Journal of Advanced Research (2016) **7**, 193–208

Cairo University

Journal of Advanced Research

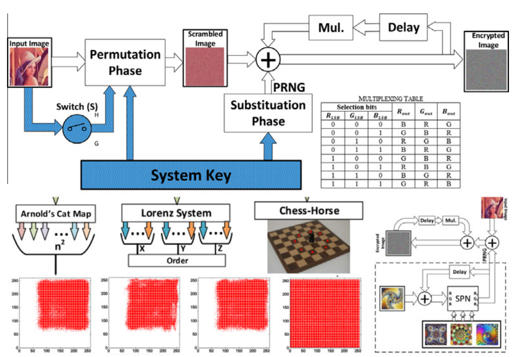
REVIEW

Symmetric encryption algorithms using chaotic and non-chaotic generators: A review

Ahmed G. Radwan [a,b,*], Sherif H. AbdElHaleem [a], Salwa K. Abd-El-Hafiz [a]

[a] *Engineering Mathematics Department, Faculty of Engineering, Cairo University, Giza 12613, Egypt*
[b] *Nanoelectronics Integrated Systems Center (NISC), Nile University, Cairo, Egypt*

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 27 May 2015
Received in revised form 24 July 2015
Accepted 27 July 2015
Available online 1 August 2015

ABSTRACT

This paper summarizes the symmetric image encryption results of 27 different algorithms, which include substitution-only, permutation-only or both phases. The cores of these algorithms are based on several discrete chaotic maps (Arnold's cat map and a combination of three generalized maps), one continuous chaotic system (Lorenz) and two non-chaotic generators (fractals and chess-based algorithms). Each algorithm has been analyzed by the correlation coefficients

* Corresponding author. Tel.: +20 1224647440; fax: +20 235723486.
E-mail address: agradwan@ieee.org (A.G. Radwan).

Peer review under responsibility of Cairo University.

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.jare.2015.07.002

Keywords:
Permutation matrix
Symmetric encryption
Chess
Chaotic map
Fractals

between pixels (horizontal, vertical and diagonal), differential attack measures, Mean Square Error (MSE), entropy, sensitivity analyses and the 15 standard tests of the National Institute of Standards and Technology (NIST) SP-800-22 statistical suite. The analyzed algorithms include a set of new image encryption algorithms based on non-chaotic generators, either using substitution only (using fractals) and permutation only (chess-based) or both. Moreover, two different permutation scenarios are presented where the permutation-phase has or does not have a relationship with the input image through an ON/OFF switch. Different encryption-key lengths and complexities are provided from short to long key to persist brute-force attacks. In addition, sensitivities of those different techniques to a one bit change in the input parameters of the substitution key as well as the permutation key are assessed. Finally, a comparative discussion of this work versus many recent research with respect to the used generators, type of encryption, and analyses is presented to highlight the strengths and added contribution of this paper.

Ahmed G. Radwan (M'96–SM'12) received the B.Sc. degree in Electronics, and the M.Sc. and Ph.D. degrees in Eng. Mathematics from Cairo University, Egypt, in 1997, 2002, and 2006, respectively. He is an Associate Professor, Faculty of Engineering, Cairo University, and also the Director of Nanoelectronics Integrated Systems Center, Nile University, Egypt. From 2008 to 2009, he was a Visiting Professor in the ECE Dept., McMaster University, Canada. From 2009 to 2012, he was with King Abdullah University of Science and Technology (KAUST), Saudi Arabia. His research interests include chaotic, fractional order, and memristor-based systems. He is the author of more than 140 international papers, six USA patents, three books, two chapters, and h-index = 17.

Dr. Radwan was awarded the Egyptian Government first-class medal for achievements in the field of Mathematical Sciences in 2012, the Cairo University achievements award for research in the Engineering Sciences in 2013, and the Physical Sciences award in the 2013 International Publishing Competition by Misr El-Khair Institution. He won the best paper awards in many international conferences as well as the best thesis award from the Faculty of Engineering, Cairo University. He was selected to be among the first scientific council of Egyptian Young Academy of Sciences (EYAS), and also in first scientific council of the Egyptian Center for the Advancement of Science, Technology and Innovation (ECASTI).

Sherif H. AbdElHaleem received the B.Sc. degree in Electronics and Communication Engineering, a Diploma in Automatic Control and the M.Sc. degree in Engineering Mathematics from the Faculty of Engineering, Cairo University, in 2002, 2004 and 2015, respectively. From 2004 to 2015, he has been working as a professional software developer in ASIE. His research and work interests include software development, database applications, network programming, web developing and cryptography. As part of his M.Sc. work, Eng. AbdElHaleem has published several refereed papers on image encryption.

Salwa K. Abd-El-Hafiz received the B.Sc. degree in Electronics and Communication Engineering from Cairo University, Egypt, in 1986 and the M.Sc. and Ph.D. degrees in Computer Science from the University of Maryland, College Park, Maryland, USA, in 1990 and 1994, respectively. Since 1994, she has been working as a Faculty Member in the Engineering Mathematics and Physics Department, Faculty of Engineering, Cairo University, and has been promoted to a Full Professor in the same department in 2004. Since August 2014, she has also been working as the Director of the Technical Center for Job Creation, Cairo University, Egypt. She co-authored one book, contributed one chapter to another book and published more than 60 refereed papers. Her research interests include software engineering, computational intelligence, numerical analysis, chaos theory and fractal geometry.

Prof. Abd-El-Hafiz is a recipient of the 2001 Egyptian State Encouragement Prize in Engineering Sciences, recipient of the 2012 National Publications Excellence Award from the Egyptian Ministry of Higher Education, recipient of the 2014 African Union Kwame Nkrumah Regional Scientific Award for Women in basic science, technology and innovation, recipient of several international publications awards from Cairo University and an IEEE Senior Member.

Introduction

Symmetric encryption algorithms can be classified into stream ciphers and block ciphers where the image-pixels are encrypted one-by-one in stream ciphers and using blocks of bits in block ciphers. Although block ciphers require more hardware and memory, their performance is generally superior to stream ciphers since they have a permutation phase as well as a substitution phase. As suggested by Shannon, plaintext should be processed by two main substitution and permutation phases to accomplish the confusion and diffusion properties [1,2].

The target of the permutation process is to weaken the correlations of input plaintext by spreading the plaintext bits throughout the cipher text. On the other hand, the substitution process target is to decrease the relation between the plaintext and the ciphertext through nonlinear operations and a pseudo random number generator (PRNG). PRNG's can be designed by using chaotic systems or based on fractal shapes [3–5]. Recently, many fractional-order chaotic systems have also been introduced to increase the design flexibility by the added non-integer parameters [6,7].

Due to the high sensitivity of chaotic systems to parameters and initial conditions as well as the availability of many circuit realizations [8,9], chaos based algorithms are developed and studied as the core of encryption algorithms. Recently, many substitution-only encryption algorithms have been introduced based on discrete 1-D chaotic maps such as the conventional logistic map [10–12] and the conventional tent map [13], or discrete 2-D chaotic maps such as the coupled map lattice [14]. Such encryption algorithms cover the encryption of text-messages, grayscale and color images. In order to improve the encryption process, both substitution and permutation phases were used based on the conventional logistic map [15], the Gray code [16] and a 2-D hyper-chaos discrete nonlinear dynamic system with the Chinese reminder theorem [17] where compression performance was discussed. The use of conventional 1-D and 2-D discrete maps in substitution and permutation phases with noise analysis was introduced in [18,19]. Similarly the encryption algorithm can be achieved using other higher order discrete maps such as the 3D Baker map [20] and the 3D Arnold's cat map [21]. Zhang et al. [22] used an expand-and-shrink strategy to shuffle the image with reconstructed permuting plane. Furthermore, Sethi and Vijay [23] introduced two phases to encrypt the image, whereas in [24] four different chaotic maps were used in generating sub-keys, and the logistic map and the Arnold's cat map were used in [25–29].

On the other hand, non-chaotic methods have proved their existence and importance in implementing the confusion and diffusion stages. Such methods usually increase the algorithm complexity to protect against cryptanalysis. For instance, Wu et al. [30] used the Latin squares algorithm to design a new 2D substitution–permutation network. Pareek et al. [31] divided the image into non-overlapping blocks and each block was scrambled using a zigzag-like algorithm. Furthermore, [32] divided the image into a set of k-bit vectors; each of these vectors was substituted by XORing it with the previous vector and then permuted by circularly right rotating its bits. Alternatively, Pareek et al. [33] divided the image into non-overlapping blocks and for each encryption round the size of the block changed according to the round key. Within the same block, permutation was performed using a zigzag-like algorithm.

The combination of both chaotic and non-chaotic algorithms showed some advantages in many cryptosystems. For example, Li and Liu [34] used the 3D Arnold map and a Laplace-like equation to perform permutations and substitutions, respectively. Wang and Yang [35] used the water drop motion and a dynamic lookup table with the help of the logistic map to perform the diffusion and confusion processes. Furthermore, Fouda et al. [36] used a piecewise linear chaotic map to generate pseudo random numbers and these numbers were used in generating the coefficients of the Linear Diophantine Equation (LDE). By sorting the solutions of LDE, large permutations were created and used in scrambling the image pixels. Whereas Zhang and Zhou [37] used compressive sensing along with Arnold's map in order to encrypt color images into gray images, Zhang and Xiao [38] used a coupled logistic map, self-adaptive permutation, substitution-boxes and combined global diffusion to perform the encryption. Finally, AbdElHaleem et al. [39] used a chess-based algorithm to perform the permutation process and the Lorenz system to perform the substitution process. In summary, permutations and substitutions can be performed using chaotic systems, non-chaotic algorithms or a combination of both.

Although many encryption algorithms have been published during the last few decades but, up till now, there is no completely non-chaotic image encryption algorithm that can pass all NIST-tests and produce good analysis results. Therefore, three different algorithms (discrete chaos, continuous chaos and non-chaotic algorithms) have been selected for the substitution phase and another three algorithms (discrete chaos, continuous chaos and non-chaotic algorithms) for the permutation phase. The effect of the input image on all encryption algorithms has been investigated by adding a switch that affects the permutation phase. Complete analyses of 27 encryption algorithms are presented with their sensitivity analyses and comparisons with recent papers.

Section 'Encryption key and evaluation criteria' of this paper describes the fundamentals of the encryption key and the standard statistical and sensitivity evaluation criteria. In section 'Substitution-only encryption algorithm', three substitution methods are discussed, based on discrete chaotic maps, a continuous chaotic system and fractals, along with their encryption outputs and evaluations. Section 'Comparison of permutation techniques' introduces five different methods for the generation of a permutation matrix based on chaotic and non-chaotic procedures. In section 'Mixed permutation–substi tution image encryption algorithms', a complete encryption algorithm with permutation–substitution phases is discussed for all possible combinations with their evaluation criteria and a comparison between 27 encrypted images. Moreover a comparison with eleven recent papers is presented. Finally, section 'Conclusions and recommendations' provides conclusions and future work directions.

Encryption key and evaluation criteria

The encryption key is a representation of specific information that is needed for the successful operation of a cryptosystem. It usually consists of several parameters that are used to initialize and operate the cryptosystem. Modern cryptography concentrates on cryptosystems that are computationally secured against different attacks. One of the most common attacks is the brute-force attack in which all possible combinations of the encryption key are tried. Therefore, an encryption key of length 128 bits or more is considered secure against brute force attacks since it is considered to be computationally infeasible.

Encryption evaluation criteria can be divided into two main categories; the first group includes the statistical tests (pixel correlation coefficients, histogram analysis, entropy values and the NIST statistical test suite) [40,41] and the second group includes the sensitivity tests (differential attack measures, one bit change in the encryption key and the mean square error) [37,42].

Statistical tests

Pixel correlation coefficients

Since the adjacent pixel values of the original image are very close in horizontal, vertical and diagonal directions, the correlation coefficients will be close to 1 in all these directions. The correlation coefficient ρ can be calculated as follow [40]:

$$Cov(x,y) = \frac{1}{n}\sum_{i=1}^{n}\left(x_i - \frac{1}{n}\sum_{j=1}^{n}x_j\right)\left(y_i - \frac{1}{n}\sum_{j=1}^{n}y_j\right), \tag{1a}$$

$$D(x) = \frac{1}{n}\sum_{i=1}^{n}\left(x_i - \frac{1}{n}\sum_{j=1}^{n}x_j\right)^2, \tag{1b}$$

$$\rho = \frac{Cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}, \tag{1c}$$

where n is the number of elements in the two adjacent vectors x and y. For strongly encrypted images, the correlation coefficients approach zero.

Histogram analysis

Histogram analysis shows the distribution of pixel color values across the whole image where curves and peaks for some specific colors appear. For strongly encrypted images this distribution should be flat.

Entropy

The entropy of a specific image measures the randomness of the image-pixels, which enables avoiding any predictability. For a binary source producing 2^8 symbols of equal probabilities (each symbol is 8 bits long), the entropy of this source is given by [37]:

$$Entropy = -\sum_{i=1}^{2^8}P(S_i)\log_2 P(S_i). \tag{2}$$

where the optimal entropy value is 8 for a perfectly encrypted image.

NIST statistical test suite

NIST SP-800-22 statistical test suite is a group of 15 different tests designed to examine the randomness characteristics of a sequence of bits by evaluating the *P*-value distribution (PV) and the proportion of passing sequences (PP) [41]. If a *P*-value for a test is 1, then this means the sequence is considered as a truly random sequence.

Sensitivity tests

Differential attack measures

Strong encryption algorithms should be sensitive to any small change in the input image and produce a totally different output. Quantitatively, different measures are defined for evaluating the protection levels against differential attacks [42]. Let $E1$ and $E2$ be the encrypted images corresponding to the original image without changes and with only one pixel change, respectively.

The Mean Absolute Error (MAE) measures the absolute change between the encrypted image E and the source image P. Let W and H be the width and height of the source image, respectively, then:

$$MAE = \frac{1}{W \times H}\sum_{i=1}^{H}\sum_{j=1}^{W}|P(i,j) - E(i,j)| \tag{3}$$

The Number of Pixels Change Rate (NPCR) measures the percentage of different pixels between $E1$ and $E2$ and it is calculated by the following:

$$D(i,j) = \begin{cases} 0 & E1(i,j) = E2(i,j) \\ 1 & E1(i,j) \neq E2(i,j) \end{cases} \tag{4a}$$

$$NPCR = \frac{1}{W \times H}\sum_{i=1}^{H}\sum_{j=1}^{W}D(i,j) \times 100\% \tag{4b}$$

The Unified Average Changing Intensity (UACI) measures the average intensity of differences between $E1$ and $E2$ and it is calculated by the following:

$$UACI = \frac{1}{W \times H}\sum_{i=1}^{H}\sum_{j=1}^{W}\frac{|E1(i,j) - E2(i,j)|}{255} \times 100\% \tag{5}$$

Sensitivity to one bit change in the encryption key

A good encryption process should also be sensitive to any slight change in any of its parameters and, hence, one bit change in the encryption key should lead to a totally different behavior in the encryption process [37]. This sensitivity is evaluated using the Mean Square Error (MSE) which indicates how far the wrong decrypted image is from the original image. The encryption algorithm becomes better as this value gets larger. MSE is calculated as follows.

$$MSE = \frac{1}{W \times H}\sum_{i=1}^{H}\sum_{j=1}^{W}(P(i,j) - E(i,j))^2 \tag{6}$$

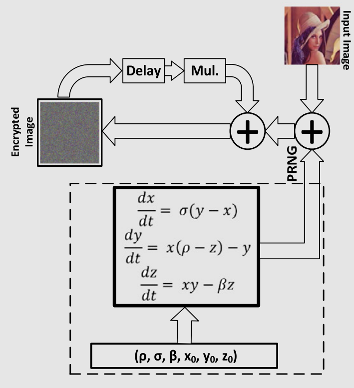
where W and H are the width and height of the image respectively, is the original pixel value at location (i,j) and $E(i,j)$ is the encrypted pixel value at the same location.

The previous evaluation criteria are used to evaluate 27 different simple encryption algorithms by selecting three different substitution techniques as well as three different permutation techniques. The first three encryption algorithms are based only on substitution techniques, and the outputs of another six encryption algorithms are based on three permutation techniques under two different cases when the permutation key is independent of (fixed) or dependent on (dynamic) the input image. Moreover, the outputs of 18 cases, with all possible combinations of mixed permutations (three techniques) and substitutions (three techniques), are investigated under either fixed or dynamic permutation key.

Substitution-only encryption algorithm

The simplest encryption algorithm is described by a delay element, a multiplexer and a PRNG, previously discussed [7,43]. Table 1 shows three different substitution encryption algorithms where the PRNG is based on continuous Lorenz discretization using Euler method [44], a combination of generalized discrete (sine, tent and logistic) maps [43,45] and fractals [7]. It is worthy to note that the multiplexer adds the

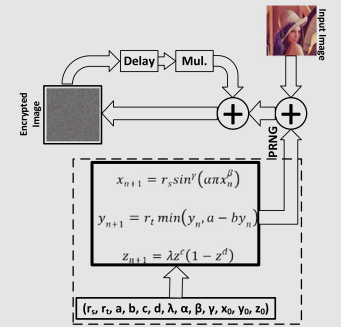
Table 1 Correlation coefficients and differential attack measures for three different substitution only encryption algorithms.

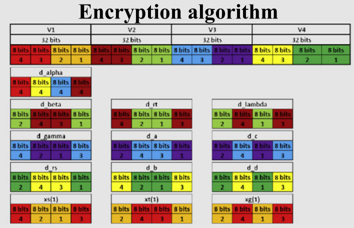
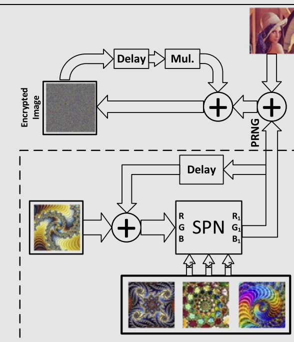
<table>
<tr><th></th><th>Continuous Chaos
(Lorenz Differential Equations)</th><th>Generalized Discrete Chaotic Maps
(Sine, Tent and Logistic maps)</th><th>Fractals</th></tr>
<tr><td>Simple Encryption algorithms</td><td>

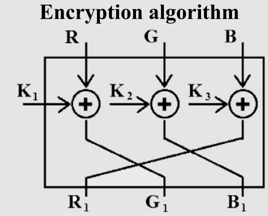
Encryption algorithm

$$\frac{dx}{dt} = f(x, y, z)$$

$$x_{n+1} = x_n + hf(x_n, y_n, z_n)$$

Euler-method formula</td><td>

Encryption algorithm

Encryption key design</td><td>

Encryption algorithm


Substitution-Permutation Network</td></tr>
<tr><td>Average Correlation Coefficients</td><td>Horz. 0.0015 | Vert. 0.0011 | Diag. 0.0021</td><td>Horz. 0.0019 | Vert. 0.0015 | Diag. 0.0020</td><td>Horz. 0.0010 | Vert. 0.0013 | Diag. 0.0032</td></tr>
<tr><td>Diff. Attack Measures</td><td>MAE 77.6603 | NPCR% 49.1994 | UACI% 16.5322</td><td>MAE 77.6072 | NPCR% 49.1986 | UACI% 16.5233</td><td>MAE 77.6220 | NPCR% 49.1997 | UACI% 16.5295</td></tr>
</table>

required nonlinearity and the delay element improves the encryption statistics because each pixel affects all upcoming encrypted pixels.

PRNG based on Lorenz chaotic system

The continuous differential equations of Lorenz system are given by the following:

$$\frac{dx}{dt} = \sigma(y - x), \tag{7a}$$

$$\frac{dy}{dt} = x(\rho - z) - y, \tag{7b}$$

$$\frac{dz}{dt} = xy - \beta z, \tag{7c}$$

where σ, ρ and β are the system parameters and the key consists of these parameters as well as the initial conditions x_0, y_0, and z_0 [46], which guarantee chaotic behavior. There are many hardware realizations for the above system based on current/voltage active blocks or based on transistors [8]. The major problem of such analog circuits is how to control the initial conditions as well as the system parameters precisely. Another methodology to overcome this issue is to discretize this system where the state variables and parameters are represented by registers [47]. The effect of the discretization techniques on the output behavior was discussed [44] where the Euler-formula gives the highest value of Maximum Lyapunov Exponent (MLE). The Euler formula is given in Table 1, where h should be small enough and equal to 2^{h_1} in digital realization to model its multiplication effect as shift left by h_1 bits. Many encryption algorithms were introduced based on the Lorenz chaotic system [39,48].

For the substitution phase using Lorenz attractor, the attractor output is XORed with the current pixel from the scrambled image and the last encrypted pixel after being multiplexed as shown in Table 1. To ensure that the chosen bits of Lorenz are chaotic, it is recommended to choose 8 bits from the least significant part of each output. Then, the output from the Lorenz attractor is mapped to the range from 0 to 255 as follows:

$$x_l = mod(int(abs(x) \times sf), 256), \tag{8a}$$

$$y_l = mod(int(abs(y) \times sf), 256), \tag{8b}$$

$$z_l = mod(int(abs(z) \times sf), 256), \tag{8c}$$

where x, y and z are the outputs from the Lorenz attractor, sf is a scaling factor chosen as 10^{12}, int returns the integer part of a number, abs returns the absolute value of a number and mod returns the remainder. It should be pointed out that the scaling factor sf is chosen such that the selected bits are highly chaotic.

PRNG based on generalized discrete maps

Due to the fact that integer-order continuous chaotic systems can only be achieved with third or higher order differential equations having nonlinear element(s) [46], then discrete chaotic maps are used in most encryption algorithms due to their simple realizations. However, the encryption keys for such algorithms are limited to two or three parameters, which limit the encryption performance. Recently, there have been many efforts to increase the complexity of such maps by generalizing their recurrence relations [43,45] where the generalized sine, tent and logistic maps are introduced, respectively, as follows:

$$x_{n+1} = r_s sin^{\gamma}(\alpha \pi x_n^{\beta}) \tag{9a}$$

$$y_{n+1} = r_t \min(y_n, a - b y_n) \tag{9b}$$

$$z_{n+1} = \lambda z^c (1 - z^d) \tag{9c}$$

It is clear that the number of parameters increases by two or three for each map separately. The effect of these new parameters on the chaotic behavior is discussed in detail by the calculation of the MLE for each parameter individually [43,45]. Due to the huge number of design parameters $\{a, b, c, d, \alpha, \beta, \gamma, r_t, r_s, \lambda\}$ and initial values, $\{x_0, y_0, z_0\}$ a special mixed-parameters key $\{V_1, V_2, V_3, V_4\}$ is designed to enhance the sensitivity of each parameter and initial value of all used maps as shown in Table 1 (refer to [43] for more details).

PRNG based on fractals

A fractal object is self-similar at numerous scales of magnification and can be represented as a mathematical equation that is iterated for a finite number of times. Hence, a fractal image has many variations in details and colors at all scales. The third PRNG is based on the detailed complexity, self-similarity, and fine structure of fractal images as well as the Substitution Permutation Network (SPN) and a delay element [7,49]. The relationships between the inputs and outputs of the SPN of Table 1 are shifted XOR-functions as follows:

$$R_1 = B \oplus K_3, \tag{10a}$$

$$G_1 = R \oplus K_1, \tag{10b}$$

$$B_1 = G \oplus K_2, \tag{10c}$$

where K_1, K_2 and K_3 are three channels selected from the RGB channels of the chosen fractals [49]. The key of this PRNG consists of the available number of fractals, $\{S\}$ and the numbers of the four used fractals $NPCR$ $\{N_{o1}, N_{o2}, N_{o3}, N_{o4}\}$.

To validate the performance of these encryption algorithms, Fig. 1 shows the encrypted images and the correct decrypted images when the Lena 512×512 image is used [50]. It should be mentioned here that the decryption process is the reverse of the encryption process. As shown in Table 1, the encryption quality is measured using standard evaluation criteria, which include pixel correlation coefficients [40] and differential attack measures [42]. The differential attack measures evaluate the sensitivity of the encryption algorithm to one-pixel change in the input plain image. They are calculated by taking the average of running the algorithm for 50 times, where in each time a random pixel from the original image is selected and changed. The average RGB correlation coefficients and differential attack measures are reported in Table 1 for the three algorithms, where the correlation coefficients are very good but the average values of differential attack measures are poor, especially and $UACI$. To discuss the encryption-key sensitivity, the Least-Significant-Bit (LSB) of the parameters x_0, V_4 and N_{o1} is changed in the decryption process for the Lorenz, generalized maps and fractals algorithms, respectively. Fig. 1 shows the wrongly decrypted images, which look random as clear from the values of the MSE and entropy.

Comparison of permutation techniques

The objective of the permutation phase is to randomize the pixels' positions within a specific block. This phase increases the complexity of the encryption algorithm and improves the differential attack measures. This section gives a comparative study of five different permutation matrix generation techniques using discrete chaos, permutation vectors, Arnold's cat map, continuous chaos and chess-based horse move where the permutation phase related to each of the aforementioned techniques is described briefly. Let us divide the input image into blocks where each block is of size $N \times N$. Then, the objective of each technique is to generate a permutation matrix that defines the new position of each pixel instead of its old position. Different permutation matrices are generated for each block and they should be independent.

Permutation based on logistic map

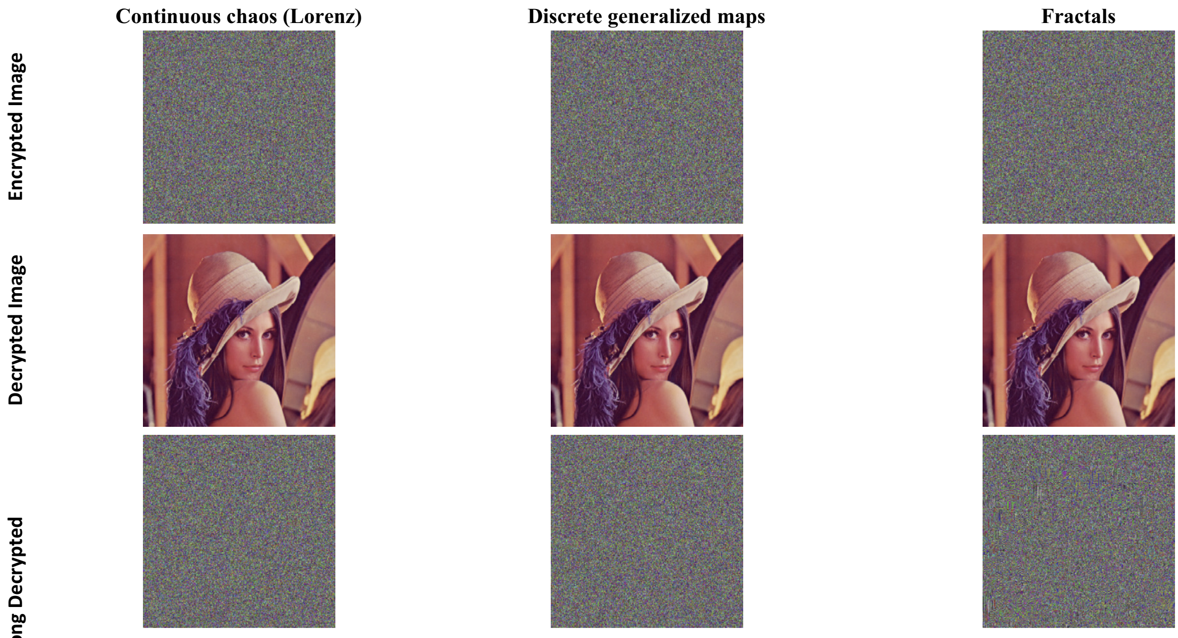
The first technique is based on the conventional logistic map given by the following:

$$x_{n+1} = \lambda x_n (1 - x_n). \tag{11}$$

For each block of size, $N \times N$ the map is calculated for N^2 iterations. Then, the output is sorted in ascending order to constitute the permutation matrix for this block. Only one parameter exists for this logistic map which is λ, but x_0 is the initial value as shown in Table 2. Fig. 2(a) shows a simple example with $N = 3$, which shows the original and modified locations of the pixels. In this case, the permutation matrix is given by, $P_L = \begin{pmatrix} 9 & 1 & 5 \\ 8 & 6 & 3 \\ 4 & 7 & 2 \end{pmatrix}$ which means that the pixel with indices $(1,1)$ will be transferred to location, 9, i.e., indices $(3,3)$. The problem in this permutation technique is that the sorting time increases nonlinearly as the block size increases.

Permutation based on indices vectors

To minimize the sorting time of the previous technique, another permutation technique can be used based on sorting the row and column indices separately as shown in Fig. 2(b). Therefore, to permute a block size $N \times N$ using the logistic map, $2N$ iterations are required from the map (see Table 2), where every N outputs are sorted to represent the new row and column indices such as (312) and (231) in Fig. 2(b). While the sorting time is linear in this technique, the



LSB change	R	G	B
MSE (x_0)	10648.8	9056.16	7097.60
Entropy (x_0)	7.9992	7.9994	7.9993

LSB change	R	G	B
MSE (V_4)	10619.8	9053.74	7077.78
Entropy (V_4)	7.9992	7.9993	7.9993

LSB change	R	G	B
MSE (N_{o1})	10671.6	9080.98	7103.14
Entropy (N_{o1})	7.9994	7.9993	7.9993

Fig. 1 The encrypted images and their correctly and wrongly decrypted images for the three substitution algorithms.

Table 2 Brief description and comparison of the five different permutation techniques.

Name	Logistic Map	Indices Vectors	Arnold's Cat Map	Lorenz System	Chess-Based Horse Move
Type	Discrete Chaos	Discrete Chaos	Discrete Chaos	Continuous chaos	Non-chaotic algorithm
Sorting	Yes	Yes	No	Yes	No
Iterations ($N \times N$ Matrix)	N^2	$2N$	N^2	$N^2/3$	N^2
Parameters	λ	λ	a, b	a, b, c	Algorithm-based
Initial value	x_0 (initial value)	x_0 (initial value)		x_0, y_0, z_0 (initial values)	S_r, S_c (initial position)
Brief Description	Order the n^2 values from $\{1,2,\ldots,n^2\}$	Order the first n values as new row indices $\{1,2,\ldots,n\}$ and the other n for the new column indices.	The new location can be obtained from the previous one without any kind of sorting.	Eliminate the short term predictability by removing the integer part and then order the remaining fractions set $\{X_{1,2,3,\ldots}, Y_{1,2,3,\ldots}, Z_{1,2,3,\ldots}$	Follow the flowchart discussed in [42]
Chosen Parameters	$\lambda = 3.999$	$\lambda = 3.999$	$a = 2, b = 3$	$a = 10, b = 8, c = 8/3$	$S_r = 2, S_c = 3$

permutation efficiency may be poor relative to the previous logistic map technique.

Permutation based on Arnold's cat map

One of the most used permutation algorithms, which does not require sorting, is based on the Arnold's cat map [25–29] where the new location is a function of the old one as follows:

$$\begin{bmatrix} x_{new} \\ y_{new} \end{bmatrix} = \begin{bmatrix} 1 & a \\ b & 1+ab \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} mod(N) + \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \tag{12}$$

Table 2 shows a comparison with the previous techniques and Fig. 2(c) shows an example using this technique.

Permutation based on Lorenz system

The fourth common permutation technique is based on continuous chaotic differential equations such as the Lorenz equations given by (7) [46,8]. In this technique, the three outputs are collected and the first N^2 values are sorted to identify the permutation matrix as shown in Fig. 2(d). One of the major problems in this technique is the time required for solving the differential equations.

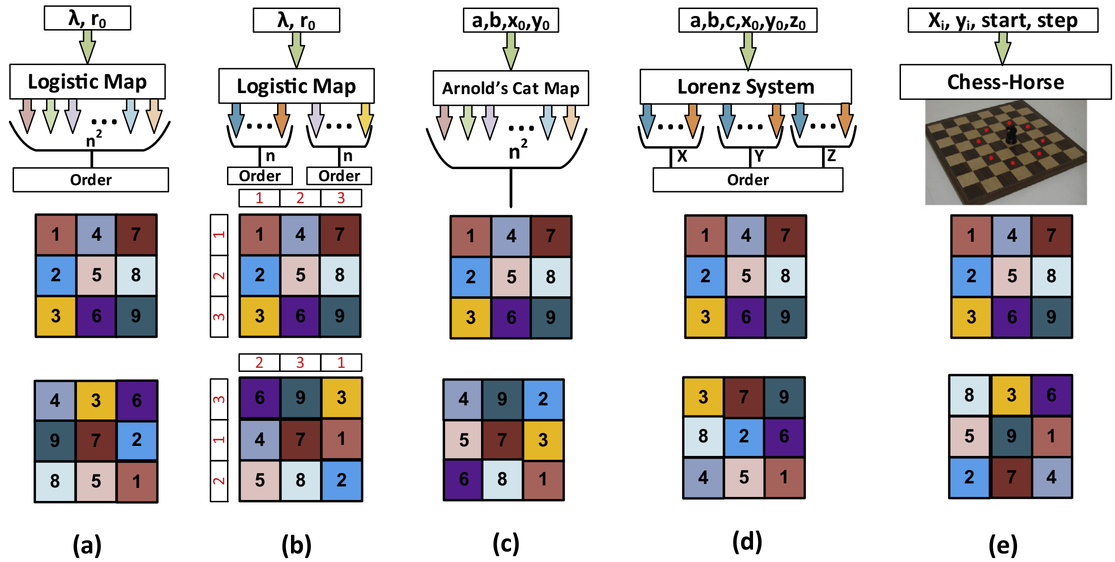
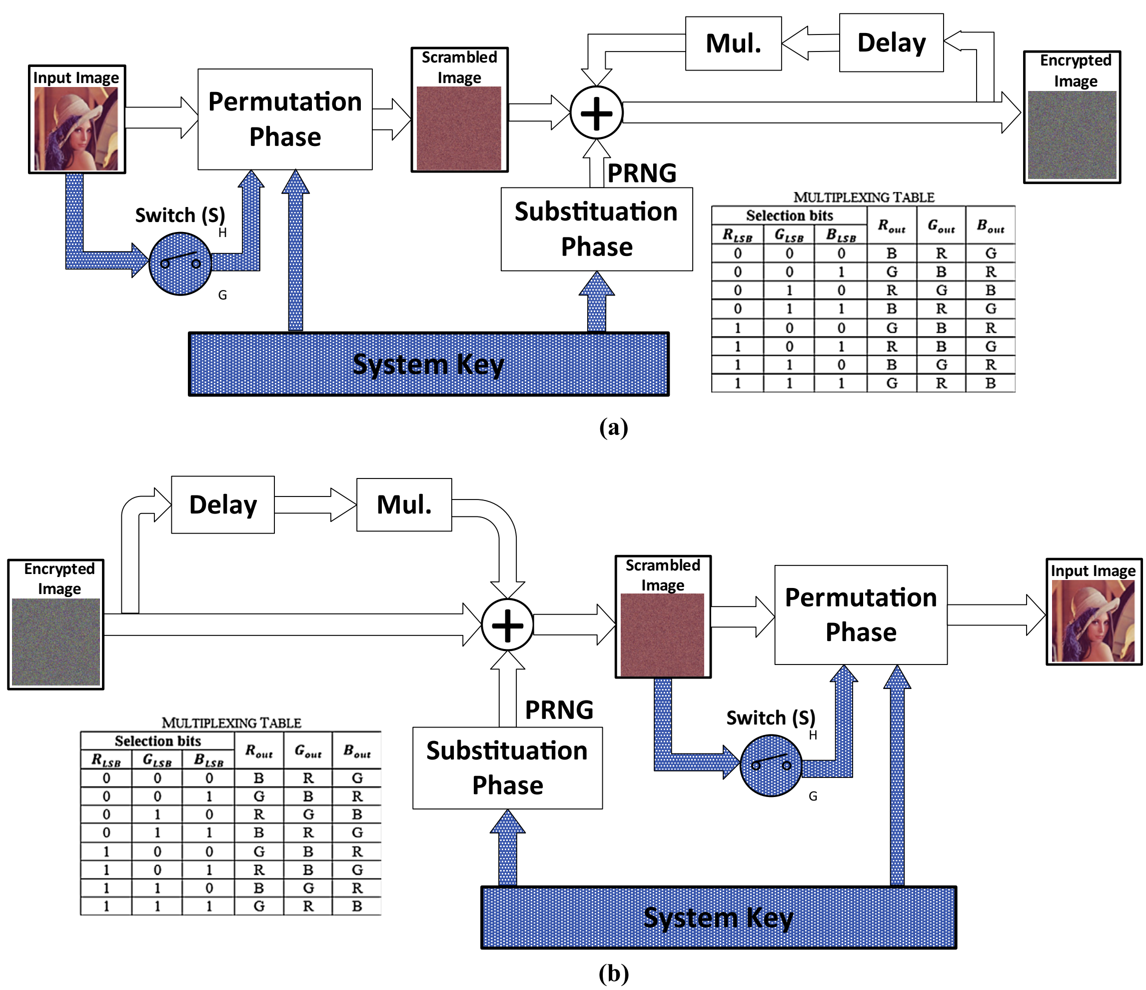


Fig. 2 Illustration of the five different permutation techniques and how they permute a block of size 3×3.



Selection bits			R_{out}	G_{out}	B_{out}
R_{LSB}	G_{LSB}	B_{LSB}			
0	0	0	B	R	G
0	0	1	G	B	R
0	1	0	R	G	B
0	1	1	B	R	G
1	0	0	G	B	R
1	0	1	R	B	G
1	1	0	B	G	R
1	1	1	G	R	B

Fig. 3 (a) Block diagrams of encryption algorithm and (b) block diagrams of decryption algorithm.

Permutation based on chess-algorithm

While all the previous techniques are based on chaotic systems, either discrete or continuous, this permutation technique is based on the chess horse-move. The general block diagram of the proposed encryption algorithm was previously discussed [51], where the next position is generated in a cyclic way based on the horse-move and available locations as shown in Fig. 2(e).

Table 2 and Fig. 2 show a comparison and process evaluation of each technique. Because we chose three different substitution techniques, let us similarly choose three different permutation techniques. The Arnold's cat map, Lorenz system and the chess-based algorithms are chosen as they represent discrete chaotic maps, continuous chaotic maps and non-chaotic systems, respectively.

Mixed permutation–substitution image encryption algorithms

This section investigates the encryption response of 24 different algorithms where Fig. 3(a) shows a complete block diagram for these encryption algorithms based on both permutation and substitution phases. In these algorithms, the permutation phase block represents one of the selected permutation techniques (Lorenz chaotic system, Arnold's cat map and chess-based algorithm) and the substitution phase block represents one of the selected substitution techniques (Lorenz chaotic system, generalized discrete maps and the fractal-based algorithm). Therefore, nine different cases are investigated to cover all possible permutation–substitution combinations. It is to be noted that the output of each permutation phase is stored as a scrambled image as shown in Fig. 3(a), which represents the effect of permutation-only encryption algorithms and, thus, a total of twelve cases are evaluated. Moreover, there is a switch in the encryption block diagram which relates the permutation key to the input image. Hence, these outputs will be repeated when $S = 0$ and $S = 1$, which correspond to static permutation key (independent of the input image) and dynamic permutation key (dependent on the input image).

In this section, the color version of the "Lena" image (512×512) is encrypted. In this symmetric-key cryptosystem, the decryption process is the inverse of the encryption process as shown in Fig. 3(b). To encrypt a source image, the whole image is first scrambled using the chosen permutation algorithm. The permutation parameters are extracted from the encryption key and the switch S controls their dependence on the source image. If the switch S is disconnected ($S = 0$), the parameters are calculated from the key only. If S is connected ($S = 1$), the source image contributes to the calculation of the permutation parameters. When, $S = 1$ the algebraic sum of the input image three color channels is calculated by the following:

$$P_{Sum} = R_{Sum} + G_{Sum} + B_{Sum}, \tag{13}$$

where R_{Sum}, G_{Sum} and B_{Sum} are the sums of the red, green and blue channels of the input image, respectively.

Encryption key design

Fig. 4 shows the structure of the encryption key. It consists of two sets of parameters for each technique: the substitution parameters and the permutation parameters. Since the switch S affects the permutation parameters only, then the new parameters can be calculated from the following equations:

Lorenz permutation parameters

$$x_0 = x_{key} + \frac{mod(P_S, F) + 1}{F}, \tag{14a}$$

$$y_0 = y_{key} + \frac{mod(P_S, F) + 1}{F}, \tag{14b}$$

$$z_0 = z_{key} + \frac{mod(P_S, F) + 1}{F}, \tag{14c}$$

General Encryption Key — Substitution Parameters | Permutation Parameters

Cont. Chaos (Lorenz) — Substitution Parameters

X	Y	Z
32 bits	32 bits	32 bits
5.538	3.627	9.183

Discrete Maps — Substitution Parameters

V1	V2	V3	V4
32 bits	32 bits	32 bits	32 bits
B93E61A2	A2F49CB5	8EA37B51	C49A5E68

Fractals — Substitution Parameters

Fractals Count	Fractal No. 1	Fractal No. 2	Fractal No. 3	Fractal No. 4
8 bits	N bits	N bits	N bits	N bits
16	1	2	3	4

Continuous Chaos — Permutation Parameters

L	x_{key}	y_{key}	z_{key}
4 bits	32 bits	32 bits	32 bits
9	6.294	-6.756	2.886

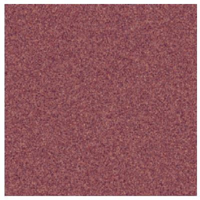
Arnold's Cat Map — Permutation Parameters

L	a_{key}	b_{key}
4 bits	L bits	L bits
9	73	35

Chess-based — Permutation Parameters

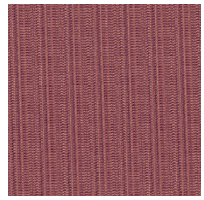
L	$S_{r\text{-}key}$	$S_{c\text{-}key}$
4 bits	L bits	K bits
9	256	256

Fig. 4 Design of the encryption key for each of the chosen substitution and permutation techniques.



	Horz.	Vert.	Diag.
Correlation Coefficients	0.0003	0.0011	0.0018

(a)

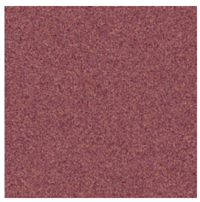


	Horz.	Vert.	Diag.
Correlation Coefficients	0.4607	0.0235	0.0409

(b)

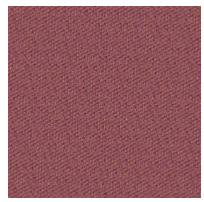
	Horz.	Vert.	Diag.
Correlation Coefficients	0.0875	0.9202	0.0871

(c)



	Horz.	Vert.	Diag.
Correlation Coefficients	0.0024	0.0004	0.0018

(d)



	Horz.	Vert.	Diag.
Correlation Coefficients	0.0928	0.0139	0.0999

(e)

	Horz.	Vert.	Diag.
Correlation Coefficients	0.0641	0.9201	0.0635

(f)

Fig. 5 The scrambled image and its adjacent pixel correlation coefficients where (a–c) and (d–f) are for the continuous chaos, discrete chaos and chess-based algorithm when $S = 0$ and $S = 1$, respectively.

where F is an integer value, which reflects the effective precision of P_S on the initial conditions.

Arnolds' Cat map permutation parameters

$$a = mod(P_S + a_{key}, N - 1) + 1, \tag{15a}$$

$$b = mod(P_S + b_{key}, N - 1) + 1. \tag{15b}$$

Chess-based permutation parameters

$$S_c = mod(P_S + S_{c-_{key}}, N) + 1, \tag{16a}$$

$$S_r = mod(P_S + S_{r-_{key}}, N) + 1, \tag{16b}$$

where the value of P_s depends on the switch S and (13) as follows:

$$P_s = \begin{cases} 0 & S = 0 \\ P_{sum} & S = 1 \end{cases}. \tag{17}$$

For the color version of Lena (512×512); i.e. $N = 512 = 2^9$, $L = 9$, so it requires 4 bits to store L. Then, the total encryption key length can be calculated from both the substitution and permutation key lengths as shown in Fig. 4. It is to be noted that some of the substitution parameters are chosen to enhance the sensitivity to any bit change in that key. For example, although the generalized discrete chaotic maps have 10 parameters and 3 initial values as shown in Table 1, they are merged into only 4 key parameters $\{V_1, V_2, V_3, \text{ and } V_4\}$ as shown in Fig. 4. In the substitution phase, the substitution-key length can be controlled as in the case of fractals-based substitution, $(4N + 8)$ bits, or fixed as in the two other cases (96 and 128 bits for the Lorenz and generalized maps, respectively). Similarly for the permutation phase, the key length can be controlled for the two cases of Arnold's cat map and chess-based algorithm with $(4 + 2L)$ and $(4 + L + K)$ bits, respectively. In the Lorenz-based permutation technique, the key length is fixed and equals 100 bits. For example, let us assume that the Lorenz technique is selected for both substitution and permutation then the key length will be 96 bits for the substitution phase and 100 bits for the permutation phase. This gives a total key length of 196 bits, which is large enough to resist brute-force attacks.

Permutation-only encryption algorithm

The output of the scrambled images of Lena is shown in Fig. 5 for six different cases: three permutations with $S = 0$ and three with $S = 1$. These outputs represent the permutation-only encryption algorithm, where the encrypted images are visually more random in chaotic generators than in the chess-based algorithm. The average correlation coefficients of the three channels are shown in Fig. 5 where the effect of continuous Lorenz is better than that of the discrete chaos. It is clear that $S = 1$ (dynamic permutation key) does not highly affect the continuous permutation because the correlation coefficients are already in the good range. However, it enhances the correlation coefficients of the discrete permutation such that the horizontal correlation coefficients are divided by 5, which decreases the gaps between the correlation coefficients in different directions. Regarding the chess-based algorithm shown in Fig. 5(c) and (f), the encrypted image is visually not good as clear from the average correlation coefficients, especially the vertical measure, which reflects the vertical lines in the encrypted images either with $S = 0$ or $S = 1$. Note that, in the permutation algorithms, the pixels RGB values do not change but the locations of the pixels do change. Therefore, the histograms of all six cases are identical to those of the original image, which makes all these algorithms unsecured. Moreover, the differential attack measures and other evaluation techniques will fail for these outputs, which clarifies the need for permutation–substitution encryption algorithms.

Table 3 Average encryption measures over the three RGB channels as well as mean square error and entropy results for images with resolution 512 × 512.

Substitution Phase		(Case 1:S=0) Permutation Phase: Continuous Chaos (Lorenz System)				Discrete Chaos (Arnold's Cat Map)				Chess-Based Algorithm			
Continuous Chaos (Lorenz)	Correlation Coefficients		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.
		Encrypted Lena	0.0016	0.0020	0.0011	Encrypted Lena	0.0015	0.0020	0.0012	Encrypted Lena	0.0008	0.0008	0.0009
	Diff. Attack Measures		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%
		Encrypted Lena	77.4891	46.5716	15.6504	Encrypted Lena	77.5520	33.5623	11.2651	Encrypted Lena	77.6031	49.1517	16.4985
	Mean Square Error & Entropy	LSB change	R	G	B	LSB change	R	G	B	LSB change	R	G	B
		MSE (x_{0P})	4792.65	5575.98	2314.71	MSE (a_P)	4308.85	5403.68	2272.35	MSE (S_{rP})	4694.11	5437.94	2213.14
		Entropy (x_{0P})	7.2531	7.5940	6.9684	Entropy (a_P)	7.2531	7.5940	6.9684	Entropy (S_{rP})	7.2531	7.5940	6.9684
		MSE (z_{0S})	10648.02	9085.31	7104.02	MSE (z_{0S})	10653.04	9076.12	7110.76	MSE (z_{0S})	10660.36	9067.68	7113.67
		Entropy (z_{0S})	7.9993	7.9992	7.9992	Entropy (z_{0S})	7.9992	7.9995	7.9992	Entropy (z_{0S})	7.9994	7.9993	7.9993
Generalized Discrete Chaos	Correlation Coefficients		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.
		Encrypted Lena	0.0030	0.0008	0.0025	Encrypted Lena	0.0014	0.0013	0.0010	Encrypted Lena	0.0023	0.0008	0.0011
	Diff. Attack Measures		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%
		Encrypted Lena	77.5792	46.5704	15.6496	Encrypted Lena	77.6165	33.5619	11.2759	Encrypted Lena	77.6018	49.1483	16.5172
	Mean Square Error & Entropy	LSB change	R	G	B	LSB change	R	G	B	LSB change	R	G	B
		MSE (x_{0P})	4792.65	5575.98	2314.71	MSE (a_P)	4308.85	5403.68	2272.35	MSE (S_{rP})	4694.11	5437.94	2213.14
		Entropy (x_{0P})	7.2531	7.5940	6.9684	Entropy (a_P)	7.2531	7.5940	6.9684	Entropy (S_{rP})	7.2531	7.5940	6.9684
		MSE (V_{4S})	10655.27	9052.15	7082.43	MSE (V_{4S})	10623.12	9066.00	7090.83	MSE (V_{4S})	10641.98	9094.64	7097.73
		Entropy (V_{4S})	7.9993	7.9992	7.9992	Entropy (V_{4S})	7.9993	7.9994	7.9992	Entropy (V_{4S})	7.9993	7.9993	7.9993
Fractal-Based Algorithm	Correlation Coefficients		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.
		Encrypted Lena	0.0014	0.0033	0.0021	Encrypted Lena	0.0009	0.0012	0.0015	Encrypted Lena	0.0016	0.0008	0.0008
	Diff. Attack Measures		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%
		Encrypted Lena	77.5786	46.5729	15.6327	Encrypted Lena	77.6595	33.5622	11.2749	Encrypted Lena	77.5148	49.1507	16.5128
	Mean Square Error & Entropy	LSB change	R	G	B	LSB change	R	G	B	LSB change	R	G	B
		MSE (x_{0P})	4792.65	5575.98	2314.71	MSE (a_P)	4308.85	5403.68	2272.35	MSE (S_{rP})	4694.11	5437.94	2213.14
		Entropy (x_{0P})	7.2531	7.5940	6.9684	Entropy (a_P)	7.2531	7.5940	6.9684	Entropy (S_{rP})	7.2531	7.5940	6.9684
		MSE (No_{1S})	10661.43	9046.97	7103.65	MSE (No_{1S})	10683.86	9058.40	7083.23	MSE (No_{1S})	10675.71	9044.52	7112.35
		Entropy (No_{1S})	7.9992	7.9994	7.9994	Entropy (No_{1S})	7.9992	7.9993	7.9992	Entropy (No_{1S})	7.9994	7.9994	7.9992

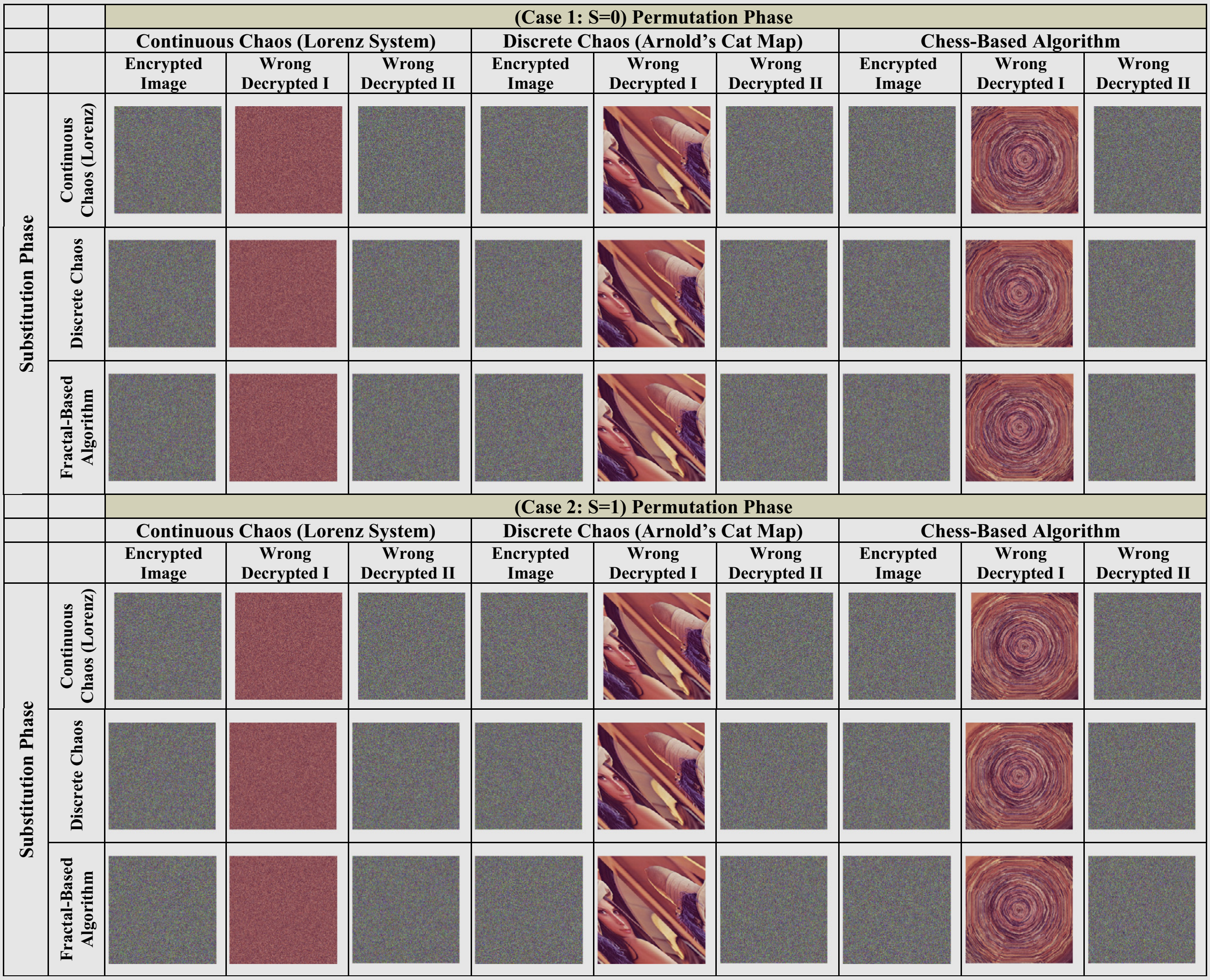
Substitution Phase		(Case 2: S=1) Permutation Phase: Continuous Chaos (Lorenz System)				Discrete Chaos (Arnold's Cat Map)				Chess-Based Algorithm			
Continuous Chaos (Lorenz)	Correlation Coefficients		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.
		Encrypted Lena	0.0007	0.0009	0.0020	Encrypted Lena	0.0015	0.0013	0.0036	Encrypted Lena	0.0010	0.0017	0.0020
	Diff. Attack Measures		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%
		Encrypted Lena	77.7023	99.6085	33.4921	Encrypted Lena	77.5994	99.6080	33.4580	Encrypted Lena	77.4786	99.6089	33.4522
	Mean Square Error & Entropy	LSB change	R	G	B	LSB change	R	G	B	LSB change	R	G	B
		MSE (x_{0P})	4822.16	5601.99	2325.93	MSE (a_P)	4308.85	5403.68	2272.35	MSE (S_{rP})	4483.81	5326.46	2176.29
		Entropy (x_{0P})	7.2531	7.5940	6.9684	Entropy (a_P)	7.2531	7.5940	6.9684	Entropy (S_{rP})	7.2531	7.5940	6.9684
		MSE (z_{0S})	10635.35	9062.48	7105.96	MSE (z_{0S})	10635.35	9062.48	7105.96	MSE (z_{0S})	10635.35	9062.48	7105.96
		Entropy (z_{0S})	7.9993	7.9994	7.9993	Entropy (z_{0S})	7.9993	7.9994	7.9993	Entropy (z_{0S})	7.9992	7.9993	7.9994
Generalized Discrete Chaos	Correlation Coefficients		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.
		Encrypted Lena	0.0022	0.0011	0.0003	Encrypted Lena	0.0013	0.0009	0.0011	Encrypted Lena	0.0009	0.0002	0.0007
	Diff. Attack Measures		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%
		Encrypted Lena	77.6723	99.6093	33.4741	Encrypted Lena	77.5898	99.6084	33.4597	Encrypted Lena	77.6082	99.6090	33.4557
	Mean Square Error & Entropy	LSB change	R	G	B	LSB change	R	G	B	LSB change	R	G	B
		MSE (x_{0P})	4822.16	5601.99	2325.93	MSE (a_P)	4308.85	5403.68	2272.35	MSE (S_{rP})	4483.81	5326.46	2176.29
		Entropy (x_{0P})	7.2531	7.5940	6.9684	Entropy (a_P)	7.2531	7.5940	6.9684	Entropy (S_{rP})	7.2531	7.5940	6.9684
		MSE (V_{4S})	10627.53	9049.20	7096.98	MSE (V_{4S})	10669.98	9083.27	7064.11	MSE (V_{4S})	10671.25	9060.97	7094.07
		Entropy (V_{4S})	7.9993	7.9993	7.9994	Entropy (V_{4S})	7.9993	7.9992	7.9993	Entropy (V_{4S})	7.9993	7.9992	7.9993
Fractal-Based Algorithm	Correlation Coefficients		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.		Horz.	Vert.	Diag.
		Encrypted Lena	0.0014	0.0011	0.0035	Encrypted Lena	0.0014	0.0018	0.0012	Encrypted Lena	0.0013	0.0011	0.0010
	Diff. Attack Measures		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%		MAE	NPCR%	UACI%
		Encrypted Lena	77.5794	99.6086	33.4648	Encrypted Lena	77.5934	99.6064	33.4701	Encrypted Lena	77.6491	99.6104	33.4692
	Mean Square Error & Entropy	LSB change	R	G	B	LSB change	R	G	B	LSB change	R	G	B
		MSE (x_{0P})	4822.16	5601.99	2325.93	MSE (a_P)	4308.85	5403.68	2272.35	MSE (S_{rP})	4483.81	5326.46	2176.29
		Entropy (x_{0P})	7.2531	7.5940	6.9684	Entropy (a_P)	7.2531	7.5940	6.9684	Entropy (S_{rP})	7.2531	7.5940	6.9684
		MSE (No_{1S})	10685.09	9054.69	7107.48	MSE (No_{1S})	10662.01	9045.09	7077.56	MSE (No_{1S})	10634.85	9072.98	7107.36
		Entropy (No_{1S})	7.9993	7.9994	7.9992	Entropy (No_{1S})	7.9993	7.9993	7.9992	Entropy (No_{1S})	7.9993	7.9993	7.9993

Permutation–substitution encryption algorithms

Two sets of results have been tested based on the switch S, where 9 cases are discussed in each scenario showing all possible combinations of the selected substitution and permutation techniques. When $S = 1$ the input image channels are processed using (13) to calculate P_{Sum}, then, the permutation parameters obtained from the encryption key are further modified using P_{Sum} as in (14)–(17).

Table 3 shows the average correlation coefficients of the RGB channels and the differential attack measures for 18

Table 4 Encrypted and wrong decrypted images.



different encrypted outputs (9 cases for both $S = 0$ and $S = 1$. Moreover, the MSE and entropy are also added in Table 3 for the 18 encryption algorithms under two different wrong decryption processes when the LSB of the substitution and permutation keys is changed.

It is worth noting that the average correlation coefficients for all algorithms are in the order of 10^{-3}, which reflects that the pixels are almost uncorrelated in all directions. Table 4 shows the 18 encrypted images and Fig. 6 illustrates the horizontal correlation distributions in the RGB channels for the original Lena image and four different encrypted outputs. The first observation from this figure is that the influences of all permutation-only algorithms are limited and their effect exists in similar regions related to the original distribution and they do not cover the whole domain. However, the horizontal distribution of the correlations in the RGB channels becomes similar in the 18 mixed permuta tion–substitution algorithms as shown in the last column, where uniform distributions are obtained in all channels. The minimum correlation values from these 18 outputs are in the order of 10^{-4} when using the chess-algorithm for permutation, generalized discrete maps for substitution and $S = 1$.

The differential attack measures are among the main requirements for secure encryption. From the previous studies and Table 3, the effect of different substitution techniques for one permutation technique is minor and can be neglected in both $S = 0$ and $S = 1$. Nevertheless, the main objective of the switch S is to improve the differential attack measures and, especially, the NPCR and UACI measures as shown in Table 3. The NPCR measures jump from 46%, 33%, 49% at $S = 0$ to 99.6%, 99.6%, 99.6% at $S = 1$ corresponding to Lorenz, Arnold and chess-algorithm permutation techniques, respectively. Similarly, the UACI measures jump from 15%, 11%, 16% at $S = 0$ to 33.4%, 33.4%, 33.4% at $S = 1$ corresponding to Lorenz, Arnold and chess-algorithm permutation

Table 5 Sample NIST results for encrypted Lena (1024 × 1024).

Test	Permuted (Arnold)		Permuted (Lorenz)		Permuted (Chess)		Lorenz + Fractals + S=0		Lorenz + Fractals + S=1		Chess-based + Fractals + S=0		Chess-based + Fractals + S=1	
	PV	PP	PV	PP	PV	PP	PV	PP	PV	PP	PV	PP	PV	PP
Frequency	×	0.000	×	0.000	×	0.042	✓	1.000	✓	1.000	✓	1.000	✓	1.000
Block Frequency	×	1.000	×	1.000	×	0.000	✓	1.000	✓	1.000	✓	1.000	✓	1.000
Cumulative Sums	×	0.000	×	0.000	×	0.000	✓	1.000	✓	1.000	✓	1.000	✓	1.000
Runs	×	0.000	×	0.000	×	0.000	✓	0.958	✓	1.000	✓	0.958	✓	1.000
Longest Run	×	0.000	×	0.000	×	0.000	✓	0.958	✓	0.958	✓	1.000	✓	1.000
Rank	✓	1.000	✓	1.000	×	0.000	✓	0.958	✓	1.000	✓	1.000	✓	0.875
FFT	×	0.000	×	0.000	×	0.000	✓	1.000	✓	1.000	✓	1.000	✓	1.000
Non Overlapping Template	✓	0.280	✓	0.313	×	0.010	✓	0.991	✓	0.991	✓	0.992	✓	0.991
Overlapping Template	×	0.000	×	0.000	×	0.000	✓	0.958	✓	1.000	✓	1.000	✓	1.000
Universal	×	0.000	×	0.000	×	0.000	✓	1.000	✓	1.000	✓	0.917	✓	1.000
Approximate Entropy	×	0.000	×	0.000	×	0.000	✓	1.000	✓	0.958	✓	1.000	✓	0.958
Random Excursions		N/A		N/A		N/A	✓	0.983	✓	1.000	✓	1.000	✓	0.993
Random Excursions Variant		N/A		N/A		N/A	✓	0.981	✓	1.000	✓	1.000	✓	1.000
Serial	×	0.000	×	0.000	×	0.000	✓	1.000	✓	1.000	✓	1.000	✓	1.000
Linear Complexity	✓	1.000	✓	0.917	✓	1.000	✓	1.000	✓	1.000	✓	1.000	✓	1.000
Final Result	**Failure**		**Failure**		**Failure**		**Success**		**Success**		**Success**		**Success**	

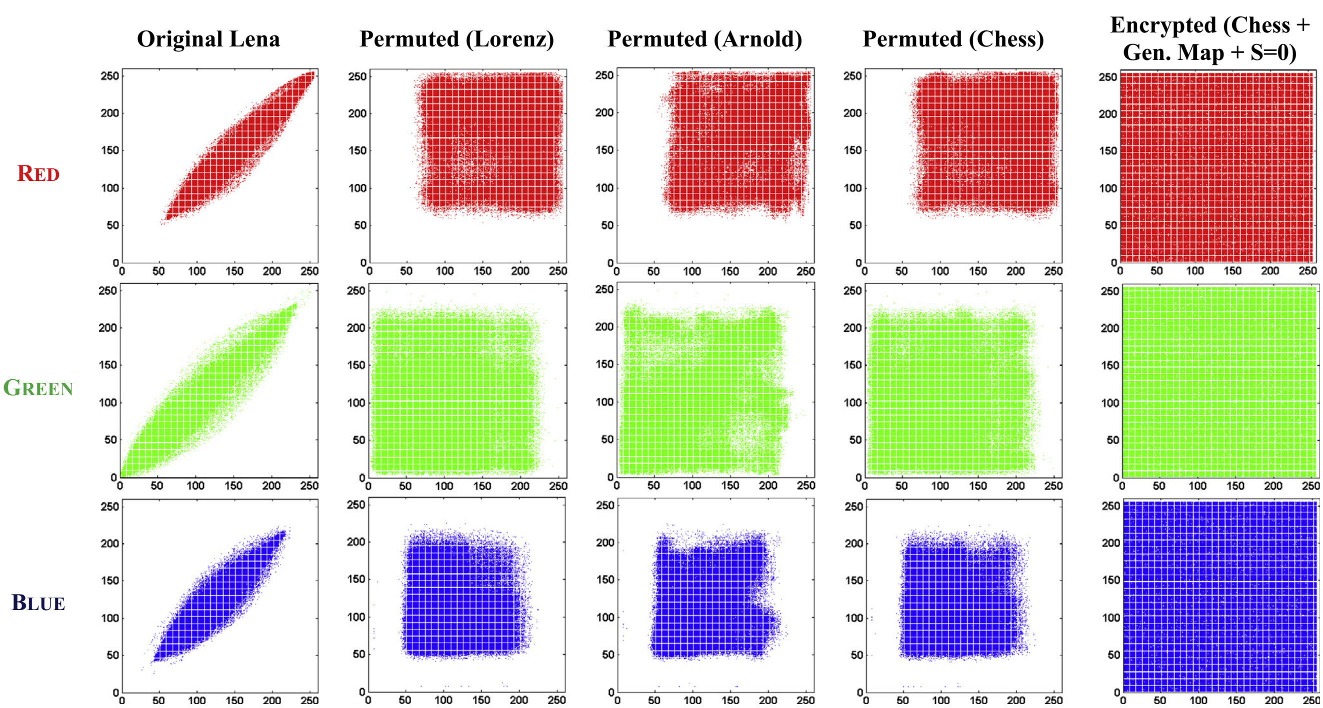


Fig. 6 The horizontal pixel correlation distribution for the RGB channels.

techniques, respectively. These NPCR and UACI values are in the good ranges as reported before [42].

The sensitivity analyses for two different cases are shown in Table 4 for each encryption algorithm and their RMS and entropy values are given in Table 3. The first case is when wrong decryption is applied after changing a single LSB of one parameter from the permutation key with a subscript *P*. The second case is when the LSB is chosen from the substitution key with a subscript *S*. Based on the results of Table 3 for all encryption algorithms, the wrong decryption permutation-key gives the best performance using the Lorenz permutation algorithm. In the chess-based algorithm, the cyclic rotation effect of the horse-move is illustrated in Table 4. The main disadvantage of using Arnold's cat map is that the wrong decrypted images are very bad as all the details of the original image exist as shown in Table 4. However, the second wrong decryption case for all 18 algorithms illustrates a great response as evident from the higher values of the RMS and the entropy, which are very close to 8. Therefore, the key design should focus on the substitution case to improve the sensitivity analysis and the Arnold's cat map is not recommended for secure encryption.

Table 6 Comparison between this review article and eleven recent books and papers. (See below-mentioned reference for further information.)

Ref	Sub	Per.	Used PRNG		Basic Idea	Input Data	Types of Analyses							Extra Information
			Chaotic	Non-chaotic			CC	DA	Sen.	Ent.	MSE	NIST	Time	
This Review	OK	OK	Generalized discrete maps, continuous chaos, and Arnold map	❖ Fractals ❖ Chess based	This paper includes:- ❖ Three substitution only algorithms. ❖ Six permutation only algorithms. ❖ 18 substitution-permutation algorithms	Color image	OK	OK	OK	OK	OK	OK	OK	❖ Two categories are introduced based on the dependency of the encryption key on input image through a switch.
[2]	OK	OK	Conventional Discrete maps (1D, and higher dimensional)		❖ The PRNGs were based on conventional chaotic maps, Chebyshev polynomials, and mod operations with analyses. Others based on higher dimensional chaotic maps, and discretizing chaotic systems based DE with post processing & neural network.	Color images, and video	OK	OK	OK	OK	OK	OK	OK	❖ FPGA Hardware design using Lorenz's chaotic system (for greyscale image) is provided
[10]	OK		Conventional Logistic map		❖ Each character of the message was encrypted by using the logistic map many times.	Text message								❖ Frequency distribution analysis is provided
[11]	OK		Conventional Logistic map		❖ The image encryption algorithm was based on many logistic maps in cascaded loops with constraints	455×569 Color image			OK				169 Sec	❖ The used image is not of standard size
[12]	OK		Conventional Logistic map		❖ The image encryption algorithm was based on Logistic map (many), mixing function with repeated cycles.	559×348 Greyscale							41mSec	❖ The used image is not of standard size.
[13]	OK		Conventional tent map		❖ The image encryption algorithm was based on using two tent maps with bit-plane decomposition.	512×512 Greyscale and Color	OK	OK	OK	OK	OK		0.6 sec for 256×256 Greyscale	❖ Mean, median, standard deviation ❖ UACI in the range of 20%
[14]	OK		2D coupled map lattice		❖ The image encryption was based on flowchart with different maps, and If conditions.	256×256 Greyscale	OK	OK						❖ NPCR = 0.40%, UCAI = 0.3192%
[15]	OK	OK	Conventional logistic map		❖ The image encryption algorithm was based on conventional logistic map, and artificial neural network (ANN) approach	256×256 512×512 Greyscale	OK	OK	OK	OK				❖ The process has many feedbacks.
[16]	OK	OK	Conventional Logistic map	Gray code	❖ The image encryption algorithm was based on P-box and S-box where Gray code and logistic maps are used, respectively.	256×256 Greyscale							14 mSec	❖ The Gray code uses matrices with mod operations.
[17]	OK	OK	2D maps, and the Chinese remainder theorem		❖ The image encryption algorithm is based on two sorting processes, permutation phase (based on 2D hyper-chaos discrete nonlinear dynamic system), and the diffusion (based on Chinese reminder theorem).	512×512 Greyscale	OK		OK	OK		OK	50 mSec	❖ Compression performance was discussed
[18]	OK	OK	Conventional discrete maps (tent, logistic, sine) with the mod operation		❖ Introduced a new map based on a combination of two conventional discrete maps. ❖ In the image encryption algorithm, they inserted random pixels, then row separation, four 1D-map substitutions, row combination, and rotation.	256×256 512×512 1024×1024 Color	OK		OK	OK			0.2 Sec 0.67 Sec 3.15 Sec	❖ Analysis and MLE of the new maps are provided. ❖ Data loss and noise attacks discussions were added.
[52]	OK	OK	Conventional Logistic, tent, and 2D maps		❖ The image encryption algorithm was based on different conventional maps. ❖ No examples are provided.									❖ Theoretical discussion without examples.

Sub. : Substitution; Per. : Permutation between cells; PRNG : Pseudo Random Number Generator; CC : Correlation Coefficients between pixels; DA : Differential Attack measures; Sen. : Sensitivity Analysis; Ent. : Entropy; MSE. : Mean Square Error; NIST : National Institute of Standards and Technology tests

Table 5 shows the results of the 15 NIST tests [41] performed on Lena 1024×1024 where seven cases are discussed: three permuted images and four fractal-based substitution cases having Lorenz and chess permutation techniques with $S = 0$ and $S = 1$. It is clear from these results that the permutation only techniques are not enough to pass all tests but the mixed techniques succeed in all tests based on chaotic/non-chaotic systems such as in the Lorenz/fractals case or even non-chaotic/non-chaotic algorithms as in the chess/fractals results. Those results further assert the randomness of the encrypted images.

Because it is difficult to simultaneously achieve the best encryption execution time and high security, the objective of this review article is not to provide the best execution time but to provide good encryption quality with nonconventional algorithms. The encryption time for the studied cases can be estimated from the times of the substitution and permutation phases. Using a computer with 2.2 GHz processor, 4G RAM, and for the 256×256 Lena color image, the substitution-only times are 1.149, 3.78 and 0.782 s for the Lorenz, generalized maps and fractals, respectively. Although substitution based on generalized discrete maps has the largest execution time, its complexity and security are high due to the number of parameters and calculations of the generalized maps. Regarding the permutation phase times, they are 0.017, 0.005 and 8.85 s for the Lorenz, Arnold and chess based algorithms, respectively.

The comparison results of the recent publications drawn from 11 sources are presented in Table 6 with respect to the used PRNG's (chaotic and non-chaotic), basic idea of the encryption algorithm, the input data, the applied encryption analyses and some additional details. It is clear that all these papers are based on chaotic generators in the substitution phase and some of them focus only on substitution encryption algorithms [10–14]. The permutation phase of the other papers is related to the conventional discrete chaotic maps except for Zanin and Pisarchik [16], which is based on the Gray code (linear matrices) but without any analysis. Some analyses were not reported and some results are not in the good ranges such as UACI [13], which is 20%, and the NPCR [11]. Some papers reported the execution time for grayscale images and three papers [11,13,18] for color-images. In addition, some analyses such as the NIST statistical tests are not performed. Additional features, which are not covered in this review article, have been introduced in some of these references such as the FPGA hardware design and post-processing [2], data loss and noise attacks [18], and the compression performance [17].

Conclusions and recommendations

This paper covered both substitution and permutation phases, where different techniques were discussed such as discrete chaotic maps (the conventional Arnold's cat map and a combination of three generalized maps), a continuous chaotic system (Lorenz) and non-chaotic algorithms (fractals-based and chess-based horse movement). Complete analyses of 27 different encryption algorithms were summarized in which substitution-only, permutation-only and permutation–substitu tion phases are discussed with and without dependency on the input image. Therefore, several complete encryption algorithms were provided and compared using miscellaneous analyses, which include the NIST statistical tests, key-sensitivity tests and execution times. A comparison with eleven recent publications is provided in Table 6, which illustrates the advantages and wide scope of this review article.

Based on the presented analyses and comparisons, the following recommendations, on how to design a secure image encryption algorithm, can be given. Even though some of these recommendations can be considered as common rules in modern symmetric encryption algorithms, they have not been widely followed. Finally, some future research directions are also provided.

- Permutation-only image encryption schemes are generally insecure: A permutation-only encryption algorithm reallocates the pixels so that the correlation coefficients may be improved but the encrypted image still has the same histogram. Such histograms can reveal some useful information about the plain images. For example, images of human faces usually have narrower histograms than images of natural scenes. In addition to revealing such information, permutation-only encryption schemes usually fail in key sensitivity analysis and NIST results and have poor differential attack measures.
- Substitution-only image encryption schemes are generally more secure than permutation-only schemes: Whether the substitution algorithm is based on discrete chaotic, continuous chaotic or non-chaotic (e.g., fractals) generators, it improves the correlation coefficients, flattens the histograms and can pass the key sensitivity and NIST tests. However, the differential attack results are not good enough since there are no changes in the pixels' positions.
- Permutation–substitution encryption algorithms generally have the best security: A substitution phase can make the cipher-image look random and pass many evaluation criteria. A permutation phase can improve the differential attack measures and is useful in increasing the computational complexity of a potential attack and in making the cryptanalysis of the encryption scheme more complicated or impractical. Hence, permutation–substitution encryption algorithms usually improve all the encryption evaluation criteria and will, most probably, pass the NIST tests.
- Cipher-image feedback with multiplexing is very useful for enhancing the security: The multiplexer adds nonlinearity and the delay element improves the encryption statistics because each pixel affects all upcoming encrypted pixels.
- Permutation phases which are dependent on the input image enhance the security: When the permutation parameters are dynamic, the permutation–substitution encryption algorithm becomes sensitive to any small change in the input image, produce a totally different output and, hence, the differential attack measures are improved.
- Key sensitivity results may not be satisfactory for some permutation techniques: A one bit change in the encryption-key should lead to a totally different behavior in the encryption process. The substitution parameters are usually sensitive to such small changes. However, care should be taken when including the permutation parameters in the encryption-key design.
- Combining chaotic and non-chaotic generators can yield a fast and secure encryption algorithm: For the studied algorithms, performing substitutions using fractals and permutations using a chaotic generator represents a good encryption choice. In addition to security, which was the main objective of this review article, focusing on the speed of the encryption algorithm should be the target of future research so that video encryption can be performed.
- Additional features can enhance the utilization of an image encryption algorithm: For instance, image compression can be performed along with image encryption. Implementing an FPGA hardware design that corresponds to the software design is also needed.

Conflict of Interest

The authors have declared no conflict of interest.

Compliance with Ethics Requirements

This article does not contain any studies with human or animal subjects.

Acknowledgment

This research was supported financially by the Science and Technology Development Fund (STDF), Egypt, Grant No. 4276.

References

[1] Alvarez G, Li S. Some basic cryptographic requirements for chaos-based cryptosystems. Int J Bifurcat Chaos (IJBC) 2006;16(8):2129–51.

[2] Kocarev L, Lian S. Chaos-based cryptography theory, algorithms and applications. Springer; 2011.

[3] Barakat ML, Mansingka AS, Radwan AG, Salama KN. Generalized hardware post processing technique for chaos-based pseudo random number generators. ETRI J 2013;35(3):448–58.

[4] Barakat ML, Mansingka AS, Radwan AG, Salama KN. Hardware stream cipher with controllable chaos generator for colour image encryption. IET Image Process 2014;8(1):33–43.

[5] Abd-El-Hafiz SK, Radwan AG, AbdElHaleem SH, Barakat ML. A fractal-based image encryption system. IET Image Process 2014;8(12):742–52.

[6] Moaddy K, Radwan AG, Salama KN, Momani S, Hashim I. The fractional-order modeling and synchronization of electrically coupled neurons system. Comput Math Appl 2012;64:3329–39.

[7] Radwan AG, Moaddy K, Salama KN, Momani S, Hashim I. Control and switching synchronization of fractional order chaotic systems using active control technique. J Adv Res 2014;5(1):125–32.

[8] Radwan AG, Soliman AM, EL-sedeek AL. MOS realization of the modified Lorenz chaotic system. Chaos Soliton Fract 2004;21:553–61.
[9] Radwan AG, Soliman AM, Elwakil AS. 1-D digitally-controlled multi-scroll chaos generator. Int J Bifurcat Chaos (IJBC) 2007;17(1):227–42.
[10] Baptista MS. Cryptography with chaos. Phys Lett 1998;A240:50–4.
[11] Pisarchik AN, Flores-Carmona NJ, Carpio-Valadez M. Encryption and decryption of images with chaotic map lattices. Chaos 2006;16:033118.
[12] Pisarchik AN, Zanin M. Image encryption with chaotically coupled chaotic maps. Physica D 2008;237:2638–48.
[13] Soma S, Sen S. A non-adaptive partial encryption of grayscale images based on chaos. Proc Technol 2013;10:663–71.
[14] Sun F, Liu S, Li Z, Lu Z. A novel image encryption scheme based on spatial chaos map. Chaos, Solitons Fract 2008;38: 631–40.
[15] Telem ANK, Segning CM, Kenne G, Fotsin HB. A simple and robust gray image encryption scheme using chaotic logistic map and artificial neural network. Adv Multim 2014:602921.
[16] Zanin M, Pisarchik AN. Gray code permutation algorithm for high-dimensional data encryption. Inf Sci 2014;270:288–97.
[17] Zhu H, Zhao C, Zhang X. A novel image encryption–compression scheme using hyper-chaos and Chinese remainder theorem. Signal Process: Image Commun 2013;28(6):670–80.
[18] Zhou Y, Bao L, Chen CLP. A new 1D chaotic system for image encryption. Signal Process 2014;97:172–82.
[19] Zhang W, Wong K, Yu H, Zhu Z. An image encryption scheme using reverse 2-dimensional chaotic map and dependent diffusion. Commun Nonlin Sci Numer Simul 2013;18(8): 2066–80.
[20] Koduru SC, Chandrasekaran V. Integrated confusion-diffusion mechanisms for chaos based image encryption. In: IEEE international conference on computer and information technology (ICIT); 2008. p. 260–3.
[21] Kanso A, Ghebleh M. A novel image encryption algorithm based on a 3D chaotic map. Commun Nonlin Sci Numer Simul 2012;17(7):2943–59.
[22] Zhang W, Wong K, Yu H, Zhu Z. A symmetric color image encryption algorithm using the intrinsic features of bit distributions. Commun Nonlin Sci Numer Simul 2013;18(3): 584–600.
[23] Sethi N, Vijay S. Comparative image encryption method analysis using new transformed – mapped technique. In: Conference on advances in communication and control systems (CAC2S); 2013. p. 46–50.
[24] Sathishkumar GA, Bagan KB, Sriraam N. Image encryption based on diffusion and multiple chaotic maps. Int J Netw Sec Its App (IJNSA) 2011;3(2):181–94.
[25] Ye R. A novel chaos-based image encryption scheme with an efficient permutation-diffusion mechanism. Opt Commun 2011;284:5290–8.
[26] Run-he Q, Yun C, Yu-Zhen F. Integrated confusion-diffusion mechanisms for chaos based image encryption. In: 4th International congress on image & signal processing; 2011. p. 629–32.
[27] Zhang W, Wong K, Yu H, Zhu Z. An image encryption scheme using lightweight bit-level confusion and cascade cross circular diffusion. Opt Commun 2012;285:2343–54.
[28] Zhang X, Shao L, Zhao Z, Liang Z. An image encryption scheme based on constructing large permutation with chaotic sequence. Comput Electr Eng 2014;40:931–41.
[29] Sathishkumar GA, Bagan KB, Vivekan V. A novel algorithm for image encryption by integrated pixel scrambling plus diffusion [IISPD] utilizing duo chaos mapping applicability in wireless systems. Proc Comp Sci 2011(3):378–87.
[30] Wu Y, Zhou Y, Noonan JP, Agaian S. Design of image cipher using Latin squares. Inf Sci 2014;264:317–39.
[31] Pareek NK, Patidar V, Sud KK. Diffusion–substitution based gray image encryption scheme. Digit Signal Process 2013;23(3): 894–901.
[32] Al-Husainy MAF. A novel encryption method for image security. Int J Sec Its Appl 2012;6(1):1–8.
[33] Pareek NK, Patidar V, Sud KK. Substitution-diffusion based image cipher. Int J Net Sec Its Appl (IJNSA) 2011;3(2):149–60.
[34] Li Z, Liu X. The image encryption algorithm based on the novel diffusion transformation. In: International conference on control and electronic engineering (CMCE); 2010. p. 345–8.
[35] Wang X, Yang L. A novel chaotic image encryption algorithm based on water wave motion and water drop diffusion models. Opt Commun 2012;285(20):4033–42.
[36] Fouda JSAE, Effa JY, Sabat SL, Ali M. A fast chaotic block cipher for image encryption. Commun Nonlin Sci Numer Simul 2014;19(3):578–88.
[37] Zhang A, Zhou N. Color image encryption algorithm combining compressive sensing with Arnold transform. J Comp 2013;8(11): 2857–63.
[38] Zhang Y, Xiao D. Self-adaptive permutation and combined global diffusion for chaotic color image encryption. Int J Electron Commun (AEÜ) 2014(68):361–8.
[39] AbdElHaleem SH, Radwan AG, Abd-El-Hafiz SK. A chess-based chaotic block cipher. In: IEEE international new circuits and system conference (NEWCAS); 2014. p. 405–8.
[40] Corrochano EB, Mao Y, Chen G. Chaos-based image encryption: handbook of geometric computing. Berlin-Heidelberg: Springer; 2005.
[41] Rukhin A et al. A statistical test suite for random and pseudorandom number generators for cryptographic applications. Publication: NIST Special; 2001, p. 800–22.
[42] Wu Y, Noonan JP, Agaian S. NPCR and UACI randomness tests for image encryption. J Select Areas Telecomm (JSAT) 2011:31–8.
[43] Abd-El-Hafiz SK, Radwan AG, AbdElHaleem SH. Encryption applications of a generalized chaotic map. Appl Math Inf Sci (AMIS) 2015;9(6):1–19.
[44] Zidan MA, Radwan AG, Salama KN. The effect of numerical techniques on differential equation based chaotic generators. In: The 23rd international conference on microelectronics (ICM); 2011. p. 1–4.
[45] Radwan AG. On some generalized logistic maps with arbitrary power. J Adv Res 2013;4:163–71.
[46] Gonzales OA, Han G, De Gyvez J, Sanchez-Sinencio E. Lorenz-based chaotic cryptosystem: a monolithic implementation. IEEE Trans Circ Syst I 2000;47:1243–7.
[47] Zidan MA, Radwan AG, Salama KN. Controllable v-shape multi-scroll butterfly attractor: system and circuit implementation. Int J Bifurcat Chaos (IJBC) 2012;22(6): 1250143.
[48] Radwan AG, Abd-El-Hafiz SK, Abd-El-Haleem SH. Image encryption in the fractional-order domain. In: International conference on engineering and technology (ICET); 2012. p. 1–6.
[49] AbdElHaleem SH, Radwan AG, Abd-El-Hafiz SK. Design of pseudo random keystream generator using fractals. In: IEEE international conference on electrical circuits & systems (ICECS); 2013. p. 877–80.
[50] USC-SIPI Image Database, University of Southern California, Signal and Image Processing Institute., November 2013, http://sipi.usc.edu/database/.
[51] AbdElHaleem SH, Radwan AG, Abd-El-Hafiz SK. A chess-based chaotic block cipher. In: IEEE international new circuits and system conference (NEWCAS); 2014. p. 405–8.
[52] Pisarchik AN, Zanin M. Chaotic map cryptography and security. Horizons in computer science, vol. 4. Springer; 2012.