.0039	
Airplane	Ref. [23]	7.9993	33.5169	99.6243	0.0108	0.0015	0.0049
	Ref. [26]	7.9989	33.3378	99.7903	0.0012	0.0009	0.0014
	Proposed	7.9995	36.9962	99.61	-0.0206	-0.0086	-0.0099
Splash	Ref. [23]	7.9993	33.4268	99.6185	-0.0057	-0.0024	0.0114
	Ref. [26]	7.9991	33.3540	99.7910	0.0018	0.0014	0.0021
	Ref. [35]	-	33.2210	99.7599	-	-	-
Proposed	7.9997	33.4687	99.61	0.0061	-0.0238	-0.0104	
Barbara	Ref. [23]	7.9994	33.4762	99.6021	-0.0014	0.0075	0.0116
	Ref. [26]	7.9992	33.3417	99.7369	-0.0036	-0.0029	0.0018
	Proposed	7.9996	33.4006	99.61	0.0069	-0.0187	-0.0216

**TABLE 15.** Comparison results of the encryption time for the presented method and similar approaches.

Image	Size	Algorithm	Encryption time (s)
Camera man	(256 × 256)	Ref. [22]	0.99
		Ref. [49]	0.667
		Proposed	0.2388
Gray Lena	(256 × 256)	Ref. [22]	0.93
		Ref. [42]	0.46
		Ref. [49]	0.613
		Proposed	0.2378
		Ref. [20]	19.2
		Ref. [28]	2.792
		Ref. [42]	1.26
		Proposed	1.2301
Gray Baboon	(512 × 512)	Ref. [20]	18.7
		Proposed	1.0436
Lake	(512 × 512)	Ref. [20]	19.0
		Proposed	1.2293
Color Lena	(256 × 256)	Ref. [39]	2.12
		Proposed	0.7858
Color Lena	(512 × 512)	Ref. [26]	3.768142
		Ref. [39]	8.41
		Proposed	3.3843
Color Lena	(1024 × 1024)	Ref. [39]	33.22
		Proposed	14.2093
Peppers	(512 × 512)	Ref. [26]	3.741908
		Proposed	2.2079
Airplane	(512 × 512)	Ref. [26]	3.783167
		Proposed	2.6410
Splash	(512 × 512)	Ref. [26]	3.796834
		Proposed	3.2181

## VI. COMPARATIVE ANALYSIS

In this subsection, we present a comparison between our approach and the state of art approaches in terms of the key space size, entropy, UACI, NPCR, correlation coefficient, MSE, PSNR, SSIM, and total encryption time. The comparative results are shown in the Tables (12-16) respectively. Table 12 shows the MSE and PSNR metric scores. The results indicates that the presented method achieves larger MSE scores and smaller PSNR scores in comparison with Refs. [19] and [55] for colour Lena image. However, the results of those two metrics in the [23] and [26] are very close for all compared images. Also, the earlier mentioned approaches achieved lower SSIM values than our proposed approach. The key space size is straight proportional to the number of initial conditions and control parameters of the chaotic system utilized in the algorithm. Or in another word, the cryptosystem security relies on the secret key security [31]. Therefore, using Lorenz and Rössler systems in the proposed scheme yields in increasing the key size space as shown from the results of the key space comparison in Table 13. This table reveals that the proposed approach possesses the largest key space size among the compared approaches except for [29]. Table 14 show the global entropy, UACI, NPCR and correlations in three directions. The results in the Table 14 reveals that the entropy scores of our approach are nearly close to those obtained by other approaches except

**TABLE 16.** Comparison results of the PSNR and SSIM for the presented method and similar approaches under a Gaussian noise attack.

Gaussian Noise density	PSNR (dB)		SSIM	
	Proposed	Algorithm	Proposed	Algorithm
0.2	11.9395	11.25	0.2674	0.0661
0.1	14.7785	12.99	0.3677	0.0851
0.05	17.4650	-	0.4804	-
0.02	20.5970	15.41	0.6323	0.1983
0.01	22.4895	16.61	0.7310	0.2077

for [18]. Besides, this approach records better values for UACI and correlations than other references and higher results in terms of NPCR except for [31], [42], [44] and [48] for gray Lena image. For gray Baboon image in Table 14, this technique achieves same results as the other methods except for [9] in the means of global entropy and better scores in terms of UACI and correlation coefficients. Also, it achieves higher or equal to those results obtained via other methods except for [9], [35] and [44] in terms of NPCR test. For Goldhill image in Table 14, this scheme has larger values in terms of global entropy except for [42] and it got better scores values in the means of NPCR and correlations along three directions. For Lake image in Table 14, the proposed algorithm attains better results than [20] in terms of entropy, UACI, NPCR and correlation coefficients. For color Lena image in Table 14, the gained outcomes are better in terms of global entropy except for [29]. Further, it attains higher scores in terms of UACI and correlations, but the techniques in [1], [19], [26] and [35] achieve better outcomes in terms of NPCR. For color Baboon image in Table 14, this image cryptosystem gets better entropy and correlation values than the current references, but the obtained NPCR values in [23], [26] and [35] are better. For Peppers image in Table 14, the entropy, UACI and correlation scores acquired via this method are larger whilst Refs. 26 and 35 attain larger NPCR scores. Finally, For Airplane, Splash, and Barbara images in Table 14, this mechanism has higher entropy, UACI, and correlation values except for the NPCR results. The presented scheme possesses the shortest encryption time than compared methods as illustrated in Table 15. The comparison of the PSNRs and SSIMs under a Gaussian noise attack of different densities is exhibited in Table 16. The larger SSIMs and PSNRs in this table manifest that the introduced cryptosystem can restore the decrypted image under a noise attack with better quality. Finally, Tables (12-16) provide us with sufficient evidence that the proposed approach for image security is an efficient and fast image encryption method.

As a future work, we suggest to combine other information security technologies with the proposed chaos based cryptography approach. Such technologies may be biometrics, multifactor authentication techniques, deep learning algorithms etc. Such merged technologies provide reliable solutions in very vulnerable environments for different kinds of threats and attacks.



## VII. CONCLUSION

This research introduces a novel image security approach to protect digital images by utilizing a scan mechanism, the El-Gamal asymmetric key cryptosystem and chaotic systems. The scrambling phase yields a higher shuffling level to improve the image security in terms of the confusion and diffusion phases. The Lorenz chaotic system is adopted for the resulting permuted images to confuse the relation between the grey image pixels by changing their locations, whereas the Rössler chaotic system is applied to the obtained confused images to diffuse the image pixels through varying their values to gain a second layer of encryption. The visual and numerical results prove that the proposed approach can effectively resist statistical, differential, and exhaustive attacks and it has potent immunity to most widespread attacks. In future work, we will use other kinds of chaotic systems, cryptographic algorithms and biometrics to make stronger cryptosystems that can resist most attacks.

## ACKNOWLEDGMENT

The authors would like to thank the University of Diyala and the College of Engineering for supporting our research by providing all facilities that were necessary to successfully accomplish this research.

## REFERENCES

- [1] C. Fu, G.-Y. Zhang, M. Zhu, Z. Chen, and W.-M. Lei, "A new chaos-based color image encryption scheme with an efficient substitution keystream generation strategy," *Secur. Commun. Netw.*, vol. 2018, pp. 1–13, Jan. 2018.
- [2] Q. Zhang, L. Guo, and X. Wei, "A novel image fusion encryption algorithm based on DNA sequence operation and hyper-chaotic system," *Optik-Int. J. Light Electron Opt.*, vol. 124, no. 18, pp. 3596–3600, Sep. 2013.
- [3] J. Ali Abboud, "Multifactor authentication for software protection," *Diyala J. Eng. Sci.*, vol. 8, no. 4, pp. 479–492, 2015.
- [4] A. J. Abboud, A. N. Albu-Rghaif, and A. K. Jassim, "Balancing compression and encryption of satellite imagery," *Int. J. Electr. Comput. Eng.*, vol. 8, no. 5, p. 3568, Oct. 2018.
- [5] A. J. L. Abboud and A. S. Jassim, "Image quality guided approach for adaptive modelling of biometric intra-class variations," *Proc. SPIE, Int. Soci. Opt. Eng.*, vol. 7708, Apr. 2010, Art. no. 77080L.
- [6] R. M. Rad, A. Attar, and R. E. Atani, "A new fast and simple image encryption algorithm using scan patterns and XOR," *Int. J. Signal Process., Image Process. Pattern Recognit.*, vol. 6, no. 5, pp. 275–290, Oct. 2013.
- [7] H. Al-Assam, J. Ali Abboud, H. Sellaheewa, and S. Jassim, "Exploiting relative entropy and quality analysis in cumulative partial biometric fusion," in *Transactions on Data Hiding and Multimedia Security VIII*. Berlin, Germany: Springer, 2012, pp. 1–18.
- [8] J. A. Abboud and A. S. Jassim, "Incremental fusion of partial biometric information," in *Proc. SPIE, Mobile Multimedia/Imag. Process., Secur., Appl. Int. Soc. Opt. Photon.*, vol. 8406, May 2012, Art. no. 84060K.
- [9] X. Chai, Z. Gan, K. Yuan, Y. Chen, and X. Liu, "A novel image encryption scheme based on DNA sequence operations and chaotic systems," *Neural Comput. Appl.*, vol. 31, no. 1, pp. 219–237, 2019.
- [10] A. S. Hameed, "Image encryption based on fractional order Lorenz system and wavelet transform," *Diyala J. Eng. Sci.*, vol. 10, no. 6, pp. 81–91, Mar. 2017.
- [11] J. N. Shehab, H. Y. Radhi, and R. A. Ibrahim, "Multimedia cryptography based on Liu and Chen systems," *Diyala J. Eng. Sci.*, vol. 9, no. 4, pp. 24–35, Dec. 2016.
- [12] X. Deng, C. Liao, C. Zhu, and Z. Chen, "A novel image encryption algorithm based on hyperchaotic system and shuffling scheme," in *Proc. IEEE 10th Int. Conf. High Perform. Comput. Commun. IEEE Int. Conf. Embedded Ubiquitous Comput.*, Nov. 2013, pp. 109–116.
- [13] S. Vashisth, H. Singh, A. K. Yadav, and K. Singh, "Image encryption using fractional Mellin transform, structured phase filters, and phase retrieval," *Optik-Int. J. Light Electron Opt.*, vol. 125, no. 18, pp. 5309–5315, Sep. 2014.
- [14] M. Khan, "A novel image encryption scheme based on multiple chaotic S-boxes," *Nonlinear Dyn.*, vol. 82, nos. 1–2, pp. 527–533, Oct. 2015.
- [15] H. Niu, C. Zhou, B. Wang, X. Zheng, and S. Zhou, "Splicing model and hyper-chaotic system for image encryption," *J. Electr. Eng.*, vol. 67, no. 2, pp. 78–86, Apr. 2016.
- [16] X. Chai, Z. Gan, Y. Chen, and Y. Zhang, "A visually secure image encryption scheme based on compressive sensing," *Signal Process.*, vol. 134, pp. 35–51, May 2017.
- [17] J. Wu, X. Liao, and B. Yang, "Color image encryption based on chaotic systems and elliptic curve ElGamal scheme," *Signal Process.*, vol. 141, pp. 109–124, Dec. 2017.
- [18] J. Wu, X. Liao, and B. Yang, "Image encryption using 2D Hénon-sine map and DNA approach," *Signal Process.*, vol. 153, pp. 11–23, Dec. 2018.
- [19] M. L. Sahari and I. Boukemara, "A pseudo-random numbers generator based on a novel 3D chaotic map with an application to color image encryption," *Nonlinear Dyn.*, vol. 94, no. 1, pp. 723–744, 2018.
- [20] Nasrullah, J. Sang, M. A. Akbar, B. Cai, H. Xiang, and H. Hu, "Joint image compression and encryption using IWT with SPIHT, kd-tree and chaotic maps," *Appl. Sci.*, vol. 8, no. 10, p. 1963, Oct. 2018.
- [21] X. Chai, X. Zheng, Z. Gan, D. Han, and Y. Chen, "An image encryption algorithm based on chaotic system and compressive sensing," *Signal Process.*, vol. 148, pp. 124–144, Jul. 2018.
- [22] X. Zhang, L. Wang, Z. Zhou, and Y. Niu, "A chaos-based image encryption technique utilizing Hilbert curves and H-Fractals," *IEEE Access*, vol. 7, pp. 74734–74746, 2019.
- [23] B. Y. Irani, P. Ayubi, F. A. Jabalkandi, M. Y. Valandar, and M. J. Barani, "Digital image scrambling based on a new one-dimensional coupled Sine map," *Nonlinear Dyn.*, vol. 97, no. 2, pp. 1–29, 2019.
- [24] X.-Y. Wang and Z.-M. Li, "A color image encryption algorithm based on Hopfield chaotic neural network," *Opt. Lasers Eng.*, vol. 115, pp. 107–118, Apr. 2019.
- [25] X. Wang, L. Feng, R. Li, and F. Zhang, "A fast image encryption algorithm based on non-adjacent dynamically coupled map lattice model," *Nonlinear Dyn.*, vol. 95, no. 4, pp. 2797–2824, 2019.
- [26] M. Y. Valandara, M. J. Barania, and P. Ayubib, "A fast color image encryption technique based on three dimensional chaotic map," *Optik-Int. J. Light Electron Opt.*, vol. 193, pp. 1–17, Sep. 2019.
- [27] S. Sun, Y. Guo, and R. Wu, "A novel plaintext-related image encryption algorithm based on stochastic signal insertion and block swapping," *IEEE Access*, vol. 7, pp. 123049–123060, 2019.
- [28] Z. Tang, Y. Yang, S. Xu, C. Yu, and X. Zhang, "Image encryption with double spiral scans and chaotic maps," *Secur. Commun. Netw.*, vol. 2019, pp. 1–15, Jan. 2019.
- [29] A. Flores-Vergara, E. E. García-Guerrero, E. Inzunza-González, O. R. López-Bonilla, E. Rodríguez-Orozco, J. R. Cárdenas-Valdez, and E. Tlelo-Cuautle, "Implementing a chaotic cryptosystem in a 64-bit embedded system by using multiple-precision arithmetic," *Nonlinear Dyn.*, vol. 96, no. 1, pp. 497–516, Apr. 2019.
- [30] E. E. García-Guerrero, E. Inzunza-González, O. R. López-Bonilla, J. R. Cárdenas-Valdez, and E. Tlelo-Cuautle, "Randomness improvement of chaotic maps for image encryption in a wireless communication scheme using PIC-microcontroller via Zigbee channels," *Chaos, Solitons Fractals*, vol. 133, Apr. 2020, Art. no. 109646.
- [31] T. Li, J. Shi, X. Li, J. Wu, and F. Pan, "Image encryption based on pixel-level diffusion with dynamic filtering and DNA-level permutation with 3D Latin cubes," *Entropy*, vol. 21, no. 3, p. 319, Mar. 2019.
- [32] S. Somaraj and M. A. Hussain, "An image encryption technique using scan based approach and image as key," in *Proc. 1st Int. Conf. Comput. Intell. Informatics.*, Singapore, 2017, pp. 645–653.
- [33] R. Candra, S. Madenda, S. A. Sudiro, and M. Subali, "The implementation of an efficient zigzag scan," *J. Telecommun., Electron. Comput. Eng.*, vol. 9, no. 2, pp. 95–98, 2017.
- [34] T. Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Inf. Theory*, vol. 31, no. 4, pp. 469–472, Jul. 1985.
- [35] Z. Li, C. Peng, L. Li, and X. Zhu, "A novel plaintext-related image encryption scheme using hyper-chaotic system," *Nonlinear Dyn.*, vol. 94, no. 2, pp. 1319–1333, Oct. 2018.

- [36] F. Abundiz-Pérez, C. Cruz-Hernández, M. A. Murillo-Escobar, R. M. López-Gutiérrez, and A. Arellano-Delgado, "A fingerprint image encryption scheme based on hyperchaotic Rössler map," *Math. Problems Eng.*, vol. 2016, pp. 1–15, Jan. 2016.
- [37] M. SaberiKamarpshiti, D. Mohammad, M. S. M. Rahim, and M. Yaghobi, "Using 3-cell chaotic map for image encryption based on biological operations," *Nonlinear Dyn.*, vol. 75, no. 3, pp. 407–416, Feb. 2014.
- [38] G. Ye, C. Pan, X. Huang, and Q. Mei, "An efficient pixel-level chaotic image encryption algorithm," *Nonlinear Dyn.*, vol. 94, no. 1, pp. 745–756, Oct. 2018.
- [39] X. Zhang and X. Wang, "Multiple-image encryption algorithm based on the 3D permutation model and chaotic system," *Symmetry*, vol. 10, no. 11, p. 660, Nov. 2018.
- [40] Z. Tang, F. Wang, and X. Zhang, "Image encryption based on random projection partition and chaotic system," *Multimedia Tools Appl.*, vol. 76, no. 6, pp. 8257–8283, Mar. 2017.
- [41] X.-J. Tong, M. Zhang, Z. Wang, Y. Liu, H. Xu, and J. Ma, "A fast encryption algorithm of color image based on four-dimensional chaotic system," *J. Vis. Commun. Image Represent.*, vol. 33, pp. 219–234, Nov. 2015.
- [42] X. Chai, Y. Chen, and L. Broyde, "A novel chaos-based image encryption algorithm using DNA sequence operations," *Opt. Lasers Eng.*, vol. 88, pp. 197–213, Jan. 2017.
- [43] C. Pak and L. L. Huang, "A new color image encryption using combination of the 1D chaotic map," *Signal Process.*, vol. 138, pp. 129–137, Sep. 2017.
- [44] X. Chai, Z. Gan, and M. Zhang, "A fast chaos-based image encryption scheme with a novel plain image-related swapping block permutation and block diffusion," *Multimedia Tools Appl.*, vol. 76, no. 14, pp. 15561–15585, Jul. 2017.
- [45] W. Liu, K. Sun, and C. Zhu, "A fast image encryption algorithm based on chaotic map," *Opt. Lasers Eng.*, vol. 84, pp. 26–36, Sep. 2016.
- [46] Y. Li, C. Wang, and H. Chen, "A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation," *Opt. Lasers Eng.*, vol. 90, pp. 238–246, Mar. 2017.
- [47] Y. Zhang and Y. Tang, "A plaintext-related image encryption algorithm based on chaos," *Multimedia Tools Appl.*, vol. 77, no. 6, pp. 6647–6669, Mar. 2018.
- [48] X. Chai, "An image encryption algorithm based on bit level Brownian motion and new chaotic systems," *Multimedia Tools Appl.*, vol. 76, no. 1, pp. 1159–1175, Jan. 2017.
- [49] Y. Wu, J. P. Noonan, and S. Agaian, "NPCR and UACI randomness tests for image encryption," *Cyber J., Multidisciplinary J. Sci. Technol., J. Select. Areas Telecomm.*, vol. 1, pp. 31–38, Apr. 2011.
- [50] X. Chai, K. Yang, and Z. Gan, "A new chaos-based image encryption algorithm with dynamic key selection mechanisms," *Multimedia Tools Appl.*, vol. 76, no. 7, pp. 9907–9927, Apr. 2017.
- [51] Z. Hua and Y. Zhou, "Design of image cipher using block-based scrambling and image filtering," *Inf. Sci.*, vol. 396, pp. 97–113, Aug. 2017.
- [52] L. Xu, X. Gou, Z. Li, and J. Li, "A novel chaotic image encryption algorithm using block scrambling and dynamic index based diffusion," *Opt. Lasers Eng.*, vol. 91, pp. 41–52, Apr. 2017.
- [53] P. L'Ecuyer and R. Simard, "TestU01: A C library for empirical testing of random number generators," *ACM Trans. Math. Softw.*, vol. 33, no. 4, pp. 1–22, 2007.
- [54] S. Zhu and C. Zhu, "Plaintext-related image encryption algorithm based on block structure and five-dimensional chaotic map," *IEEE Access*, vol. 7, pp. 147106–147118, 2019.
- [55] M. Khan and T. Shah, "A novel statistical analysis of chaotic S-box in image encryption," *3D Res.*, vol. 5, no. 3, pp. 1–8, Sep. 2014.



**SURA F. YOUSIF** (Member, IEEE) received the B.Sc. degree in electronics engineering from the University of Diyala, Iraq, in 2004, and the M.Sc. degree in electronic and communication engineering from Al-Mustansiriyah University, Iraq, in 2015. She has been teaching (assistant lecturer) with the Department of Chemical Engineering, College of Engineering, University of Diyala, since 2015. Her research interests include secure wireless communication systems, digital image processing, information hiding, chaotic signals, and multimedia cryptography.



**ALI J. ABBOD** was born in Diyala, Iraq, in 1978. He received the B.Sc. degree in computer and software engineering from Al-Mustansiriyah University, Iraq, in 2001, the M.Sc. degree in computer engineering from the University of Technology, Iraq, in 2005, and the Ph.D. degree in computer science from the University of Buckingham, U.K., in 2011. He is working currently as an Associate Professor with the Department of Computer Engineering with the University of Diyala. His research interests include image processing, computer vision, biometrics, machine learning, pattern recognition, and information security.



**HUSSEIN Y. RADHI** received the B.Sc. degree in electronics engineering from the College of Engineering, University of Diyala, Iraq, in 2003, and the M.Sc. degree in electronics and communication from the College of Engineering, Al-Mustansiriyah University, Iraq, in 2014. He is currently working as an Assistant Lecturer with the Department of Computer Engineering, University of Diyala. His research interests include cryptography, wireless sensor security, image processing, and intelligent systems.

...