Egyptian Informatics Journal 24 (2023) 100376

Contents lists available at ScienceDirect

Egyptian Informatics Journal

journal homepage: www.sciencedirect.com

Full length article

Diagonalize three-dimensional nonlinear chaotic map to encrypt color image

Mahmoud I. Moussa [a,*], Eman I. Abd El-Latif [b], Ahmed H. Abu El-Atta [a]

[a] Computer Science Department, Faculty of Computer and Artificial Intelligence, Benha University, Egypt
[b] Department of Mathematics and Computer Science, Faculty of Science, Benha University, Egypt

ARTICLE INFO

Article history:
Received 17 October 2021
Revised 26 March 2022
Accepted 1 May 2023
Available online 7 June 2023

Keywords:
Chaotic system
Cryptography
Image encryption
Trilinear system

ABSTRACT

In this paper, multidimensional chaos systems called a trilinear chaotic system for encryption the color image is used. Combining the multi-dimensional chaotic system with encryption algorithms improves the security. The trilinear system is built from logistic and cubic maps and generates six completely random bijections $T_1(x), T_2(x), T_2(y), T_3(x), T_3(y)$, and $T_3(z)$. The map $T_1(x)$ randomly shuffles the image pixels, the two maps $T_2(x)$ and $T_2(y)$ scramble the position of the pixels, and the other three maps change the pixel values. The correlations among the three components of RGB components are reduced, and the security of the algorithm increases. The simulation results demonstrated that the procedure has very large key spaces and a high level of security.

1. Introduction

A variety picture consists of many lines and segments of pixels; these pixels have specific values that address the level of the variety in the picture. As a result, the color image is represented as a three numerical matrix for each small area to define the color. Image security is a crucial issue because sensitive images can invite attacks from different places. The transformation of a plain image into a different cipher image is the goal of image encryption. Due to its sensitivity to initial parameters, high security and totally random performance, chaos theory is used in image encryption. Matthews created the first chaotic system in 1989 [1]. Numerous scientific algorithms have defined image encryption with a few secret parameters based on one- or two-dimensional chaotic systems. This work conquers the past works by offering a totally irregular multi-layered framework with different mystery boundaries. The presented chaotic system simultaneously encrypts RGB image and their components. The proposed chaos system consists of six maps $T_1(x)$, $T_2(x)$, $T_2(y)$, $T_3(x)$, , $T_3(y)$, and $T_3(z)$ to increase conjugation of the items $x_i^3, x_i^2, y_i^2, x_iy_i, y_i^3, z_i^2x_i, \cdots z_i^3$ and expand the security of the system. Initially, the proposed Pixel Transform Table (PTT) procedure inputs the first three maps $T_1(x)$, $T_2(x)$, and $T_2(y)$ to create three $M \times N$ matrix keys (M_r, M_b, M_g), and it inputs the other three maps $T_3(x)$, , $T_3(y)$, and $T_3(z)$ to output three vector keys (V_r, V_b, V_g) of length MN. Utilizing vector keys, the rows and columns of images are shuffled and scrambled. While the remainder keys are applied many times to growth the difficulty and the security of the system. The first matrix M_r is performed to shuffle the block of the image and the remainder matrix M_b, M_g scramble the location in rows and columns, respectively. Finally, in order to increase the system's complexity and security, the three vector keys (V_r, V_b, V_g) are utilized to twice alter the values of the matrix.

2. List of contributions

In this paper, the authors developed new techniques based on the trilinear chaotic system to clearly increase the key space size and the robustness of the encryption techniques of digital images correlated. List of the advantages of this work are given below:

- The proposed chaotic system uses the quadratic and cubic coupling of the items $\boldsymbol{x_i}, \boldsymbol{y_i} \boldsymbol{and} \boldsymbol{z_i}$.
- The diagonalizations of the proposed system (2) increase the number of secret keys and the key space.
- During the diagonalization process, there are a total of 24 initial values of parameters.
- The Pixel Transform Table induced three $\boldsymbol{M} \times \boldsymbol{N}$ matrix keys and three vector keys of length MN.

* Corresponding author.
E-mail addresses: mahmoud.mossa@fci.bu.edu.eg (M.I. Moussa), eman.mohamed@fsc.bu.edu.eg (E.I. Abd El-Latif), ahmed.aboalatah@fci.bu.edu.eg (A.H. Abu El-Atta).

https://doi.org/10.1016/j.eij.2023.05.001
1110-8665/

- The proposed cryptosystem runs randomly multi-level diffusion of matrix values.
- The proposed approach increases the secret key space up to 10^{415}., which is the highest.
- Anaconda 4.8.3 is used to implement the algorithm using Python programming language.
- The computational study between the proposed algorithm with some pervious works shows high competitiveness regarding statistical analysis and security.

3. Related works

Over the past few decades, chaos-theoretic encryption algorithms have emerged as a powerful strategy for increasing security. By far most picture encryption calculations have been presented in view of ID &2D dimensional chaotic maps. Zhang et al. in 2005 [3] presented an algorithm for encrypting the given image that was depend on a chaotic map and a large encryption key by using a pixel shambling process applied an induced chaotic permutation matrix. Chong et al. [4] use Lorenz of a 3D chaotic system to improve image cryptography's performance and security. Xiangdong et al. in 2008 [5] introduced a tumultuous rearranging calculation utilizing an arranging change of a turbulent grouping to get address codes for picture interpretation. The disadvantages of image scrambling methods, such as increasing complexity and requiring knowledge of probability distributions, were avoided by their algorithm [8]. Juan et al. used a discrete chaotic method a safety key produced from the logistic system's initial conditions and parameters [7]. A Lorenz and Rosslere chaotic system was used by researchers in 2011 [9] to create an image encryption scheme with a large key space to increase security and complexity. An image crypto-system based on two maps was presented in 2011 by Keshari and Modani [12]; a chaotic map lattice to alter pixel values by rearranging the pixel's position and iterating the chaotic map under specific primary conditions. Using a skew tent chaotic system, Zhang, Liu, and others [10] presented a new image encryption algorithm in the same year. They shuffle the order in which all of the image's pixels are positioned during their work. Permutation-diffusion architecture served as the foundation for their algorithm. Another approach based on Chebyshev and 3D Logistic maps to encrypt a color image was presented in 2012 by Khade and Narnaware [15]. The proposed turbulent guides subbed the RGB parts, produced a key, and mixed the picture pixels. The pre-owned procedure relying upon a strategic guide used to portray a dark scale picture. All matrix value is swapped according to sequences from chaotic series using a digital matrix approach, enabling simultaneous pixel replacement and mixing. Wang et al. (2012) [16] proposed another calculation utilizing a three-layered framework of the variety picture and a two-layered Lorenz and tent tumultuous framework simultaneously to encode RGB parts. Their calculation has four stages. In the first place, the three-layered grid is changed over completely to a two-layered network and the low-recurrence wavelet coefficient is separated into covering blocks. After that, a completely random and chaotic sequence is used to scramble the pixel value diffusion for encryption. Song et al. [19] used Coupled Map Lattices (CML) and a neighborhood nonlinear map to define the framework based on the spatiotemporal advantages of the Nonlinear Chaotic Algorithm (NCA) chaos. Younes [21] wrote a helpful overview of various image encryption methods in 2016, describing several methods used between 2013 and 2015. Pak and Huang [22] wrote about a new chaotic system that was made by combining the results of two different 1D chaotic maps in 2017. Using on a linear-nonlinear-linear structure, their algorithm produced complete shuffling. Color images have received a lot of attention because they are full of information. The density of RGB components in the color image is determined by the numerical values of RGB components found in each pixel [14]. There have been numerous descriptions of encryption algorithms [6,11,15,16,18,23,27]. Because they disregard the correlations that exist among color components, these algorithms are more susceptible to attack. Utilizing 2D Hénon-Sine map to generate a pixel permutation, Wu et al. [24] suggested an algorithm based on pixel diffusion and a DNA approach in 2018. Wu et al. [25] encrypted an asymmetric multi-image using compressed sensing and a nonlinear operation in the cylindrical diffraction domain. In 2020, Yasser, et al. [27], produced approach to encrypt images that shuffles pixels and performs substitution operations with a chaotic system and DWT. El Shafai et al. [28] introduced a DNA encoding method for medical images based on a piecewise linear chaotic system.

4. Proposed chaos system

Fig. 1 shows the geometric visualisation of the trilinear interpolation chaotic system, the resulting function f at a given point (x, y, z) is equal to the sum of the multiplication of points at each corner subject to coefficient values of γ_{ijk} and the inside volume is equal to the partial volume diagonally opposite to the corner. The trilinear chaotic system contains $1D, 2D,$ and$3D$ equations resulting from log map and cubic maps. The proposed approach increases the quadratic and cubic coupling of the items $y_i^2, x_i^2, x_i y_i, x_i^3, z_i^3$ and provided more security to the system. The overall of the proposed system is simply:

$$f(x,y,z) = \sum\nolimits_{i=0}^{3}\sum\nolimits_{j=0}^{3}\sum\nolimits_{k=0}^{3}\gamma_{ijk}x^i y^j z^k \tag{1}$$

Six chaotic equations (2) will be derivative from Eq. (1) within the range (0, 1). The control parameters γ_{ijk} outside the range have no chaotic behavior.

The proposed system (2) comprises of $1D, 2D,$ and$3D$ equations derivative from (1).

Fig. 2 demonstrates that in the area (0,1), the $1D\&2D$ systems go to a chaotic state and produce a chaotic sequence that is subject to: 5.73< μ, μ_1 <12.87, 2.71< μ_2 <3.58, 0.039 <γ_1, γ_2 <0.251. Fig. 3 displays that, the 3D system go into a chaotic form in the area (0,1) subject to: 3.47 <λ <3.84 and 0.038 <β, α <0.041. The bifurcation diagram of $1D, 2D,$ and$3D$ are presented in Fig. 4 and Fig. 5 respectively.

5. Chaotic map diagonalizing

The proposed method in (2) can be denoted as the set $\boldsymbol{T}$ of bijections as follows:

$$T = (T_1(x),\ T_2(x), T_3(x),\ T_2(y), T_3(y), T_3(z)) \tag{3}$$

where P is a prime number and bijection $T_i(..) \in T$ can be defined as:

$$T_i\begin{bmatrix} x_{n+1} \\ y_{n+1} \\ z_{n+1} \end{bmatrix} = \left(\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}\right) \times \begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix} mod\boldsymbol{p} \tag{4}$$

$$T_i\begin{bmatrix} x_{n+1} \\ y_{n+1} \\ z_{n+1} \end{bmatrix} = A \times \begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix} mod\boldsymbol{p} \tag{5}$$

The matrix A is invertible if $gcd(|A|,\ p) = 1$ is met and $|A| \neq 0$. The inverse of eq. (5) is the Eq.(5′)

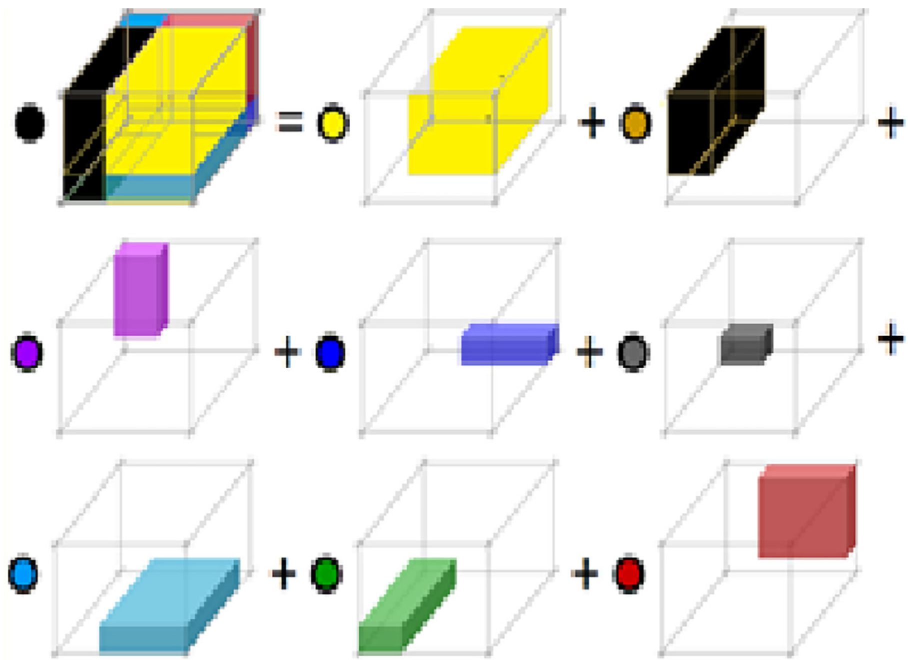


Fig. 1. Visualization of trilinear interpolation.

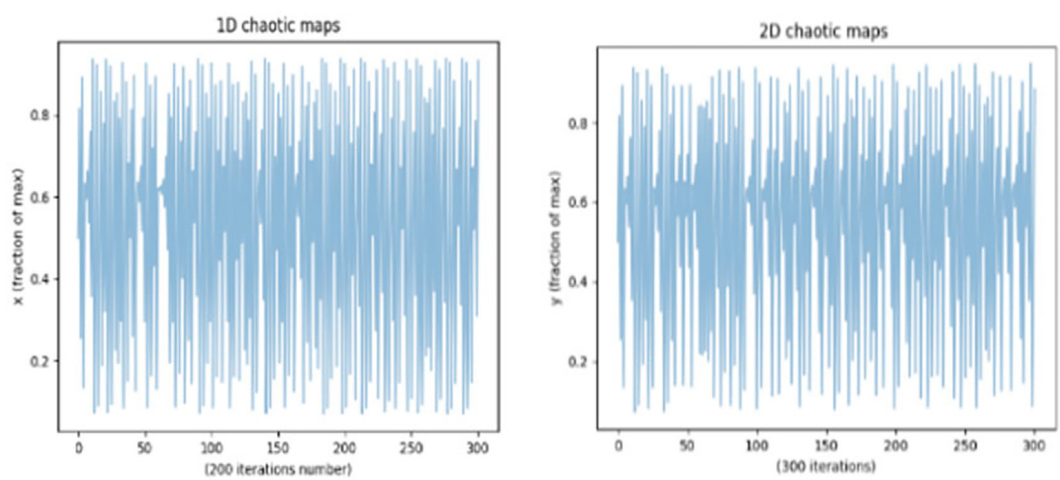


Fig. 2. The performance of the (1,2)-D chaotic map in first 300 iteration at $\mu = \mu_1$ = 6.27, μ_2 = 3.18 and $\gamma_1 = \gamma_2 = 0.111$ in *x-y* plane.

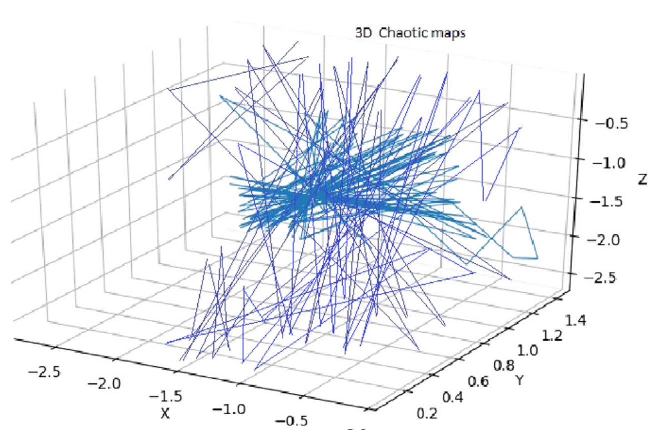


Fig. 3. The performance of the 3D chaotic map in first 1000 iteration at λ = 3.54, β = 0.036, and α = 0.039 in xyz space.

$$T_i \begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix} = A^{-1} \times \begin{bmatrix} x_{n+1} \\ y_{n+1} \\ z_{n+1} \end{bmatrix} mod\boldsymbol{p}. \tag{5.1}$$

Finding a new x', y', z' system requires diagonalizing the equations in (2). In system (2), the diagonalizations of the quadratic form in 1D and 2D are $(x'_{n+1})^T H x'_{n+1}$ $(x'_{n+1})^T D x'_{n+1}$, $(y'_{n+1})^T Q y'_{n+1}$, where H, D and Q are 2×2 matrices within four parameters for each. In addition, the number of parameters increases to nine when the three variables x, y, and z are diagonalized in their 3D form [11].

6. The proposed algorithm

A. Key Generation

The RGB image is converted to three parts with $M \times N$ matrices, with M rows and N columns of pixels for every component. Three

$M \times N$ matrix keys (M_r, M_b, M_g), and three vector keys (V_x, V_y, V_z) of length MN are generated by the proposed system used a new method called Pixel Transform Table (PTT). In addition, three sequences of real numbers are generated by $T_1(x)$, $T_2(x)$, $and T_2(y)$ and then converted to (M_r, M_b, M_g), and the chaotic maps $T_3(x)$, $T_3(y)$, and $T_3(z)$ produce three extra sequences which are changed into (V_x, V_y, V_z). The PTT algorithm is displayed below.

PTT Algorithm

Input: The chaotic equation
Output: T_j map
1. Set the system's chaotic parameters.
2. Repeat the Eq. (2) to create $S_j(i)$ sequences
$S_j(i) = \{\mathscr{S}_1(\mathrm{i}), \mathscr{S}_2(\mathrm{i}), .., \mathscr{S}_6(i)\}$.
where $j = 1 \rightarrow 6$, and $i = 1, 2, .., \rho$.
3. For each j, generate $S_j^I(\rho)$ integer sequences as:
$S_j^I(\rho) = \lfloor \left(S_j(i) \mathrm{x}\, 10^{14}\right) \rfloor \, mod\, \rho$
4. For each $j = 1 \rightarrow 6$, calculate the position of values $S^j(\,)$ as:
$S^j(\rho)$= Sort$\left(S_j^I(\rho)\right)$
then construct set of transfer
$T^j = \{t_1(i), t_2(i), .., t_6(i)\}$,
where the value $S_j^I(t_j(i)) = S^j[i]$, $i = 1, 2, .., \rho$.
5. The sequences ($t_1(i)$, $t_2(i)$, $t_3(i)$) are converted to three $M \times N$ keys (M_r, M_b, M_g), and the sequences ($t_4(i)$, $t_5(i)$, $t_6(i)$) are converted to the three keys (V_x, V_y, V_z) of length MN, respectively.

B. Image Encryption Algorithm (IEA)

IEA algorithm involves three phases as follows:

$IEA : PlainImage(P) \rightarrow CipherImage P_5$

Phase 1:

Along with the alteration of pixels' values, we describe a secure block shuffling tool. The first diffusion level, or random change in RBG pixel values, is the crucial effect of PPT in this process: The RBG's random pixel values, or first diffusion level, are as follows:

$$R^r = M_r\, R\, (M_r)^T\, mod\, 256$$

$$B^b = M_b\, B\, (M_b)^T mod\, 256$$

$$G^g = M_g\, G\, (M_g)^T mod\, 256$$

In every matrix (R^r, B^b, G^g), odd index $(2k+1)$ row is exchanged with an even index $(2k)$ row, and repeat this process for the columns, as shown in Fig. 6. We replication the exchange process σ-times and become $(R^r(\sigma), B^b(\sigma), G^g(\sigma))$.

Phase 2: Utilizing the two vector keys (V_x, V_y), the positions of pixel were muddled in columns and rows during this phase. We apply the second diffusion level by dividing the values of the components $R^r(\sigma)$, $B^b(\sigma)$, and $G^g(\sigma)$ by their size MN to produce the values of the pixels, i.e.

$$R'(\sigma) = \lfloor \frac{B^r(\sigma)}{\rho} \rfloor \mathrm{mod}\, 256$$

$$B'(\sigma) = \lfloor \frac{B^b(\sigma)}{\rho} \rfloor \mathrm{mod}\, 256$$

$$G'(\sigma) = \lfloor \frac{G^g(\sigma)}{\rho} \rfloor\, mod\, 256$$

Where $\rho \geq 3MN$, merge the matrices $R'(\sigma)$, $B'(\sigma)$, and $G'(\sigma)$ horizontally to obtain the $M \times 3N$ matrix P_1, and produce a sequence of $3MN$ numbers $Y_1 = y_1, ..., y_{3NM}$ from P_1. Permute the row $Y_1 = y_1, ..., y_{3NM}$ by the vector key V_x. We obtain the scramble vector $Y_1' = y\prime_1, ..., y\prime_{3NM}$. Reshape the vector Y_1' into three $M \times N$ matrices; $Ry(\sigma)$, $By(\sigma)$ and $Gy(\sigma)$.

Phase 3:

P_{now} is the current plain pixel value, D_{pre} is the previous cipher pixel value following the current diffusion, and P_{pre} is the previous plain value. D_{now} is the current ciphered pixel value following the current diffusion. Their underlying qualities in $Rz(\sigma)$, $Bz(\sigma)$, and $Gz(\sigma)$ are equivalent to nothing. Monitor columns in P_3 to set three vectors $V(Rz(\sigma))$, $V(Bz(\sigma))$, and $V(Gz(\sigma))$ each with length MN. Utilizing the vector product with the key vector V_z, we apply the third level of pixel diffusion to these vectors in the following manner:

$$\begin{cases} Zz^r = \left(V(Rz(\sigma)) \times V_g\right) mod256 \\ Zz^b = \left(V(Bz(\sigma)) \times V_g\right) mod256 \\ Zz^g = \left(V(Gz(\sigma)) \times V_g\right) mod256 \end{cases} \tag{6}$$

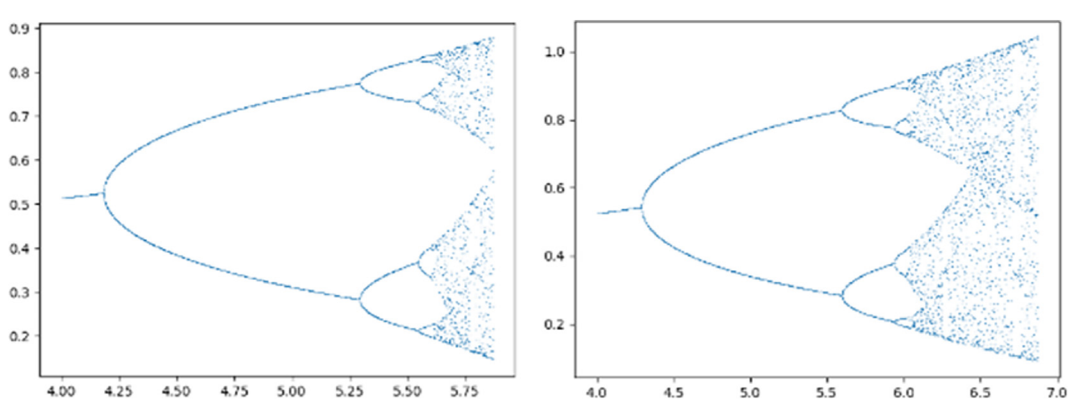


Fig. 4. Bifurcation diagram for 1D and 2D chaotic.

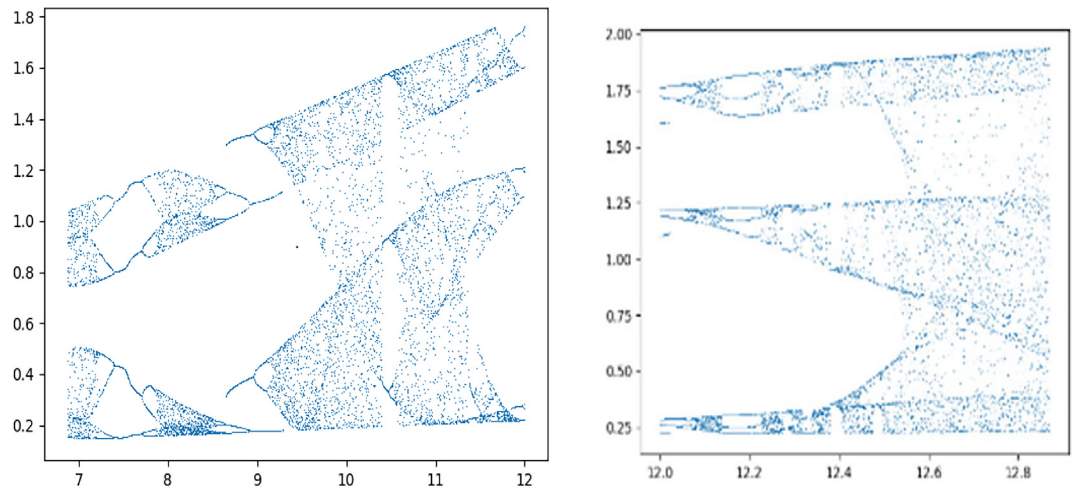


Fig. 5. Bifurcation diagram for 3D chaotic.

Where Zz^r, Zz^b, and Zz^g are the generating vectors that contains the chaotic values of V_z. In the next section, we apply 4th and final level of random diffusion where [(:)] refers to the components Zz^r, Zz^b, and Zz^gwhile [:] refers to a random value.

Random Diffusion algorithm

Input: $V_z = (Zz^r, Zz^b, Zz^g)$
Output: Vector $(D_{now}(Zz^r), D_{now}\left(Zz^b\right), D_{now}(Zz^g))$
Produce $\varepsilon_{1l}\varepsilon_{2l}$, ε_{3l} random values as following:
 ε_{1l} = rand () % 3
 ε_{2l}, ε_{3l} =select two random value from V_z mod 256
for ω in MN range
 If ε_{1l}= 0 then
  $D_{now}(Zz^r)$=$\left(\varepsilon_{2l}.P_{now}(Zz^r) + \varepsilon_{3l}.(D_{pre}Zz^r \times P_{pre}Zz^r)\right)$mod 256. (7)
 else If ε_{1l}=1
$D_{now}(Zz^b)$=$\left(\varepsilon_{2l}.P_{now}\left(Zz^b\right) + \varepsilon_{3l}.(D_{pre}Zz^b \times P_{pre}Zz^b)\right)$mod 256. (8)
 else If ε_{1l}=2
$D_{now}(Zz^g)$=$\left(\varepsilon_{2l}.P_{now}(Zz^g) + \varepsilon_{3l}.(D_{pre}Zz^g \times P_{pre}Zz^g)\right)$mod 256. (9)
end For.

Reshape $D_{now}(Zz^r), D_{now}\left(Zz^b\right)$, and $D_{now}(Zz^g)$ into three $M \times N$ matrices; $Rz(\sigma)$, $Bz(\sigma)$, and $Gz(\sigma)$. We obtain the components of converted image P_5 of size $M \times N$.

C. Image Decryption Algorithm (IDA)

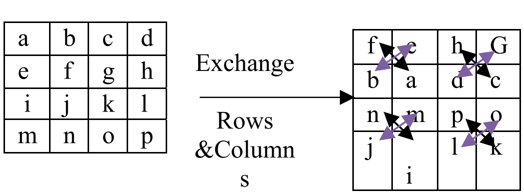


Fig. 6. Swapping rows & columns.

The steps of the encryption system with the actuated 6 keys are displayed in Fig. 7. Fig. 8 depicts the decryption steps and the six keys. Similar to the image encryption pseudocode, the image decryption algorithm (IDA) works in the opposite way.

Step 1: using all random number in IDA.

Step 2: Apply 4th level of random by using the following equations:

$$Zz^r = \left(\frac{D_{now}(Zz^r) - \varepsilon_{3l}(D_{pre}Zz^r \times P_{pre}Zz^r)}{\varepsilon_{2l}}\right)\text{mod}256 \tag{10}$$

$$Zz^b = \left(\frac{D_{now}\left(Zz^b\right) - \varepsilon_{3l}(D_{pre}Zz^b \times P_{pre}Zz^b)}{\varepsilon_{2l}}\right)\text{mod}256 \tag{11}$$

$$Zz^g = \left(\frac{D_{now}(Zz^g) - \varepsilon_{3l}(D_{pre}Zz^g \times P_{pre}Zz^g)}{\varepsilon_{2l}}\right)\text{mod}256 \tag{12}$$

Step 3: Opposite 3rd level of pixel diffusion (inverse of Eq. (6)). Calculate $V(Rz(\sigma))$, $V(Bz(\sigma))$ and $V(Gz(\sigma))$ from the previous equations by using.

$$\begin{cases} V(Rz(\sigma)) = \left(\frac{Zz^r \times V_g}{V_g.V_g}\right)mod256 + \tau V_g \\ V(Bz(\sigma)) = \left(\frac{Zz^b \times V_g}{V_g.V_g}\right)mod256 + \tau V_g \\ V(Gz(\sigma)) = \left(\frac{Zz^g \times V_g}{V_g.V_g}\right)mod256 + \tau V_g \end{cases} \tag{13}$$

Step 4: utilizing the reverse vectors $\left(V_x^{-1}, V_y^{-1}\right)$.

Step 5: Inverse 2nd level of random using the following calculations:

$$B^r(\sigma) = \lfloor \rho.R'(\sigma) \rfloor \bmod 256$$

$$B^b(\sigma) = \lfloor \rho.B'(\sigma) \rfloor \bmod 256 \tag{14}$$

$$G^g(\sigma) = \lfloor \rho.G'(\sigma) \rfloor \, mod\, 256$$

Step 6: To get rid of the effect of randomly switching the columns and rows, return to the reverse paths.

Step 7: The following is an inverted first random diffusion:

$$\mathrm{R} = (\mathrm{M_r})^{-1}\,\mathrm{R^r}((\mathrm{M_r})^{\mathrm{T}})^{-1} \bmod 256$$

$$\mathrm{B} = (\mathrm{M_b})^{-1}\mathrm{B^b}((\mathrm{M_b})^{\mathrm{T}})^{-1} \bmod 256 \tag{15}$$

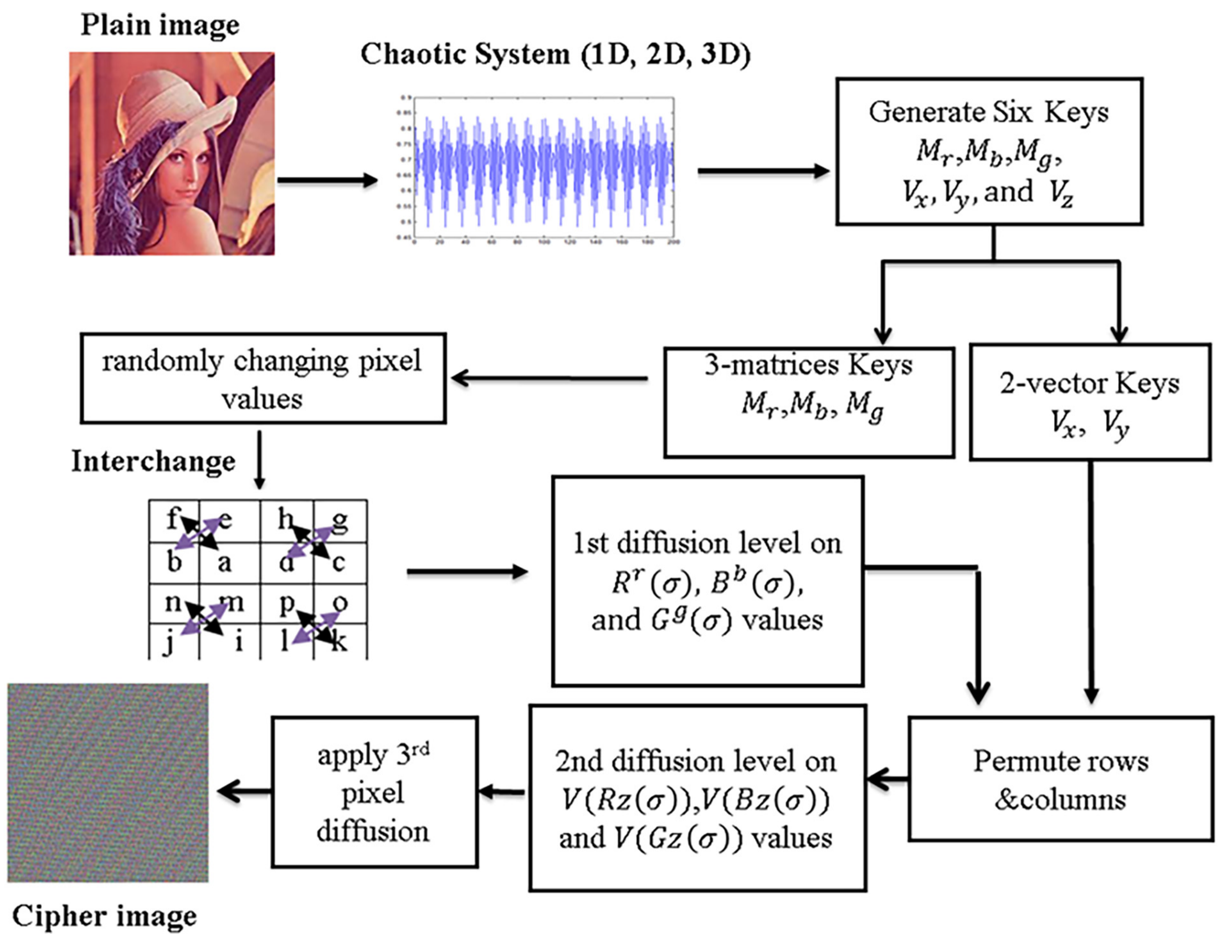


Fig. 7. The steps of Image Encryption Algorithm.

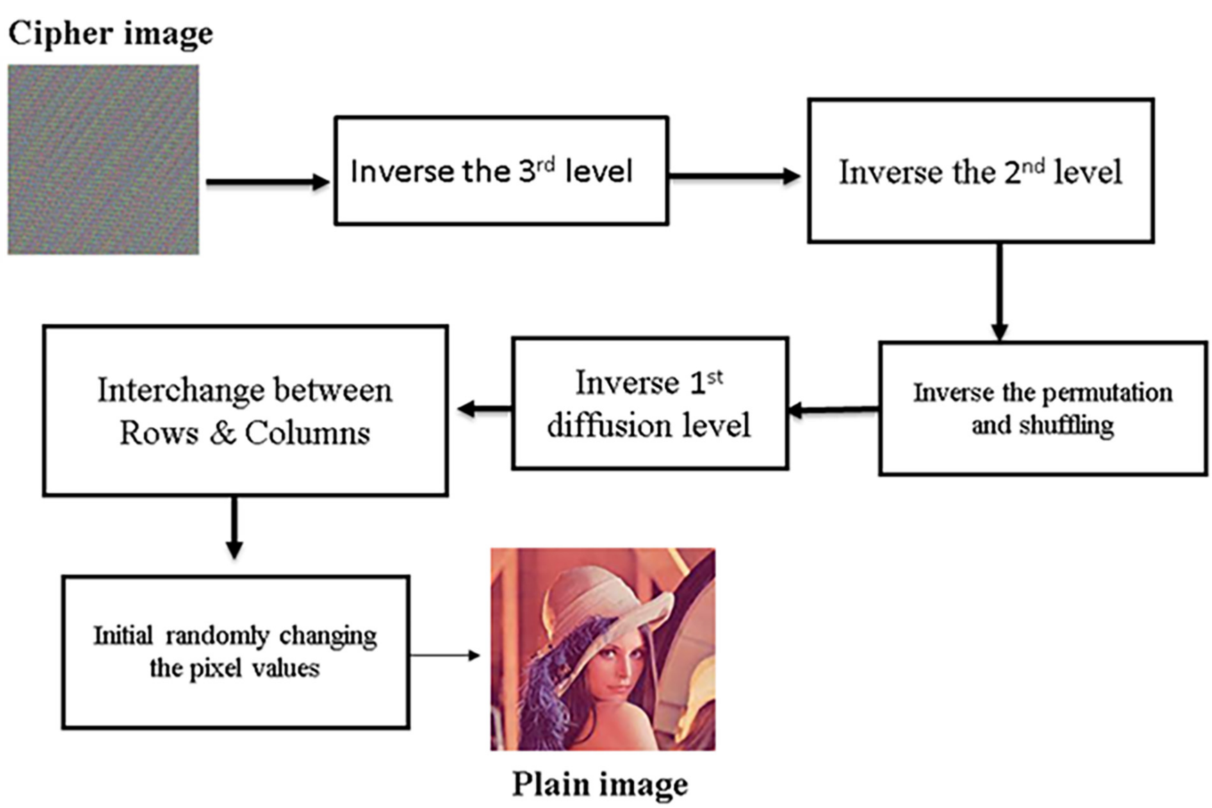


Fig. 8. The steps of Image Decryption Algorithm.

$$G = (M_g)^{-1} G^g ((M_g)^T)^{-1} \bmod 256$$

The plain image P is recovered through these seven steps. Finally, IDA can be denoted by the following formula:

IDA: Cipher Image $P_5 \rightarrow$ Plain Image (P).

D. The computational complicity

For a small constant, the running time of the PPT is $O(\boldsymbol{\rho} = MN + \epsilon)$, and the running time of steps 1 through 7 is $O(MN)$ which is a linear function of the image's size. The experi-

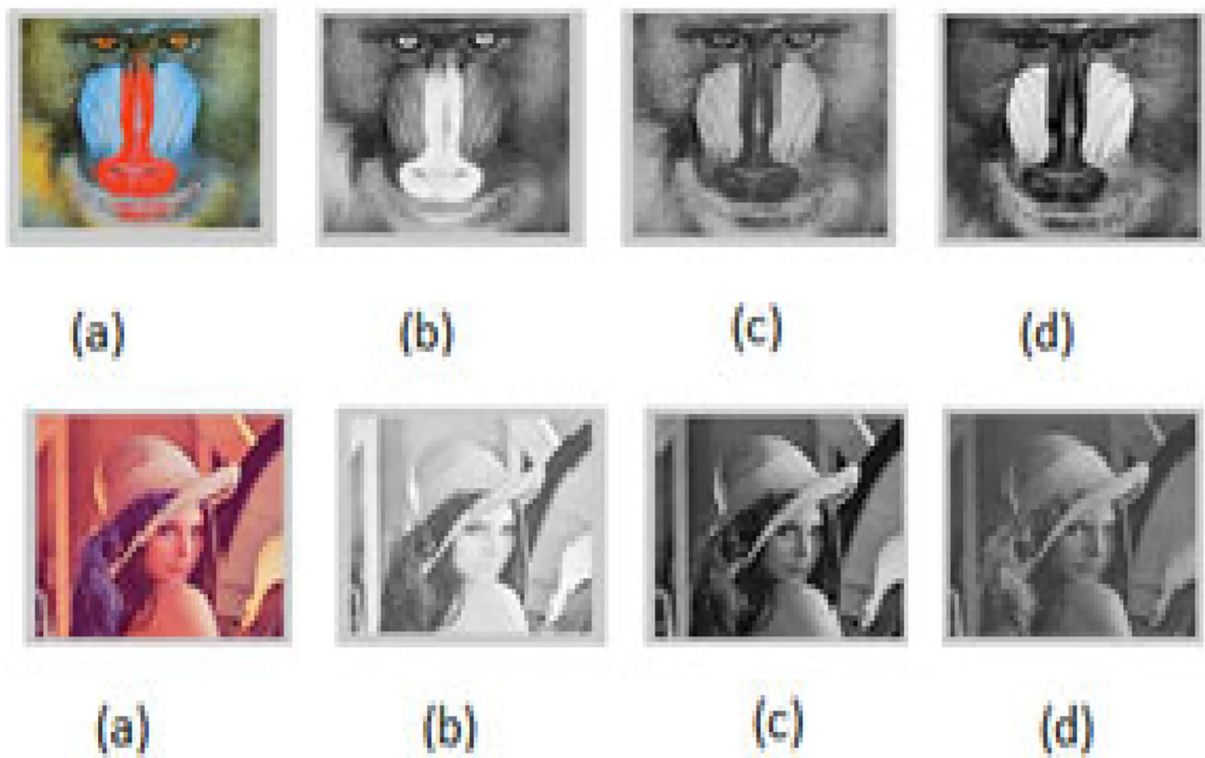


Fig. 9. Before the encryption process, (a-d) displays the color images of "Baboon" and "Lena" as well as the RGB values of those color images.

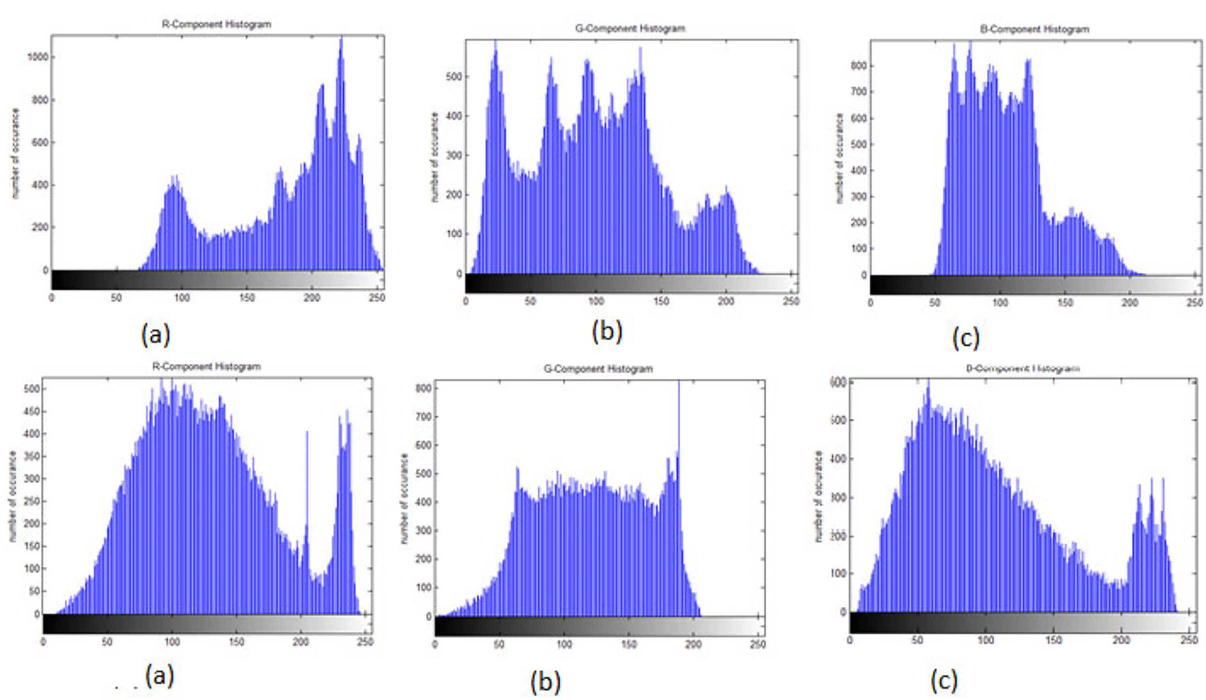


Fig. 10. Histograms of "baboon" and "Lena" before encryption respectively.

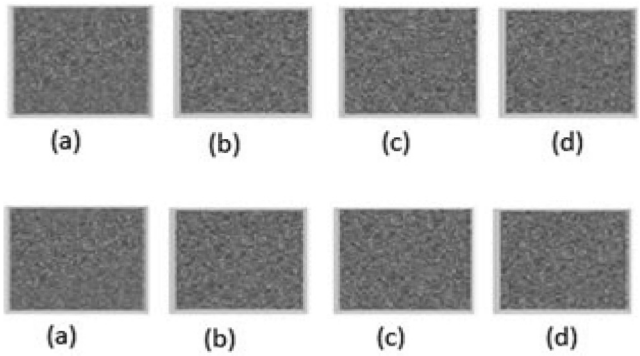


Fig. 11. The ciphered of Baboon" and "Lena" images from (a-d).

mental procedure determines the practicability of our IEA/IDA approach's consumption time.

The processor Intel(R) Center (TM) i7-8550U computer chip @ 1.80 GHz 2.00 GHz and 8 GB Slam is used in IEA/IDA is practicable. On a 256 image P/P_5 of size 256 $\times$ 256, the encryption and decryption processes are carried out with the help of an 8 GB RAM and an Intel(R) Core(TM) i7-8550U CPU running at 1.80 GHz 2.00 GHz. The average time to consume is<4.01 ms. Our running opportunity is near the running time in [2].

7. Experimental results

Anaconda 4.8.3 is used to implement the system. We have chosen Python as the programming language for our development. An Intel(R) Core(TM) i7-8550U CPU running at 1.80 GHz 2.00 GHz and 8 GB of RAM are used to test the implementation in Windows 10 64-bit OS. Take the underlying boundaries and values: 1D ($\boldsymbol{\mu} = 6.27, \boldsymbol{x}_0 = 1.2 \times 10^{18}$), 2D ($\boldsymbol{\mu}_1 = 6.27$, $\boldsymbol{\mu}_2 = 3.18$, $\gamma_1 = 0.111$,$\gamma_2 = 0.111$,$\boldsymbol{x}_0 = 2.3 \times 10^{18}$,$\boldsymbol{y}_0 = 1.2 \times 10^{18}$), 3D ($\boldsymbol{\lambda}$ = 3.54, $\boldsymbol{\beta}$= 0.036, $\boldsymbol{\alpha} = 0.039$, $\boldsymbol{x}_0 = 3.4 \times 10^{18}$, $\boldsymbol{y}_0 = 4.5 \times 10^{18}$, $\boldsymbol{z}_0 = 5.6 \times 10^{18}$), to encrypt the input image of the 256 $\times$ 256 "Baboon" and "Lena" images, as depicted in Fig. 9(a – d). Fig. 10(a – c) depicts the "baboon," "Lena," and their components prior to encryption. The histograms show the correlation among the pixels in the plain images "baboon" and "Lena" at each level of color density.

The encrypted two images are depicted in Fig. 11 (a–d) and the histograms of two images are depicted in Fig. 12(a – c). The histograms show how the ciphered RGB pixels are consistently correlated with the pixels at each level.

8. Security analysis

A. The Key Size

Brute-force attacks must be rendered ineffective by the encryption's whole number of special keys. Six values are used in our

Fig. 13. Result of decrypted image using correct parameter.

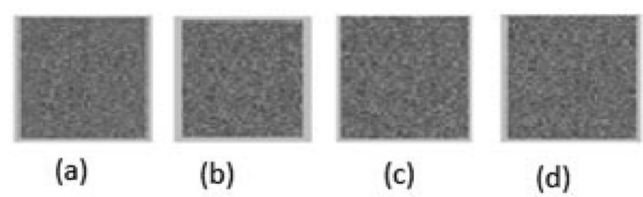


Fig. 14. Result of decrypted image using wrong parameter.

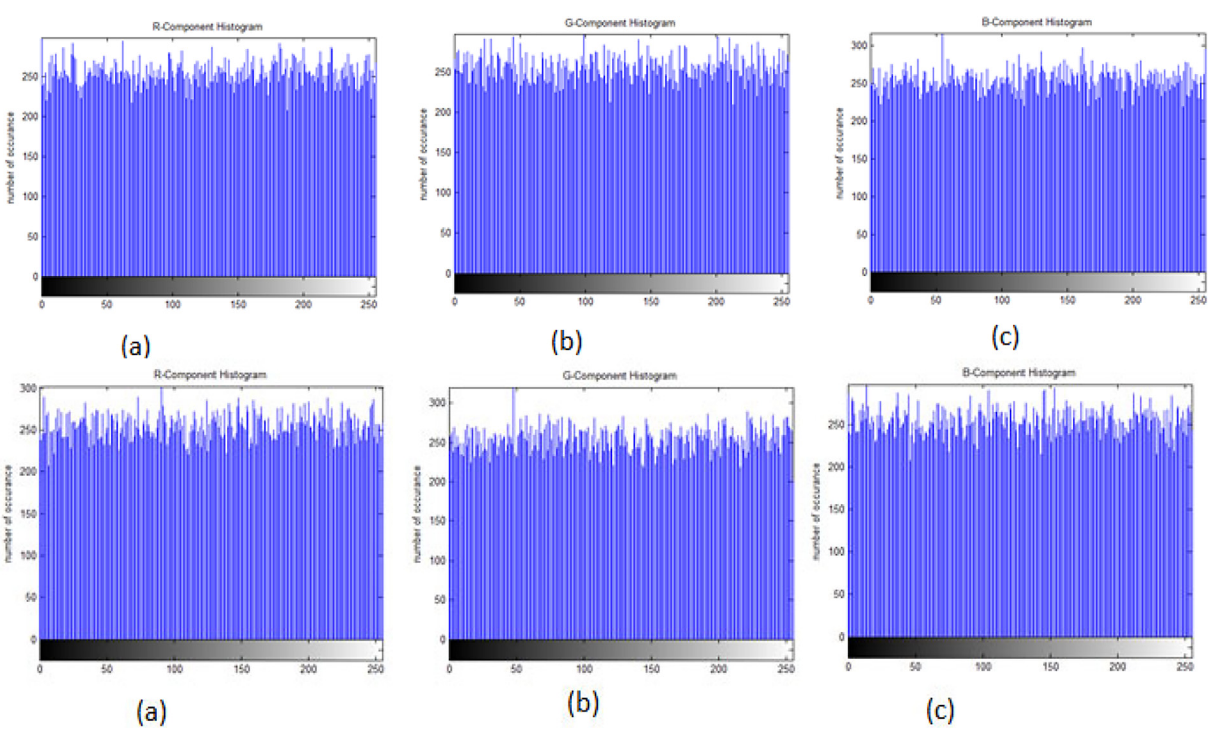


Fig. 12. (a–c): Histograms after encryption for two images.

system:$x_0^{1D}, x_0^{2D}, y_0^{2D}, x_0^{3D}, y_0^{3D}$, z_0^{3D} and the eight parameters of μ, μ_1, μ_2, γ_1, γ_2, λ, β, α, as secret keys. Wang et al. [25] proved that if the precision is 10^{-17}, the keys $K_{x_0^{1D}} = K_{x_0^{2D}} = K_{y_0^{2D}} = K_{x_0^{3D}} = K_{y_0^{3D}} = K_{z_0^{3D}} = 10^{17}$, $K_\mu = K_{\mu_1} = K_{\mu_2}$, $K_{\gamma_1} = K_{\gamma_2} = K_\lambda = K_\beta = K_\alpha = 0.5 \times 10^{17}$.

Let the input images have size 256×256. The number of iterations over six maps I_0 is $6 \times (3 \times M \times N) = 6 \times (3 \times 256 \times 256) \approx 2^{20} \approx 10^7$. The key space spreads to $\approx 1.953 \times 10^7 \times 10^{235} = 1.953 \times 10^{242}$. the space of key is larger than 2^{138}, 2^{58}, 10^{140}, 2^{256} [17,26,30]. It is greater than $2^{448} = 7.8 \times 10^{134}$, the highest number of key spaces cited in the study [23]. Within the diagonalization form (2), there are 24 initial values. Our key space increments it up to 10^{415}. The key space described by the proposed algorithms is large enough to withstand brute-force attacks.

B. The Sensitivity Examination of the Secret Keys.

Little differences between keys generate another cipher images. The Baboon's decrypted image with the correct key of λ = 3.66 is depicted in Fig. 13. On the other hand, the Baboon image is decrypted in Fig. 14 using an incorrect encryption key equal to λ = 3.66000000000000001. The algorithm was made more sensitive to the key thanks to this. A little change in the key will bring about a completely unique unscrambling result, and the attacker will not have the option to get to the right plain picture.

C. Adjacent Pixels Correlation Analysis

The following formula is used to calculate the vertical, horizontal, and diagonal correlation between 3000 randomly selected nearby pixels:

$$E(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i) \tag{16}$$

$$E(y) = \frac{1}{N}\sum_{i=1}^{N}(y_i) \tag{17}$$

$$D(x) = \sum_{i=1}^{N}(x_i - E(x))^2 \tag{18}$$

$$D(y) = \sum_{i=1}^{N}(y_i - E(y))^2 \tag{19}$$

$$cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \tag{20}$$

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{21}$$

Tables 1-3 and Fig. 15 display the values of two adjacent pixels in three formats:

Table 1
the correlation values for images (*V/D*).

Image	V/D	Plain image	Cipher Image	
			Proposed approach	Ref [29]
Lena	*R*	0.9238	−0.0023	−0.0016
	G	0.9479	0.0027	−0.0011
	B	0.8785	−0.00109	−0.0013
Baboon	*R*	0.9527	−0.0025	0.0002
	G	0.9283	−0.0048	0.0001
	B	0.9563	0.00209	0.0004

Table 2
the correlation values for images (*H/D*).

Image	H/D	Plain image	Cipher Image	
			Proposed approach	Ref [20]
Lena	*R*	0.9783	−0.0019	−0.00092
	G	0.9795	0.0036	−0.0038
	B	0.9594	−0.00068	−0.0020
Baboon	*R*	0.9413	−0.0018	0.0062
	G	0.8796	−0.0056	−0.0060
	B	0.9164	0.0019	0.0077

Table 3
the correlation values for images (*D/D*).

Image	D/D	Plain image	Cipher Image	
			Proposed approach	Ref [10]
Lena	*R*	0.9685	−0.0013	−0.0008482
	G	0.9574	0.0019	−0.0008482
	B	0.8994	−0.00031	−0.0008482
Baboon	*R*	0.6471	−0.0022	0.00370914
	G	0.9567	−0.0035	0.00370914
	B	0.9355	0.0013	0.00370914

- Horizontal direction (H/D),
- Vertical direction (V/D),
- Diagonal direction (D/D).

They show a lot of concentration in the plain image and have a tendency of one in two neighboring pixels in the ciphered image. This indicates that the neighboring pixels in the ciphered image are random, and the encryption effects resist statistical attack.

D. Detection of different Attack

To calculate the capacity, the Pixel Change Rate Number, and the Unified Average Change Intensity (UACI) tests are applied. The UACI test compares the average intensity of two images, while the NPCR test distinguishes between the numbers of distinct pixels in each [13]. UACI and NPCR can calculated as following:

$$UACI_{R,G,B} = \left(\frac{\frac{\sum_{ij}\left|C_{R,G,B}(i,j) - C'_{R,G,B}(i,j)\right|}{255}}{M \times N}\right) \times 100 \tag{22}$$

$$NPCR_{R,G,B} = \left(\frac{\sum_{ij} D_{R,G,B}(i,j)}{M \times N}\right) \times 100 \tag{23}$$

$$D_{R,G,B}(i,j) = \begin{cases} 0 if C_{R,G,B}(i,j) = C'_{R,G,B}(i,j) \\ 1 Otherwise \end{cases} \tag{24}$$

Where, M and N are the width and height of the image, $C_{R,G,B}(i,j)$ and $C'_{R,G,B}(i,j)$ are value of pixel for the two encrypted images when one pixel of the first plain picture is modified.

Table 4 indications $NPCR_{R,G,B}$ greater than 99.55% and values of $UACI_{R,G,B}$ greater than 33.44%. The previous work demonstrate that our method is extremely sensitive to minute changes in the input image; even if the two input images differ by just one bit, the output images still differ somewhat.

E. PSNR Analysis

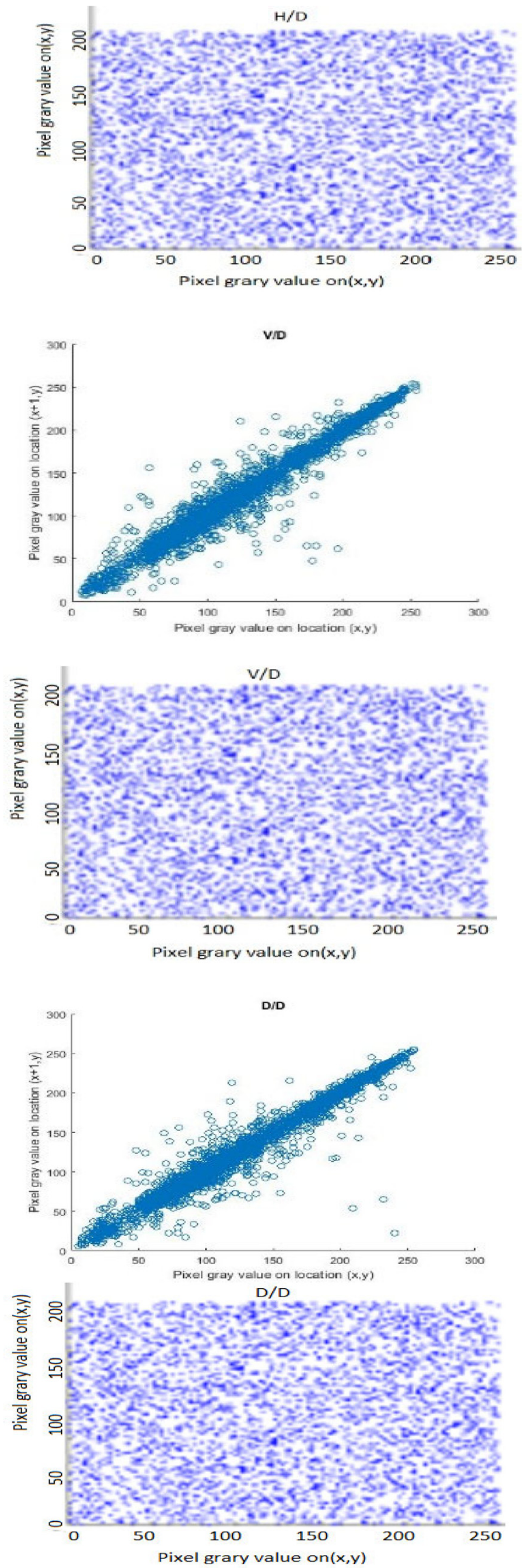


Fig. 15. The distribution in the three directions for the two images.

Peak Signal-to-Noise Ratio (PSNR) is utilized basically as a quality measurement and it can calculate as following:

Table 4
NPCR and UACI *Results.*

Image	Proposed approach			Ref [10]	
		NPCR%	UACI %	NPCR%	UACI %
Lena	R	99.5157	33.4228	99.6052	33.4132
	G	99.4792	33.4441		
	B	99.53342	33.4897		
Baboon	R	99.5845	33.6425	99.6227	33.4865
	G	99.6678	33.3433		
	B	99.6557	33.4726		

Table 5
PSNR Results.

Image	Proposed approach			Reff [20]		
	R	G	B	R	G	B
Lena	7.968	8.888	9.759	7.8992	8.576	9.678
Baboon	8.398	9.457	8.970	8.958	9.414	8.415

$$PSNR_{R,\,G,\,B} = 20 * log_{10}\left(\frac{255}{sqrt(MSE_{R,\,G,\,B})}\right) \tag{25}$$

$$MSE_{R,\,G,\,B} = \sum_i \sum_j \frac{C_{R,\,G,\,B}(i,j) - C'_{R,G,B}(i,j)}{MxN} \tag{26}$$

The mean square error (MSE) portrays the distinction in the qualities from [0 255] among the plain and the encoded picture. Table 5 also shows the differences in PSNR values among the original and encrypted image. Statistical attacks are more resistant with our method.

9. Conclusion

For image security, we recommended a key space algorithm that could withstand brute-force attacks. Three keys are generated by the proposed cryptosystem in the form of a MN-length vector matrix and three keys in the $\mathbf{M} \times \mathbf{N}$ square matrix. Multi-level diffusion of pixel values and variety among shuffling and scrambling of rows and columns in the color components of the plain image are the foundations of our cryptosystem for image. The computational comparison of the proposed procedure to other cryptosystems revealed that the proposed system has large key spaces and a high level of security. The benefit of the current system is utilizing staggered encryption in view of huge key space.

Declaration of Competing Interest

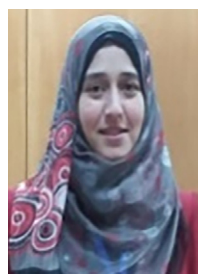
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Matthews Robert AJ. On the derivation of a chaotic encryption algorithm. J Cryptologia 1989;13(1):29–42.
[2] Chang CC, Hwang M, Chen T. A new encryption algorithm for image cryptosystem. Journal of system and software 2001;58:83–91.
[3] Zhang H, Wang XF, Li ZH, Liu DH. A Fast Image Encryption Algorithm Based on Chaos System and Henon Map. Journal of computer Research and Development 2005;42(12):2137–42.
[4] C. Fu, Z. Zhang, and Y. Cao. "An Improved Image Encryption Algorithm Based on Chaotic Maps." Third International Conference on Natural Computation (ICNC 2007). Vol. 5. 2007. 24-27.
[5] LIU Xiangdong, Zhang Junxing, Zhang Jinhai, He Xiqin." Image Scrambling Algorithm Based on Chaos Theory and Sorting Transformation". IJCSNS International Journal of Computer Science and Network Security. VOL.8 No.1. January 2008.

[6] Hongjuan Liu, Zhiliang Zhu, Huiyan Jiang, and Beilei Wang." A Novel Image Encryption Algorithm Based on Improved 3D Chaotic Cat Map". The 9th International Conference for Young Computer Scientists. 978-0-7695-3398. 2008. IEEE.
[7] Juan Li, Yong Feng, Xuqiang Yang." Discrete Chaotic based 3D Image encryption Scheme". Symposium on Photonics and Optoelectronics. September 2009. IEEE.
[8] Mazloom S, Eftekhari-Moghadam AM. Colour image encryption based on coupled nonlinear chaotic map". Chaos Solitons Fractals 2009;42(3):1745–54.
[9] Alsafasfeh QH, Arfoa Aouda A. Image Encryption Based on the General Approach for Multiple Chaotic Systems". Journal of Signal and Information Processing 2011;2:238–44.
[10] Zhang GA, Qing Liu B. A novel image encryption method based on total shuffling scheme". Opt Commun 2011;284:2775–80.
[11] Howard Anton and Chris Rorres. "Elementary linear Algebra, Applications Version". Tenth Edition. WILEY. 2011.
[12] Sudhir Keshari, Dr. S. G. Modani." Image Encryption Algorithm based on Chaotic Map Lattice and Arnold cat map for Secure Transmission". IJCST. Vol. 2. Issue 1. March 2011.
[13] Yue Wu, Noonan JP, Agaian S. NPCR and UACI randomness tests for image encryption. Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT), April Edition 2011.
[14] Seyed Mohammad Seyedzadeh. Sattar Mirzakuchaki "A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map". Signal Process 2012;92:1202–15.
[15] Khade PN, Narnaware M. 3D Chaotic Functions for Image Encryption", IJCSI. International Journal of Computer Science Issues 2012;9(Issue 3, No 1):May.
[16] Wang X, Teng L, Qin X. LinTeng, Xue Qin", A novel colour image encryption algorithm based on chaos". Signal Process 2012;92(4):1101–8.
[17] X. Wang and L. Teng." A bit-level image encryption algorithm based on spatiotemporal chaotic system and self-adaptive". Optics.
[18] . Communications 2012;285(20):4048–54.
[19] Hoppen C, Kohayakawa Y, Moreira C, Ráth B, Sampaio R. Limits of permutation sequences". Journal of Combinatorial Theory, S B 2013;103:93–113.
[20] Song C-Y, Qiao Y-L, Zhang X-Z. An image encryption scheme based on new spatiotemporal chaos. Optik-International Journal for Light and Electron Optics 2013;124(18):3329–34.
[21] NF Elabady, MI Moussa, HM Abdalkader, SF Sabbeh". Image Encryption Based on New One-Dimensional Chaotic Map"", International Conference on Engineering and Technology (ICET). 19-20. April 2014. Cairo, Egypt.
[22] Mohammed A. B. Younes. "Literature Survey on Different Techniques of Image Encryption". International Journal of Scientific & Engineering Research. Vol. 7. Issue 1. 2016.
[23] Pak C, Huang L. Lilian Huang" A new color image encryption using combination of the 1D chaoticmap". Signal Process 2017;138:129–37.
[24] M. Kumari, S. Gupta, P. Sardana. "A survey of image encryption algorithms". 3D Research. Volume 8, Number 4. (2017). Article No.:148.
[25] Wu J, Liao X, Yang B. Image encryption using 2D Hénon-Sine map and DNA approach. Signal Process 2018;153:11–23.
[26] Wu C, Wang Y, Chen Y, Wang J, Wang Q. Asymmetric encryption of multiple-image based on compressed sensing and phase-truncation in cylindrical diffraction domain. Opt Commun 2019;431:203–9.
[27] Tang Z, Yang Ye, Xu S, Yu C, Zhang X. Image Encryption with Double Spiral Scans and Chaotic Maps. Security and Communication Networks 2019;2019:1–15.
[28] Yasser I, Khalifa F, Mohamed MA, Samrah AS. A New Image Encryption Scheme Based on Hybrid Chaotic Maps. Complexity 2020;2020:1–23.
[29] El-Shafai W, Khallaf F, El-Rabaie E-S, El-Samie FEA. Robust medical image encryption based on DNA-chaos cryptosystem for secure telemedicine and healthcare applications. J Ambient Intell Human Comput 2021;12(10):9007–35.
[30] Shafique A, Hazzazi MM, Alharbi AR, Hussain I. Iqtadar Hussain, "Integration of Spatial and Frequency Domain Encryption for Digital Images". Access IEEE 2021;9:149943–54.

Mahmoud I. Moussa is an associate professor of computer science at the faculty of computers & artificial Intelligence, Benha University in Egypt. He received his Ph.D. in Parallel Algorithms (mainly in parallel graph algorithms) from faculty of informatics at Karlsruhe Institute of Technology-KIT, Germany. His research interests span both theoretical computer science and information security. Much of his work has been on improving the understanding, design, and performance of algorithms and analysis, mainly through the application of graph algorithms, bioinformatics, as well as Steganography and Cryptography. Moussa's work includes a prediction method for biological activity using random forests and kernel functions in Support Vector Machine (SVM), image encryption using chaotic maps, a method for using smartphone devices efficiently and offloading only when necessary.



Eman I. Abd El-Latif received the M.Sc. degree and Ph. D. in computer science, at Faculty of Science, Benha University, Egypt, 2016 and 2020, respectively. She is currently working as lecturer at mathematics department, Benha University, Egypt. Her areas of research include Digital Forensics, Security (Encryption – Steganography), and image processing.

Ahmed H. Abu El-Atta received the BS and MS degrees in computer science from Benha University, Egypt in 2005 and 2011, respectively, and the PhD degree from Benha University, Egypt, in 2018. He works as a lecturer and researcher at Benha University, Egypt. His current research interests include graph theory, pattern recognition, NLP, and steganography.