# Enhancing the Security of Digital Image Encryption using Diagonalize Multidimensional Nonlinear Chaotic System

Mahmoud I. Moussa[1]
Computer Science Department
Faculty of Computer and Artificial Intelligence
Benha University, Egypt

Eman I. Abd El-Latif[2]
Mathematics Department
Faculty of Science, Benha University
Benha University, Egypt

Nawaz Majid[3]
Computer Science Department
Faculty of Science, Northern Border University (NBU)
KSA

***Abstract*—This paper describes a new efficient cryptosystem for the color image encryption technique, based on a combination of multidimensional proposed chaos systems. This chaos system consists of six bisections: $T_1(x), T_2(x), T_2(y), T_3(x), T_3(y)$, and $T_3(z)$. They induce three chaotic matrix keys and three chaotic vector keys. We use a multidimensional chaotic system together with an encryption algorithm to provide better security and wide key spaces. The proposed cryptosystem uses four levels of random pixel diffusions and permutations simultaneously and $\omega$ - times interchange between rows and columns. The correlations between the RGB components of the plain image are reduced. The level of security, the computational complexity, the quality of decoding a decrypted image under closure threat is improved. The simulation results showed that the algorithm shows a high level of security, and the assurance that the image recovered at the receiving point is identified as the image at the transmission point.**

***Keywords*—Chaotic system; encryption; decryption; image; algorithms; cryptosystem***

## I. Introduction

The world is moving towards digitizing all types of data. The digital use of data is hourly increasing. Digital images are at the forefront of the most used data on the internet. With the increase in the technology of the World Wide Web and the growth of its uses, the sharing of data over the internet has diversified, and among this data is the exchange of digital images. In many cases, the images are sensitive, and they contain information that is not publishable, such as images related to the health conditions of individuals, as well as secret military locations related to the national security of countries. Since the areas of communication via the Internet are open and vulnerable to attacks on data, the use of encryption for such images has become very important and necessary. A color image holds many rows and columns of pixels; these pixels contain quantized values that represent the degree of the color at any point in the image. Thus, the color image defined as a matrix of value pixels, each containing three numerical RGB components to describe the color of a tiny area. Since sensitive images may invite attacks from anywhere around the world, image security is an important issue. The aim of image encryption is to transform a plain image to different cipher one that is difficult to read [1]. Chaos theory is one of the most important security approaches used in encrypting images due to its capacity for mixing, sensitivity to initial conditions, control parameters, and completely random behavior. In 1989 [2], Matthews developed the first chaotic encryption technique. Many researchers are interested in presenting various algorithms to create a strong and robust encryption system for digital images, and some of them depend on the chaos system to induce random secret keys to maintain the confidentiality of images. In this paper, we used a multi-dimensional chaos system with many parameters to increase the key space. Since the proposed chaos system is developed and new, the chaotic and bifurcation behavior of it have been made and the values that enter the system into a chaos state have been determined. This paper presents a cryptosystem based on the proposed multi-dimensional chaotic maps and multiplexing frequent levels of shuffling, scramble, and pixel diffusion for the digital image component RGB. The practical results showed the intensity and resistance of the proposed algorithm compared to many related ones. The analysis proved a significant increase in the size of the encryption keys, which makes our algorithm outperform others against the brute-force attacks. His work introduced a low complexity algorithm for image cryptography using a multiple definition function for linear chaotic map denoted NPWLCM. Since then, many scientific papers have described image encryption based on chaotic systems. In this work, we describe a multi-chaotic system to encrypt the RGB components of color images simultaneously such that, the RGB components affect one another. The proposed six maps; ; $T_1(x), T_2(x), T_2(y), T_3(x), T_3(y)$, and $T_3(z)$ increase conjugation of the items $x_i^3, x_i^2, y_i^2, x_i y_i, y_i^3, z_i^2 x_i, \dots z_i^3$. Pixel Transform Table (PTT) procedure input these maps to output three $M \times N$ matrix keys $(M_r, M_b, M_g)$ and three vector keys $(V_r, V_b, V_g)$ of length $MN$.. The image's rows and columns are shuffled and scrambled using the vector keys. While the matrix keys are used to change the values of pixels four times to increase the complexity and thus the security of the cryptosystem. The remainder of this paper presents the related work in Section 2. Section 3 and section 4 explain in detail the proposed chaotic system and its chaotic behavior and bifurcation. In Section 5, we describe the encryption and decryption cryptosystem. Finally, in Section 6, we show the practical results and comparisons with another research.

## II. Related Work

The encryption algorithms based on the chaos theory have been arisen as a power approach to increase the security over the last few decades. The vast majority image encryption algorithms have been introduced based on one- and two-dimensional chaotic maps. In 2017, Pak and Huang [3] described a new chaotic system created by composing the output of two existing one-dimensional chaotic maps. Their algorithm achieved a total shuffling based on a linear-nonlinear-linear structure. In 2007, Chong Fu et al. [4] used a three-dimensional Lorenz chaotic system to increase security and performance of image cryptography. In 2008, Xiangdong et al. [5] presented a chaotic shuffling algorithm using a sorting transformation of a chaotic sequence to obtain address codes for image transposition. Their algorithm avoided the drawbacks of image scrambling ways like rising of complexity and requiring understanding of probability distribution. In 2009, Juan et al. [6] described a three-dimensional image cryptography approach based on a discrete chaotic system with a security key induced from the initial conditions and parameters of the logistic system. In 2011 [7], researchers introduced an image encryption scheme using a Lorenz and Rosslere chaotic system to obtain a large key space, improving security and complexity. In 2005, Zhang et al. [8] presented an algorithm based on a chaotic map and big size encryption key to encrypt the given image followed by a pixel shuffling process using an induced chaotic permutation matrix. In 2011, Keshari and Modani [9] presented an image cryptosystem based on two random maps; a chaotic map lattice to change pixel values by iterating the chaotic map for given initial conditions and Arnold's cat map to rearrange the pixel's position. The same year, Zhang, and Liu [10] introduced a novel image encryption algorithm using a skew tent chaotic system. In their work, they shuffle the order of the positions of all the pixels in the image. Their algorithm was based on permutation-diffusion architecture. In 2012, Khade and Narnaware [6] presented another method to encrypt a color image depending on 3D Logistic and Chebyshev maps. The proposed chaotic maps substituted the RGB components, generated a key, and scrambled the image pixels. The used technique depending on a logistic map used to depict a grey-scale image. A digital matrix approach is used where the three-dimensional matrix value is replaced according to the generated chaotic sequences, whereby the pixel replacement and mixing are achieved at the same time.

Color images are rich with information and have attracted wide attention. Each pixel contains numerical values of RGB components that determine density of RGB components in the color image. Many encryption algorithms have been described [11, 12, 13, 14, 15, 16, 17]. These algorithms are more vulnerable to attack because they neglect the correlations between RGB components. In 2012, Wang et al. [18] proposed a new algorithm using a three-dimensional matrix of the color image and a two-dimensional Lorenz and tent chaotic system at the same time to encrypt RGB components. Their algorithm has four phases. First, the three-dimensional matrix is converted to a two-dimensional matrix and the low-frequency wavelet coefficient is divided into overlapping blocks. Then, encryption is achieved by scrambling the pixel value diffusion based on a completely random chaotic sequence. In 2016, Younes [19] published a useful survey of different techniques of image encryption, discussing several image encryption techniques from 2013 to 2015. In 2021, El Shafai et al. [20] introduced an encryption method to encrypt medical images depending on a piecewise linear chaotic map, and DNA encoding techniques. In 2018, Wu and Yang [21] introduced an algorithm depending on pixel diffusion and a DNA approach, which exploited the two-dimensional Hénon-Sine map to create a pixel permutation. In 2019, Wu et al. [22] used a nonlinear operation in cylindrical diffraction domain and compressed sensing to encrypt a multi-image based on an asymmetric approach. In 2013, Song et al. [23] defined a neighborhood nonlinear map and Coupled Map Lattices (CML) to describe a novel framework based on the Nonlinear Chaotic Algorithm (NCA) chaos and its spatiotemporal merits. In 2020, Yasser, et al. [13], induced an image encryption algorithm using a combination between Discrete Wavelet Transform (DWT) and a chaotic system to shuffle pixels and substitution operations.

## III. Proposed Chaos System

The proposed chaotic system consists of new 1D, 2D, and 3D equations derived from logistic map and cubic maps. The proposed system is a bijection and 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 it provides more security to the system. Six chaotic key maps will be derived within a limited range in (0,1).

$$\left\{\begin{array}{c} \mathbf{1}D\ \mathbf{chaotic\ system} \\ x_{n+1} = \mathrm{r} * x_n * (1 - x_n)(1 - \sin(x_n)) \\ \mathbf{2}D\ \mathbf{chaotic\ system} \\ x_{n+1} = = \mathrm{r}' * x_n * (1 - x_n)(1 - \sin(x_n)) + \gamma_1 y_n^2 \\ y_{n+1} = \mu\, y_n(y_n + 2)(1 - \sin(x_n)) + \gamma_2(x_n y_n + x_n^2) \\ \mathbf{3}D\ \mathbf{chaotic\ system} \\ x_{n+1} = \lambda x_n(2 + x_n)(1 - \sin(x_n)) + \beta y_n^2 x_n + \alpha z_n^3 \\ y_{n+1} = \lambda y_n(2 + y_n)(1 - \sin(y_n)) + \beta z_n^2 y_n + \alpha z_n^3 \\ z_{n+1} = \lambda z_n(2 + z_n)(1 - \sin(z_n)) + \beta x_n^2 z_n + \alpha y_n^3 \end{array}\right\} \quad (1)$$

The control parameters beyond the range have no chaotic behavior. Fig. 1 shows that the 1D and 2D systems enter a chaotic condition and outputs a chaotic series in the region (0,1) subject to: $5.77 < r', r < 12.97$, 2.68< μ < 3.6, 0.039 <$\gamma_1$, $\gamma_2$< 0.25. Fig. 2 shows that, the 3D system enters a chaotic condition in the region (0,1) subject to: 3.49 < $\lambda$ < 3.83 and 0.039 < $\beta, \alpha$ < 0.043. The Bifurcation diagram of 1D, 2D, and 3D are shown in Fig. 3 and Fig. 4, respectively.

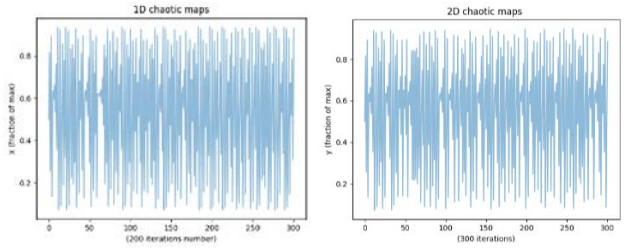



Fig. 1. The Behavior of the (1,2)-D chaotic Map in First 300 Iteration at *r* = 6.27 in *x-y* Plane.

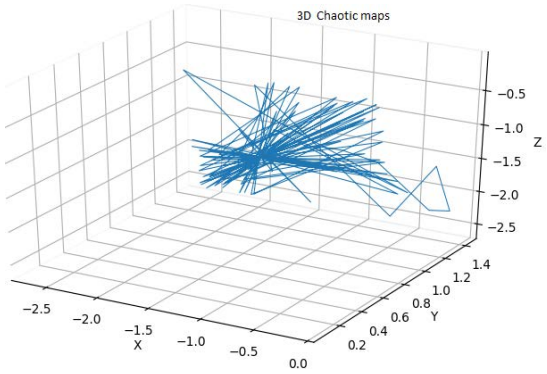



Fig. 2. The Behavior of the 3D Chaotic Map in First 1000 Iteration at λ = 3.81, β = 0.41, and α = 0.046 in x-y-z- Space.

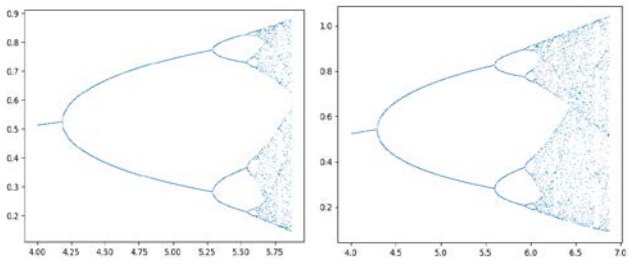


Fig. 3. Bifurcation Diagram of Sequences in 1D (2D) Chaotic System.

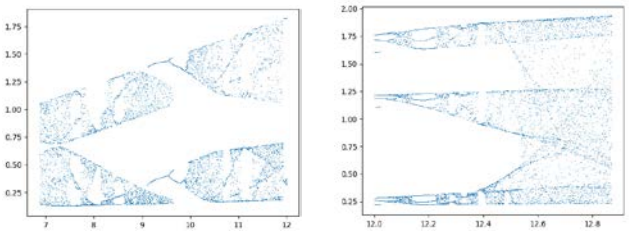


Fig. 4. Bifurcation Diagram of Sequences in 3D Chaotic System.

## IV. Diagonalizing of the Proposed Chaotic Map

For a given prime number, n, the proposed system in (1) can be represented as the set $T$ of bijections as follows:

$$T = \left(T_1(x), T_2(x), T_3(x), T_2(y), T_3(y), T_3(z)\right), \quad (2)$$

where each bijection $T_i(..) \in T$ can be defined as:

$$T_i \begin{bmatrix} x_{n+1} \\ y_{n+1} \\ z_{n+1} \end{bmatrix} = \left( \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \right) \times \begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix} mod\ n \quad (3)$$

$$T_i \begin{bmatrix} x_{n+1} \\ y_{n+1} \\ z_{n+1} \end{bmatrix} = A \times \begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix} mod\ n. \quad (4)$$

If the determinant |A| is not equal to zero and $gcd(|A|, n) = 1$ is satisfied, then the matrix A is invertible. The inverse of system (1) is: $x_{n+1} = (A^{-1} \times x_n) mod\ n$ . Diagonalizing equations in (1) means finding a new $x', y', z'$ system with no cross terms. Based on the Principal Axes theorem every quadratic form can be diagonalized. The diagonalization of the quadratic form 2D is $(x'_{n+1})^T\ Dx'_{n+1}$, $(y'_{n+1})^T\ Qy'_{n+1}$, where $D$ and $Q$ are $2 \times 2$ matrices within four parameters for each. On the other side, diagonalization of the 3D form of the three variables $x, y,$ and $z$ increases the number of parameters up to nine [12].

## V. Proposed Scheme

### A. Chaotic Key Generation

A color image $P$ decomposes to three components $RBG$: red, blue, and gray with $M \times N$ matrices each component with $M$ rows and $N$ columns of pixels. The proposed algorithm uses a novel scheme called Pixel Transform Table (PTT) to generate three $M \times N$ matrix keys $(M_r, M_b, M_g)$, and three vector keys $(V_x, V_y, V_z)$ of length $MN$. The chaotic maps $T_1(x)$, $T_2(x)$, $and\ T_2(y)$ generate three sequences of real numbers, they are converted to $(M_r, M_b, M_g)$, and the chaotic maps $T_3(x)$, $T_3(y)$, and $T_3(z)$ generate three other sequences which are transformed into $(V_x, V_y, V_z)$. The Pixel Transform Table (PTT) approach is shown below.

**PTT ($\rho \geq MN$, jD map)**

Input: The chaotic system (1)
Output: Random map $T_j$

1. Initialize the chaotic parameters in the maps (1).
2. Iterate the system (1) to generate the sequences $S_j(i)$ using jD map, where $j = 1 \rightarrow 6$, and $i = 1,2,..,\rho$.
3. $S_j(i) = \{\mathcal{S}_1(\mathrm{i}), \mathcal{S}_2(\mathrm{i}), .., \mathcal{S}_6(\ i)\}$.
4. For each $j$, generate the set of integer sequences $S_j^I(\rho)$ from the set sequences $S_j(i)$ as:
$$S_j^I(\rho) = \left\lfloor \left(S_j(i) \mathrm{x}\ 10^{14}\right) \right\rfloor mod\ \rho$$
5. For each $j = 1 \rightarrow 6$
   $S^j(\rho) = \mathrm{Sort}\left(S_j^I(\rho)\ \right)$
   find out the position of values $S^j(\ )$ in $S_j^I(\ )$, then construct set of transfer $T^j = \{t_1(i), t_2(i), .., t_6(i)\}$,
   where the value $S_j^I\left(t_{\mathrm{j}}(i)\right) = S^j[i\ ]$, $i = 1,2,..,\rho$.
6. The sequences $(t_1(i),\ t_2(i),\ t_3(i))$ are moved to three $M \times N$ matrix keys $(M_r,\ M_b,\ M_g)$, and the sequences $(t_4(i),\ t_5(i),\ t_6(i))$ are converted to the three vector keys $(V_x, V_y, V_z)$ of length $MN$, respectively.

### B. Image Encryption Algorithm (IEA)

The encryption is an operator that transforms the plain image $P$ to an unknown cipher image $P_5$. IEA algorithm involves three phases as follows:

$$IEA: Plain\ Image\ (P) \longrightarrow Cipher\ Image\ P_5$$

**Phase One**: We describe a new and double secure block shuffling mechanism together with pixels values change. The primary effect of PPT in this process is initially changing randomly the pixel values of $RBG$ called 1st diffusion level as follows:

$$R^r = M_r\ R\ (M_r)^T\ mod\ 256$$
$$B^b = M_b\ B\ (M_b)^T mod\ 256$$

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

where $R^r$, $B^b$, and $G^g$ are the producing matrices that contains the chaotic values. In each matrix $(R^r, B^b, G^g)$, we swap randomly the rows with an odd index $(2k+1)$ with the rows that have an even index $(2k)$, and randomly interchange the columns with an odd index $(2l+1)$ with the columns of an even index $(2l)$, as shown in Fig. 5. We repeat the swap process $\omega$-times and get $(R^r(\omega), B^b(\omega), G^g(\omega))$.

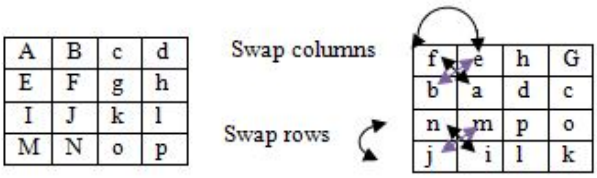



Fig. 5. One-Time Swapping Rows and Columns of the Matrix $(R^r, B^b, G^g)$.

Phase Two: In this phase, the pixel positions were scrambled in columns and rows by using two vector keys $(V_x, V_y)$. We apply 2nd diffusion level by dividing the components $R^r(\omega)$, $B^b(\omega)$, and $G^g(\omega)$ values by their size $MN$ to result the pixels values, i.e.

$$R'(\omega) = \left\lfloor \frac{B^r(\omega)}{MN} \right\rfloor \bmod 256$$

$$B'(\omega) = \left\lfloor \frac{B^b(\omega)}{MN} \right\rfloor \bmod 256$$

$$G'(\omega) = \left\lfloor \frac{G^g(\omega)}{MN} \right\rfloor mod\ 256$$

Combine the matrices $R'(\omega)$, $B'(\omega)$, and $G'(\omega)$ horizontally to obtain the $M \times 3N$ matrix $P_1$, and generate a sequence of $3MN$ numbers $Y_1 = y_1, \ldots, y_{3NM}$ from the rows of $P_1$. Permute the row $Y_1 = y_1, \ldots, y_{3NM}$ by the vector key $V_x$. We get the scramble vector $Y_1' = y'_1, \ldots, y'_{3NM}$. Reshape the vector $Y_1'$ into three $M \times N$ matrices; $Ry(\omega)$, $By(\omega)$, and $Gy(\omega)$. This process is repeated again as follows: Combine the matrices $Ry(\omega)$, $By(\omega)$, and $Gy(\omega)$ vertically to obtain the $3M \times N$ matrix $P_2$, and generate a sequence of $3MN$ numbers $Z_1 = z_1, \ldots, z_{3NM}$ from the columns of $P_2$. Permute the sequence $Z_1 = z_1, \ldots, z_{3NM}$ by the vector key $V_y$. We get the scramble vector $Z_1' = z_1, .., z_{3NM}$. Reshape the vector $Z_1'$ into three $M \times N$ matrices; $Rz(\omega)$, $Bz(\omega)$, and $Gz(\omega)$. Combine these matrices horizontally, we get a new $M \times 3N$ denoted $P_3$.

Phase Three: Let $D_{now}$ be the current ciphered pixel value after the current diffusion, $P_{now}$ the present plain pixel value, $D_{pre}$ the old cipher pixel value after the previous diffusion, and $P_{pre}$ the old plain value. Their initial values in $Rz(\omega)$, $Bz(\omega)$, and $Gz(\omega)$ are equal to zero. Keep track of rows in $P_3$ to set three vectors $V(Rz(\omega))$, $V(Bz(\omega))$, and $V(Gz(\omega))$ each with length $MN$. We apply 3rd level of pixel diffusion on these vectors using the vector product with the key vector $V_z$ as in the following:

$$\begin{cases} Zz^r = \big(V(Rz(\omega)) \times V_g\big) mod\ 256 \\ Zz^b = \big(V(Bz(\omega)) \times V_g\big) mod\ 256 \\ Zz^g = \big(V(Gz(\omega)) \times V_g\big) mod\ 256 \end{cases} \quad (5)$$

where $Zz^r$, $Zz^b$, and $Zz^g$ are the producing vectors that contains the chaotic values of $V_z$. We apply 4th and final level of pixel diffusion as shown in the next procedure where [(:)] refers to the components $Zz^r$, $Zz^b$, and $Zz^g$ while [:] refers to a random value.

**4th random Diffusion** ([(:)], [:])

Input: $V_z = (Zz^r, Zz^b, Zz^g)$

Output: Vector $(D_{now}(Zz^r), D_{now}(Zz^b), D_{now}(Zz^g))$

1. Generate three random values follows:
   $\varepsilon_{1l}$ = rand () % 3
   $\varepsilon_{2l}$ =choice a random value from $\{V_z\}$mod 256
   $\varepsilon_{3l}$ =choice a random value from $\{V_z\}$mod 256
2. For $\omega$ in range $MN$ do
3. If $\varepsilon_{1l}$= 0 then
   $D_{now}(Zz^r) = \big(\varepsilon_{2l}.P_{now}(Zz^r) + \varepsilon_{3l}.(D_{pre}Zz^r \times P_{pre}Zz^r)\big)$mo 256. (6)
4. Else If $\varepsilon_{1l}$=1
   $D_{now}(Zz^b) = \big(\varepsilon_{2l}.P_{now}\left(Zz^b\right) + \varepsilon_{3l}.(D_{pre}Zz^b \times P_{pre}Zz^b)\big)$mo 256. (7)
5. Else If $\varepsilon_{1l}$=2
   $D_{now}(Zz^g) = \big(\varepsilon_{2l}.P_{now}(Zz^g) + \varepsilon_{3l}.(D_{pre}Zz^g \times P_{pre}Zz^g)\big)$mo 256. (8)
6. End For.

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

The flowchart of the encryption process with the induced 6 keys is shown in Fig. 6. The flowchart of the decryption algorithm and the 6 keys is shown in Fig. 7.

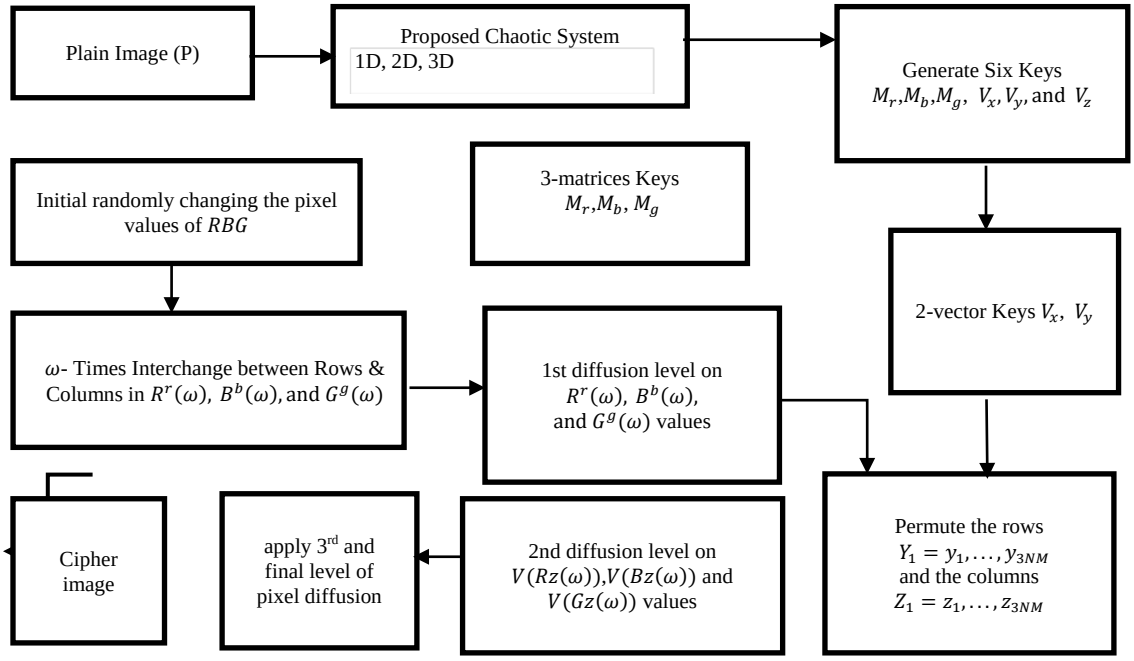



Fig. 6. The Outline of Image Encryption Algorithm IEA.

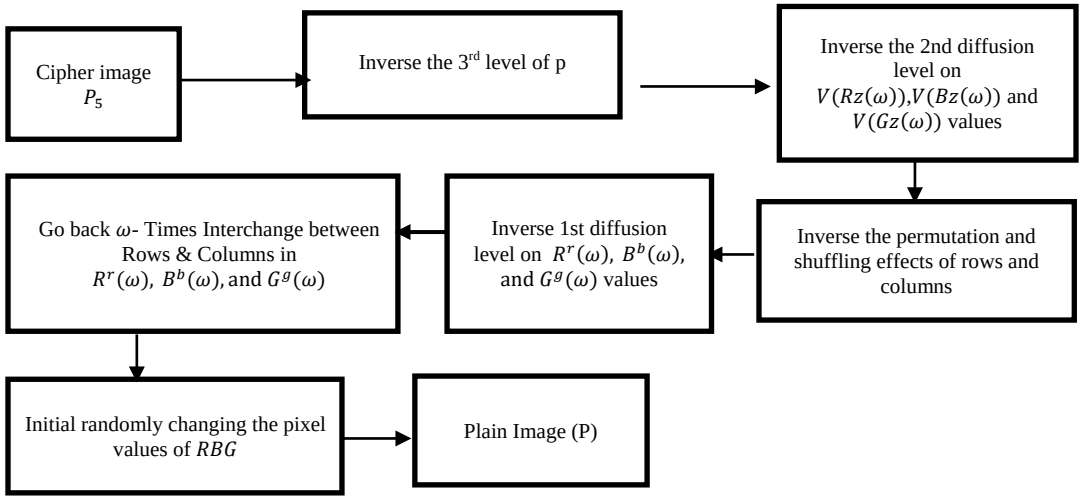



Fig. 7. The Outline of Image Decryption Algorithm IDA.

### *C. Image Decryption Algorithm (IDA)*

The image decryption algorithm (IDA) is like that of the image encryption pseudocode but in the inverse direction.

Step 1: All random values used in IEA are used in IDA. Step 2: Inverse 4th random Diffusion ([(:)] , [:] ). Obtain $Zz^r$, $Zz^b$, and $Zz^g$ from $D_{now}(Zz^r)$, $D_{now}(Zz^b)$, and $D_{now}(Zz^g)$. The inverse of the equations (6), (7), and (8) are the equations (9), (10), and (11) respectively.

$$Zz^r = \left(\frac{D_{now}(Zz^r) - \varepsilon_{3l}(D_{pre}Zz^r \times P_{pre}Zz^r)}{\varepsilon_{2l}}\right) \bmod 256 \quad (9)$$

$$Zz^b = \left(\frac{D_{now}(Zz^b) - \varepsilon_{3l}(D_{pre}Zz^b \times P_{pre}Zz^b)}{\varepsilon_{2l}}\right) \bmod 256 \quad (10)$$

$$Zz^g = \left(\frac{D_{now}(Zz^g) - \varepsilon_{3l}(D_{pre}Zz^g \times P_{pre}Zz^g)}{\varepsilon_{2l}}\right) \bmod 256 \quad (11)$$

Step 3: Inverse 3rd level of pixel diffusion. Obtain $V(Rz(\omega))$, $V(Bz(\omega))$ and $V(Gz(\omega))$ from $Zz^r$, $Zz^b$, and $Zz^g$, respectively. The inverse of the equations in (5) are the equations in (12).

$$\begin{cases} V(Rz(\omega)) = \left(\frac{Zz^r \times V_g}{V_g.V_g}\right) \bmod 256 + \tau V_g \\ V(Bz(\omega)) = \left(\frac{Zz^b \times V_g}{V_g.V_g}\right) \bmod 256 + \tau V_g \\ V(Gz(\omega)) = \left(\frac{Zz^g \times V_g}{V_g.V_g}\right) \bmod 256 + \tau V_g \end{cases} \quad (12)$$

Step 4: Delete the effect of rows and columns scramble by using the inverse vectors $(V_x^{-1}, V_y^{-1})$.

Step 5 : Inverse 2nd random Diffusion as follows:

$$B^r(\omega) = \lfloor MN.R'(\omega) \rfloor \bmod 256$$

$$B^b(\omega) = \lfloor MN.B'(\omega) \rfloor \bmod 256 \quad (13)$$

$$G^g(\omega) = \lfloor MN.G'(\omega) \rfloor \, mod \, 256$$

Step 6: Come back to the reverse paths to delete the effect of randomly interchange the columns rows.

Step 7: Inverse 1st random Diffusion as follows:

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

$$B = (M_b)^{-1} B^b((M_b)^T)^{-1} \bmod 256 \quad (14)$$

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

These seven steps recover the plain image *P*. Mathematically, the image decryption algorithm is represented as in the next formula:

*IDA: Cipher Image* $P_5 \longrightarrow$ *Plain Image (P)*

*D. The Computational Complicity*

In this section, we discuss the running time of the steps in IEA. The running time of the PPT subroutine is $O\ (\boldsymbol{\rho = MN} + \epsilon)$ for small constant $\epsilon$, and the running time of the steps 1 -7 is a linear function of the size of the image, i.e., the running time is $O(\boldsymbol{MN})$. The experimental process demonstrates that the consumption time of our approach IEA/IDA is practicable. The processor Intel(R) Core (TM) i7-8550U CPU @ 1.80 GHz 2.00 GHz and 8GB RAM is used in IEA/IDA is practicable. The processor Intel(R) Core (TM) i7-8550U CPU @ 1.80 GHz 2.00 GHz and 8GB RAM is used in encryption/decryption on 256 image $P/P_5$ of size $256 \times 256$. The average consumption time is less than 4.01ms. Our running time is close to the running time in [24].

## VI. Experimental Results and Histogram Analysis

The system is implemented by anaconda 4.8.3. Python is the programming language that we have picked for development. The implementation is examined in Windows 10 64-bit O.S. with an Intel(R) Core (TM) i7-8550U CPU @ 1.80GHz 2.00 GHz and 8GB RAM. Take the initial parameters and values: 1D (r = 5.78, $x_0 = 1.2\times 10^{18}$), 2D ( r′ = 6.37, μ =3.33, $\gamma_1 = 0.17$, $\gamma_2 = 0.14$, $x_0 = 2.3 \times 10^{18}$, $y_0 = 1.2 \times 10^{18}$ ), 3D ( $\lambda$ = 3.66, $\beta$ = 0.041, $\alpha$ = 0.039, $x_0 = 3.4\times 10^{18}$, $y_0 = 4.5\times 10^{18}$, $z_0$ =5.6× $10^{18}$), to encrypt "Baboon" and "Lena" images of size $256 \times 256$ and their RGB components as shown in Fig. 8(a-d). The histograms of the "baboon", "Lena" and their RGB components before encryption are shown in Fig. 9(a–c); the histograms illustrate how pixels in the plain images "baboon" or "Lena" are correlated to the pixels at each color density level.



Fig. 8. (a-d) shows "Baboon" and "Lena" Color Images and the RGB Components of their Respective Color Images before the Encryption Process.

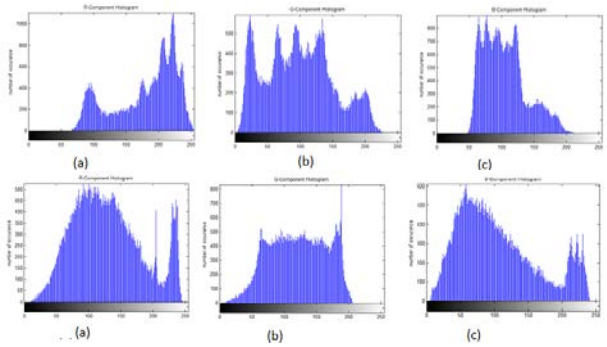



Fig. 9. Histograms of the RGB Components of "Baboon" and "Lena" before Encryption respectively.

Fig. 10 (a–d) shows the "baboon" and "Lena" encrypted images and their respective encrypted RGB components. The histograms of the RGB components of "baboon", and "Lena" after the encryption are shown below in Fig. 11 (a–c). The histograms illustrate how pixels in the ciphered RGB are uniformly correlated to the pixels at each color density level.

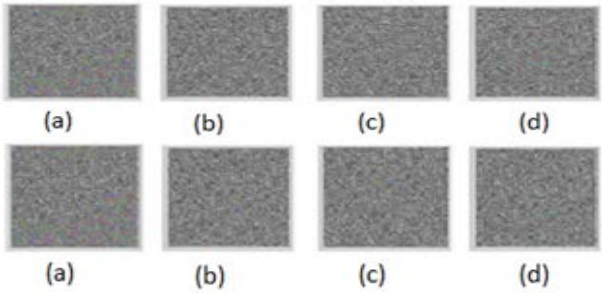



Fig. 10. (a-d) Shows "Baboon" and "Lena" Cipher images and their Ciphered RGB Components.

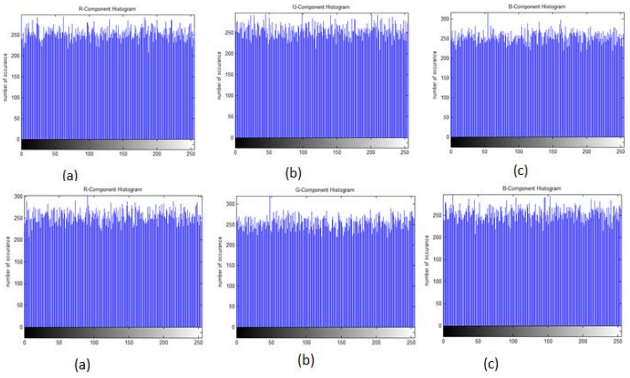



Fig. 11. (a–c): Histograms of RGB Component of "Baboon" and "Lena" after Encryption.

## VII. Security Analysis

### A. The Key Size

The total number of unique keys used in the encryption must be big enough to make brute-force attacks ineffective. Our algorithm employs six values; $x_0^{1D}, x_0^{2D}, y_0^{2D}, x_0^{3D}, y_0^{3D}$, $z_0^{3D}$ and the eight parameters of $\mu, r, r', \gamma_1, \gamma_2, \lambda, \beta, \alpha$, as secret keys. Wang and Teng [25] proved that if the precision is $10^{-17}$, the keys $K_{x_0^{1D}} = K_{x_0^{2D}} = K_{y_0^{2D}} = K_{x_0^{3D}} = K_{y_0^{3D}} = K_{z_0^{3D}} = 10^{17}$, $K_\mu = K_r = K_{r'}$, $K_{\gamma_1} = K_{\gamma_2} = K_\lambda = K_\beta = K_\alpha = 0.5 \times 10^{17}$. Let the plain color 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 total key space reaches to $\approx 1.953 \times 10^7 \times 10^{235} = 1.953 \times 10^{242}$. Our key space is larger than $2^{138}$, $2^{58}$, $10^{140}$, $2^{256}$, $10^{79}$, $4.2\times 10^{122}, 10^{60}$, and $10^{112}$ [4, 13, 6, 9, 18, 26, 27, 28] respectively. It is greater than $2^{448} = 7.8\times10^{134}$, the maximum key space mentioned in the survey paper [14]. The total numbers of keys within the diagonalization form (1) reach 24 initial values and parameters. Our key increases it up to $10^{415}$. The proposed algorithms describe a sufficiently enough key space to withstand brute-force assaults.

### B. The Sensitivity Analysis of the Secret Keys

Little differences between keys yield different cipher images. When the image is decrypted, using a wrong key induces another image. Fig. 12 displays the decrypted image of the Baboon with the proper key of λ=3.66. On the other side, Fig. 13 illustrates the decryption of the Baboon image with the incorrect encryption key λ=3.66000000000000001. It was successful in making the algorithm sensitive to the key. A small modification in the key will result in an entirely different decryption result, and the attacker won't be able to get to the right plain image.


Fig. 12. Result of Correct Parameters used to Decrypt the Baboon Image and its R, G and B Components.

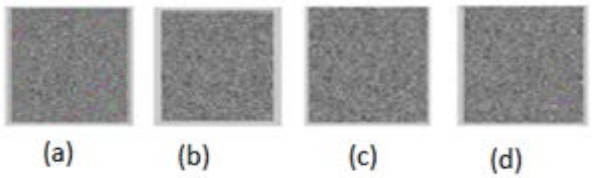



Fig. 13. Result of Wrong Parameters used to Decrypt the Baboon Image and its R, G and B Components.

### C. Adjacent Pixels Correlation Analysis

The correlation between pixels is assessed by the degree of pixel association. In general, the stronger the correlation between nearby pixels in the ciphered image, the poorer the encryption algorithm's performance will be, and vice versa. The correlation in vertical, horizontal, and diagonal directions between 3000 randomly selected nearby pixels is calculated as follows:

$$E(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i) \quad (15)$$

$$E(y) = \frac{1}{N}\sum_{i=1}^{N}(y_i) \quad (16)$$

$$D(x) = \sum_{i=1}^{N}(x_i - E(x))^2 \quad (17)$$

$$D(y) = \sum_{i=1}^{N}(y_i - E(y))^2 \quad (18)$$

$$cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}\big(x_i - E(x)\big)\big(y_i - E(y)\big) \quad (19)$$

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \quad (20)$$

The values of two neighboring pixels, $x$ and $y$, in the Horizontal direction H/D, Vertical Direction V/D and Diagonal direction D/D are shown in Tables I to III and Fig. 14. They exhibit a high level of concentration and a tendency of one in two neighboring pixels in the plain image and a correlation extremely near to 0 in the ciphered image, which means that adjacent pixels in the ciphered image are random, and the encryption impacts resist statistical attack.

TABLE I. Two Adjacent Pixel Correlation Values for Plain and Ciphered Images (V/D)

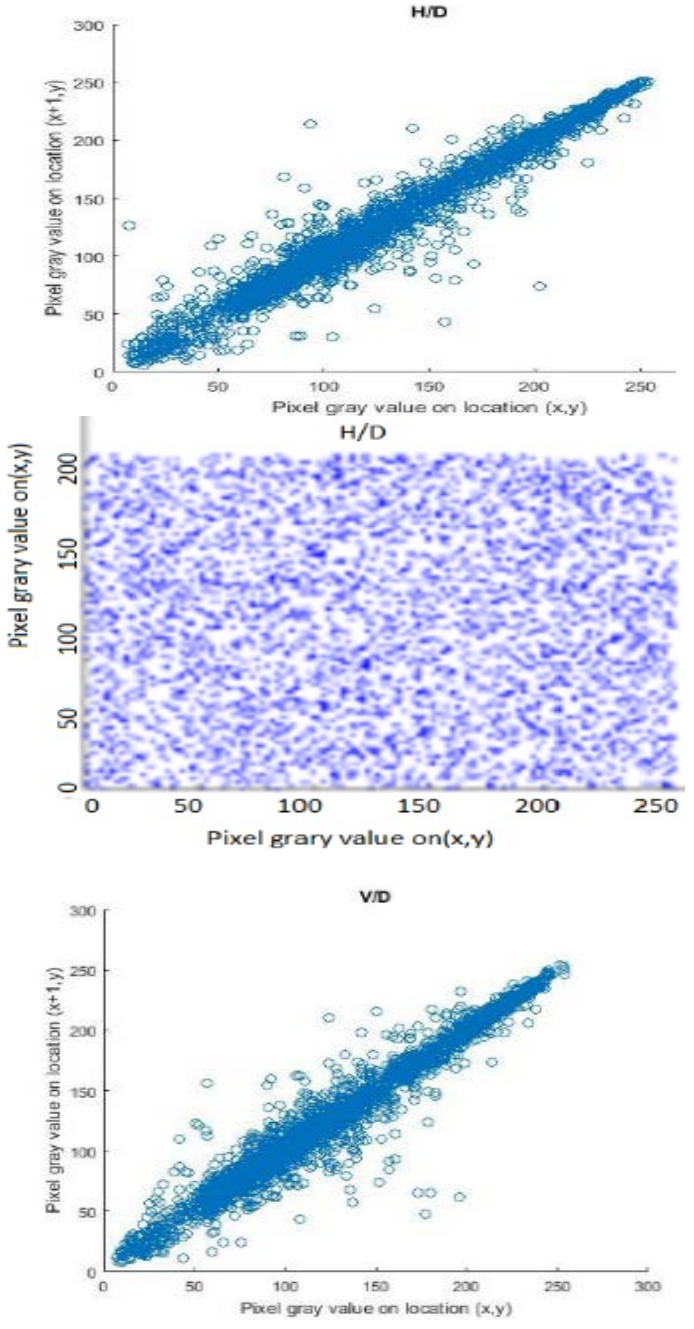
| Image | *V/D* | Plain image | Cipher Image | |
|---|---|---|---|---|
| | | | *New Algorithm* | *Ref [28]* |
| Lena | *R* | 0.9238 | -0.0022 | -0.0016 |
| | *G* | 0.9479 | 0.0026 | -0.0011 |
| | *B* | 0.8785 | -0.00103 | -0.0013 |
| Baboon | *R* | 0.9527 | -0.0021 | 0.0002 |
| | *G* | 0.9283 | -0.0047 | 0.0001 |
| | *B* | 0.9563 | 0.00201 | 0.0004 |

TABLE II. Two Adjacent Pixel Correlation Values for Plain and Ciphered Images (H/D)

| Image | *H/D* | Plain image | Cipher Image | |
|---|---|---|---|---|
| | | | *New Algorithm* | *Ref [27]* |
| Lena | *R* | 0.9783 | -0.0017 | -0.00092 |
| | *G* | 0.9795 | 0.0034 | -0.0038 |
| | *B* | 0.9594 | -0.00063 | -0.0020 |
| Baboon | *R* | 0.9413 | -0.0019 | 0.0062 |
| | *G* | 0.8796 | -0.0056 | -0.0060 |
| | *B* | 0.9164 | 0.0018 | 0.0077 |

TABLE III. TWO ADJACENT PIXEL CORRELATION VALUES FOR PLAIN AND CIPHERED IMAGES (D/D)

| Image | Component *D/D* | Plain image | Cipher Image | |
|---|---|---|---|---|
| | | | *New Algorithm* | *Ref [10]* |
| Lena | *R* | 0.9685 | -0.0011 | -0.0008482 |
| | *G* | 0.9574 | 0.0021 | -0.0008482 |
| | *B* | 0.8994 | -0.00033 | -0.0008482 |
| Baboon | *R* | 0.6471 | -0.0021 | 0.00370914 |
| | *G* | 0.9567 | -0.0036 | 0.00370914 |
| | *B* | 0.9355 | 0.0015 | 0.00370914 |



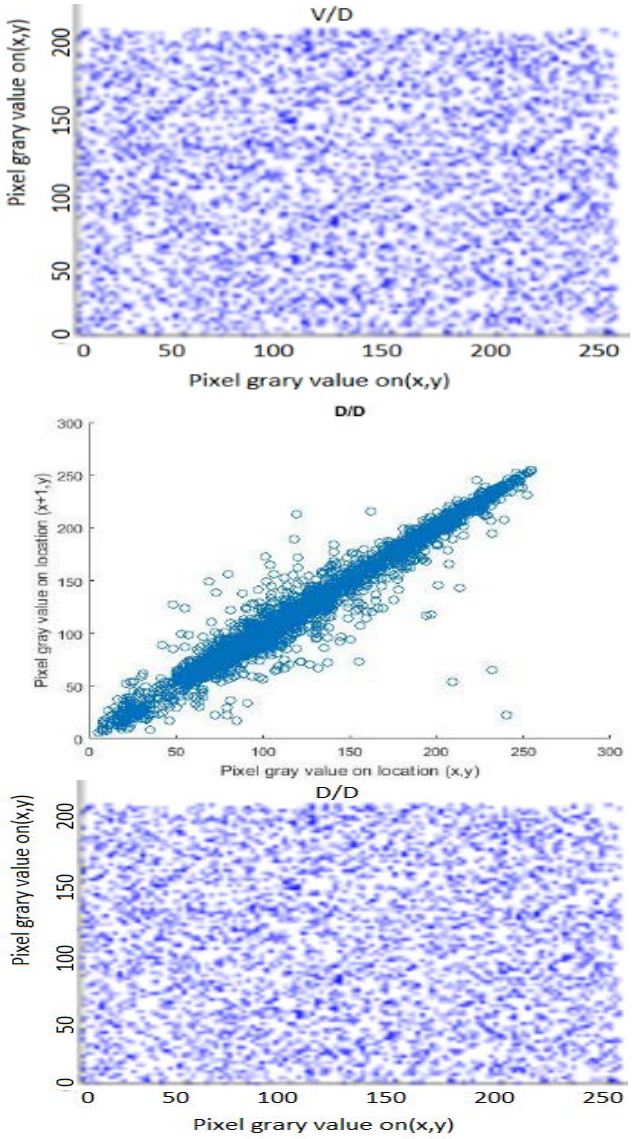




Fig. 14. Correlation Test of the RGB Components of the Plain Image and Ciphered Image. The Figure Displays the Distribution Coefficient of nearby Pixels in Three Directions (H/D, V/D, and D/D) for the Plain Image (up) and the Ciphered Image (down).

## D. Discovery of Differential Attack

To evaluate the capacity to withstand a differential attack, the Pixel Change Rate Number, and the Unified Average Change Intensity (UACI) tests are applied. The NPCR test measures the number of different pixels between plain and encrypted images, and the UACI test measures the average intensity of these two images [29]. A gentle pixel change generates a softly modified cipher image. Analyzing the relationship between the cipher image created and the plain image using NPCR and UACI becomes essential. The definition of UACI and NPCR is as follows:

$$UACI_{R,G,B} = \left( \frac{\frac{\sum_{ij} \left| c_{R,G,B}(i,j) - c'_{R,G,B}(i,j) \right|}{255}}{M \times N} \right) \times 100 \quad (21)$$

$$NPCR_{R,G,B} = \left(\frac{\sum_{ij} D_{R,G,B}(i,j)}{M \times N}\right) \times 100 \quad (22)$$

$$D_{R,G,B}(i,j) = \begin{cases} 0 & if\ C_{R,G,B}(i,j) = C'_{R,G,B}(i,j) \\ 1 & Otherwise \end{cases} \quad (23)$$

Respectively, the width and height of the image are M and N, and the pixel values in the $i$-th row and $j$-th column are $C_{R,G,B}(i,j)$ and $C'_{R,G,B}(i,j)$ for the two ciphered images before and after one pixel of the original plain image is altered. Table IV shows the values of $NPCR_{R,G,B}$ over 99.55% and values of $UACI_{R,G,B}$ above 33.44%. The studies illustrate that our technique is highly sensitive to minor changes in the original image, even if the two original images are only one-bit different, the decrypted images are somewhat different.

TABLE IV. RESULTS OF NPCR AND UACI (PERCENT)

| *Image* | *The Proposed Algorithm* | | | *Ref [10]* | |
|---|---|---|---|---|---|
| | | *NPCR%* | *UACI %* | *NPCR%* | *UACI %* |
| *Lena* | *R* | *99.5117* | *33.4218* | *99.6052* | *33.4132* |
| | *G* | *99.4751* | *33.4411* | | |
| | *B* | *99.5132* | *33.4887* | | |
| *Baboon* | *R* | *99.5895* | *33.6405* | *99.6227* | *33.4865* |
| | *G* | *99.5728* | *33.3403* | | |
| | *B* | *99.6368* | *33.4706* | | |

### *E. Peak Signal-to-Noise Ratio (PSNR) Analysis*

In image reconstruction, PSNR is used primarily as a quality metric. The following equation is calculated:

$$PSNR_{R,G,B} = 20 * log_{10}\left(\frac{255}{sqrt(MSE_{R,G,B})}\right) \quad (24)$$

$$MSE_{R,G,B} = \sum_i \sum_j \frac{C_{R,G,B}(i,j) - C'_{R,G,B}(i,j)}{MxN} \quad (25)$$

The mean square error (MSE) describes the difference in the values from 0 to 255 between the plain and the ciphered image. The variations between the original and the encrypted image in the PSNR values also are shown in Table V. Our approach shows higher resistance to statistical attacks.

TABLE V. RESULTS OF PSNR

| *Image* | *New algorithm* | | | *Reff [27]* | | |
|---|---|---|---|---|---|---|
| | *R* | *G* | *B* | *R* | *G* | *B* |
| Lena | 7.9687 | 8.8887 | 9.7595 | 7.8992 | 8.5765 | 9.6785 |
| Baboon | 8.3985 | 9.4578 | 8.9701 | 8.9581 | 9.4143 | 8.4156 |

## VIII. CONFLICT OF INTEREST

The authors declare that there is no conflict of interest and there is no competition in the financial interest.

## IX. CONCLUSION

We proposed a large enough key space algorithm to resist brute-force attacks for image security. The proposed cryptosystem generates three keys in the form of a square matrix $M \times N$, and three keys in the form of a vector matrix of length $MN$. Our cryptosystem for image encryption technique is based on a combination of multidimensional chaos systems and the diversity between shuffling and scrambling of rows and columns in the RGB components of the plain image, as well as multi-level diffusion of pixel values. The computational study between the proposed algorithm and other cryptosystem showed that the proposed algorithm has high level of security and wide key spaces. The advantage of the current algorithm is using multi-level encryption based on big key space.

## REFERENCES

[1] Mohammed A. B. Younes, "Literature Survey on Different Techniques of Image Encryption", International Journal of Scientific & Engineering Research, Vol. 7, Issue 1, 2016.

[2] Robert A J Matthews "On the derivation of a chaotic encryption algorithm" J. Cryptologia, Vol. 13, No. 1, 1989, pp. 29-42.

[3] Chanil Pak, Lilian Huang" A new color image encryption using combination of the 1D chaotic map", Signal Processing 138 (2017) 129–137.

[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, pp. 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] Juan Li, Yong Feng, Xuqiang Yang" Discrete Chaotic based 3D Image encryption Scheme", Symposium on Photonics and Optoelectronics, September 2009 IEEE.

[7] Qais H. Alsafasfeh, Aouda A. Arfoa," Image Encryption Based on the General Approach for Multiple Chaotic Systems", Journal of Signal and Information Processing, 2011, 2, 238-244.

[8] H. Zhang, X.F. Wang, Z.H. Li, and D.H. Liu, "A Fast Image Encryption Algorithm Based on Chaos System and Henon Map", Journal of computer Research and Development, Vol. 42, issue 12, 2137-2142, 2005.

[9] Seyed Mohammad Seyedzadeh, Sattar Mirzakuchaki "A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map" Signal Processing 92 (2012) 1202–1215.

[10] Guoji Zhang a, Qing Liu b," A novel image encryption method based on total shuffling scheme", Optics Communications 284 (2011) 2775–2780.

[11] 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.

[12] Howard Anton and Chris Rorres "Elementary linear Algebra, Applications Version", Tenth Edition, WILEY, 2011.

[13] Ibrahim Yasser, Fahmi Khalifa, Mohamed A. Mohamed, and Ahmed S. Samrah" A New Image Encryption Scheme Based on Hybrid Chaotic Maps" J. Complexity Volume 2020, Article ID 9597619, 23 pages.

[14] M. Kumari, S. Gupta, P. Sardana, "A survey of image encryption algorithms", 3D Research, Volume 8, Number 4, (2017) Article No.:148

[15] P. N. Khade and M. Narnaware," 3D Chaotic Functions for Image Encryption", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 3, No 1, May 2012.

[16] Xingyuan Wangn, LinTeng, Xue Qin," A novel colour image encryption algorithm based on chaos", Signal Processing 92 (2012) 1101–1108.

[17] C. Hoppen, Y. Kohayakawa, C. Moreira,B. Ráth, R. Sampaio" Limits of permutation sequences", Journal of Combinatorial Theory, S. B 103 (2013) 93–113.

[18] W. El-Shafai, et al. "Robust medical image encryption based on DNA-chaos cryptosystem for secure telemedicine and healthcare applications", Journal of Ambient Intelligence and Humanized Computing, March 2021.

[19] S. Mazloom and A.M. Eftekhari-Moghadam," Colour image encryption based on coupled nonlinear chaotic map", Chaos, Solitons & Fractals 42 (3) (2009) 1745–1754.

[20] 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.

[21] J. Wu, X. Liao, and B. Yang, "Image encryption using 2D Hénon-Sine map and DNA approach," Signal Processing, 2018, vol. 153, pp.11–23.

[22] C. Wu, Y. Wang, Y. Chen, J. Wang, and Q. Wang, "Asymmetric encryption of multiple-image based on compressed sensing and phase-truncation in cylindrical diffraction domain," Optics Communications, 2019, vol. 431, 203–209.

[23] C.-Y. Song, Y.-L. Qiao, and X.-Z. Zhang, "An image encryption scheme based on new spatiotemporal chaos," Optik-International Journal for Light and Electron Optics, 2013, vol. 124, no. 18, pp. 3329–3334.

[24] C. c. chang, M. Hwang, T. chen." A new encryption algorithm for image cryptosystem", Journal of system and software, vol. 58, 2001, 83-91.

[25] X. Wang and L. Teng," A bit-level image encryption algorithm based on spatiotemporal chaotic system and self-adaptive", Optics Communications Volume 285, Issue 20, 15 September 2012, Pages 4048-4054.

[26] Zhenjun Tang, Ye Yang, Shijie Xu, Chunqiang Yu, and Xianquan Zhang" Image Encryption with Double Spiral Scans and Chaotic Maps" Security and Communication Networks Volume 2019, Article ID 8694678, 15 pages.

[27] 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.

[28] Arslan Shafique, Mohammad Mazyad Hazzazi, Adel R. Alharbi, Iqtadar Hussain, "Integration of Spatial and Frequency Domain Encryption for Digital Images", Access IEEE, vol. 9, pp. 149943-149954, 2021.

[29] Yue Wu, Joseph P. Noonan, and Sos Agaian, "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.

## AUTHORS' PROFILE

**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.

**Nawaz Majid** is an assistant professor of computer science at Department computer science, Faculty of science, Northern Border University (NBU), KSA. His research interests span both theoretical computer science and information security. Adding more to that, he worked on research including Data mining and knowledge discovery.