Received: March 18, 2023. Revised: May 6, 2023.

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

An Efficient Cryptosystem for Image Using 1D and 2D Logistic Chaotic Maps

Ibtisam A. Taqi[1]* **Mela G. Abdul-Haleem[1]**

[1]Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq
* Corresponding author's Email: ibtisam.taqi@sc.uobaghdad.edu.iq

Abstract: Image is very important in multimedia technology and social media such as WhatsApp, Facebook, Viber, and Telegram, and in many fields, including military, bank accounts ,and communications. Therefore, protecting the image from hacking or theft during storage or transmission over the net is the most demanded matter nowadays. This paper proposes a new method for grayscale image encoding based on two types of logistic maps, one-dimension (1D) and two-dimension (2D). First, convert a color image to grayscale and then into a binary form as a confusion step. Second, compute the initial parameters of (1D) and (2D) logistic maps using newly suggested equations. After that, permute the binary image with the new positions generated by a (2D) logistic map as a diffusion step. At the same time, a highly complicated random key is generated using a new suggested equation based on a (1D) logistic map. Then, XOR is applied between the generated key and the final image to spawn the encoded image. Finally, a median filter is applied as a post-processing step for the noisy cipher image to increase the efficiency of the proposed system when retrieving the decrypted image. The proposed method encodes images of size (1024×1024) larger than the other works. A high Mean Square Error (MSE) of 7768.40, a low peak signal to noise ratio (PSNR) of 9.22749, a high number of pixel change rate (NPCR) up to 99.63%, and a high entropy of 7.99985 are obtained. Additionally, a high speed and a high key precision 10^{16} are achieved. The proposed schema outperforms the other conducted works in terms of the previous evaluation measurements. The system retrieves 92% of the images with 50% salt and pepper noise and 84% of the images with 0.1 Gaussian noise, indicating higher security against noise attacks than the other works.

Keywords: Gray image, 256-bit key, Post-processing, Median filter, Noise, Iteration.

1. Introduction

With the explosive development of network technologies and multimedia, the interest in transferring images has become a great demand. Image data has the characteristics of large amounts of data, strong correlations, and high redundancy. Image encryption has become a widespread requirement to protect images during transmission over the Internet and when storing. The traditional encryption methods such as DES and AES cannot support image encryption due to their small key size, low randomness using the same way to encrypt each block, and slowly taking long encryption time which means inefficiency and less security. Many types of chaos maps have been used by researchers as effective methods for protecting digital image transmission due to their advantages such as initial parameters sensitivity, high pseudo-randomness, non-periodicity, the complex structure that are difficult to analyze and meet the cryptographic requirement [1-3]. This paper proposes a new encoding method for grayscale image-based chaos using a 1D and 2D logistic maps combined with the median filter as a digital image post-processing method to increase the performance of the system against noise attacks when retrieving the decrypted image without any loss of data or too low loss of information. The research parts are arranged as follows: Part 2, the other works are clarified. Part 3 explains 1D and 2D logistic maps and median filter processing. In Part 4, the suggested system is explained in detail. Part 5 illustrated the metrics for evaluation of the suggested system efficiency and the

attack analysis. Finally, in Part 6 conclusions are illustrated.

2. The other works

A great deal of literature has been posted on grayscale and color image encoding as follows:

In [4], a 2D logistic chaotic map is used for confusion and diffusion to encrypt an image and to generate a high and complicated random key. The results showed that the image encryption schema has the ability to resist several types of attacks like differential and statistical attacks. The effectiveness and robustness clarify by using the number of pixel change rate (NPCR) and unified averaged changed intensity (UACI) analyses, histogram analysis, the key sensitivity analysis, correlation analysis, and entropy test. In [5], the encryption schema includes three stages, which are obtaining initial encryption by block shuffling, generating a set of secret matrices using the Arnold transform and chaotic map, and the final encryption of each block with exclusive OR operation between the block and a random secret matrix. A secret key control at each stage to provide a large key space. The schema is very fast because of the low complexity of both the block shuffling calculations and the generation of the secret matrix. In [6], a rapid image encryption method with high performance depending on a new 1D chaotic map was presented. The advantage of this map is its easy and simple structure, hardware implementation, and high speed. Both Lyapunov exponent analysis and bifurcation analysis were used for the declaration of chaotic features. The shuffling and substitution schemas are associated with the new map. The mixture of encryption algorithms and chaotic systems is the best for encrypting an image. In [7], an efficient compound system, depending on a Sine-Tent chaotic map was produced for image encryption. This system offers enhanced resistance to differential analysis attacks. In [8] a hybrid chaotic procedure depending on mixing various 1D chaotic maps was presented. Both entropy and Lyapunov exponents' values were used to validate the obtained map's performance. The authors presented an encryption algorithm depending on optimizing S-boxes via "chaotic Jaya optimization" as well as the generating of dynamic keys. The resulting S-boxes were utilized in the permutation stage of the encryption algorithm. The high resistance of the algorithm to different attacks was validated. In [9] the authors suggested a simple structured and fractional 1D chaotic map with a big space to overcome the easy prediction in the case of using low-dimensional chaotic maps and the complicated structure of high-dimensional chaotic maps. The substitution and permutation phases were merged to change the values and positions of the pixels. This process increased the speed, security, and performance of their suggested image encryption algorithm. In [10], a new fast image encoding method depending on the "Bülban map" was obtained. This map was utilized to get rows or columns randomly. Their method included performing the shuffling and substitution at the row level and then at a column level to decrease the pre-processing time, shuffling the position of the pixels using a circular shift, and finally masking the values of the pixels by performing a set of XOR and modulo operations on the bit level. The applied security analysis proved that the method is suitable for real-time image encryption. In [11], the authors used a 2D piecewise smooth nonlinear chaotic map (2DPSNCM) to produce chaotic sequences for the encryption of digital images. Their method includes the use of the logistic map to shuffle the plain image as a confusing process, producing chaotic sequences using 2DPSNCM, converting these sequences into integers, and masking the shuffled image depending on the sequences as a diffusing process. Additionally, the plain image is retrieved via the secret key. Many analyses related to security and performance have been conducted which proved that this algorithm is resistant to different attacks, safe, and quick. In [12] the authors presented a trigonometric map for symmetrical image encryption. In a comparison with the existing sine map, a bigger parameter interval and higher chaotic performance are shown in the trigonometric map. SHA-256 is used to generate a secret key from the plaintext image for resisting the chosen-plaintext attack. In [13], the authors proposed a robust image encryption method where a complicated network with a multi-stable hyper-chaos was used to generate encryption sequences used to shuffle the values of the image's pixels. A time-delayed neural network was used to generate the chaotic sequences used to scramble the positions of diffused images. The synchronized use of both networks helps in the enlargement of the key space. In [14], the authors enhanced a previously proposed cascaded chaotic map and Zigzag transform for producing a secure image encryption scheme. The aim of the enhancement in the cascaded chaotic map was to get an equally distributed chaotic sequence. Both the SHA256 hash value and the plain image were used to get the initial values and parameters of the chaotic map. The purpose of the enhancement in the Zigzag transform was to increase the randomness in the cipher image by completely scrambling the pixels of the plain image and combining them with the enhanced map. For diffusion, the authors

designed a wave-shaped diffusion algorithm that alternates different rows and columns. In [15], the authors presented a new method for image encryption depending on double chaotic systems and discrete wavelet transform (DWT). The latter is used for decomposing the image into sub-bands. The parameters and states of the presented chaotic maps are used to gain the properties of permutation and diffusion, which are then responsible for pixel shuffling and subsequent replacement operations. Security and statistical analyses showed the capability of the presented method to resist many common attacks. In [16], the parallel DNA coding technique is utilized to design a fast method for image encryption as well as a 1D fractional chaotic map with a wide chaotic range is presented by combining a fraction operation with a sine map. Using parallel computing leads to speeding up the encryption and decryption methods. The SHA-3 hash value of the original image is used to generate the initial key of the cryptosystem, which increases the resistance against a chosen-plaintext attack. The analysis results showed good performance of the proposed encryption method and high resistance to noise and data loss attacks.

This paper aims to build a more secure system than the other works by encrypting higher image sizes up to 755×755, resisting a higher ratio of noise attacks up to 20% salt and pepper noise, and up to 0.01 Gaussian noise.

3. Chaotic maps and median filter

This section presents the basic chaotic maps used in the proposed image encryption and median filter as a post-processing technique. 1D and 2D logistic chaotic maps are utilized in this paper.

3.1 1D logistic map

A 1D and 2D logistic chaotic maps are mapping of polynomials (recurrence relation) and are quoted to illustrate that complex chaotic behaviour from nonlinear dynamical equations. 1D logistic describe in Eq. (1) [1, 6]:

$$x_{n+1} = r x_n (1 - x_n) \tag{1}$$

Where $r \in (0, 4], x_n \in (0, 1), n = 0, 1, 2, ..$

3.2 2D logistic map

Eq. (2) defines the 2D logistic chaotic map [4, 11]

$$\begin{aligned} x_{n+1} &= r_1 x_n (1 - x_n) + s_1 y^2 \\ y_{n+1} &= r_2 y_n (1 - y_n) + s_2 x^2 + xy \end{aligned} \tag{2}$$

Where

$$r_1 = 0.98, r_2 = 0.66, s_1 = 0.18, s_2 = 0.15$$

3.3 Median filter

The median filter technique is an ordered non-linear statistical numerical filtering technology used to increase the performance of the suggested system by removing the noise from signals and images. It is a very important post-processing step in the image-processing field, as it is well known for preserving edges while removing noise [17, 18]

$$B = midfilt3(A, [m\ n]) \tag{3}$$

Where
$m\ and\ n$ are the block mask width and height respectively.

4. The proposed system methodology

In this part, secret random key generation, encryption proposed system, and proposed algorithms are clarified.

4.1 Secret random key generation

A 1D logistic map is utilized to generate the secret random key sequence as in Eq. (10). It is sensitive to the foremost state and control parameters. A secret key of 256-bit $K = \{k_1, k_2, ... k_{64}\}$, where k_i represents 64-bit hexadecimal to produce the initial values. The new suggested Eqs. (4)-(8) are used to initialize the parameters of 1D and 2D logistic chaotic maps.

$$x_0 = ((k_1 + k_2 + .. + k_{20})\ mod\ 256)/256\ mod\ 1 \tag{4}$$

$$x_1 = ((k_{21} + k_{22} + .. + k_{44})\ mod\ 256)/256\ mod\ 1 \tag{5}$$

$$x_2 = ((k_{45} + k_{46} + .. + k_{64})\ mod\ 256)/256\ mod\ 1 \tag{6}$$

In addition, the initial value (y_0) is obtained from Eq. (7), and (r) is obtained from the suggested Eq. (8)

$$y_0 = (x_0 + x_1 + x_2)\ mod\ 1 \tag{7}$$

$$r = 3.8 + mystream \tag{8}$$

Where

$$mystream = RandStream('mt199337ar') \tag{9}$$

The sequence, $key_x = \{x_1, x_2, ... x_{w \times h}\}$, is generated using a 1D logistic map. After that,

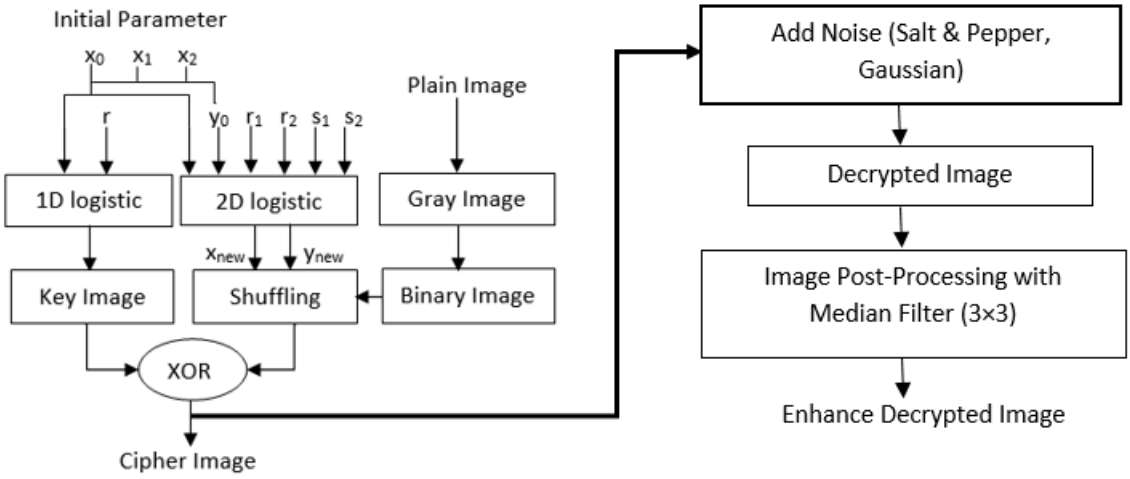
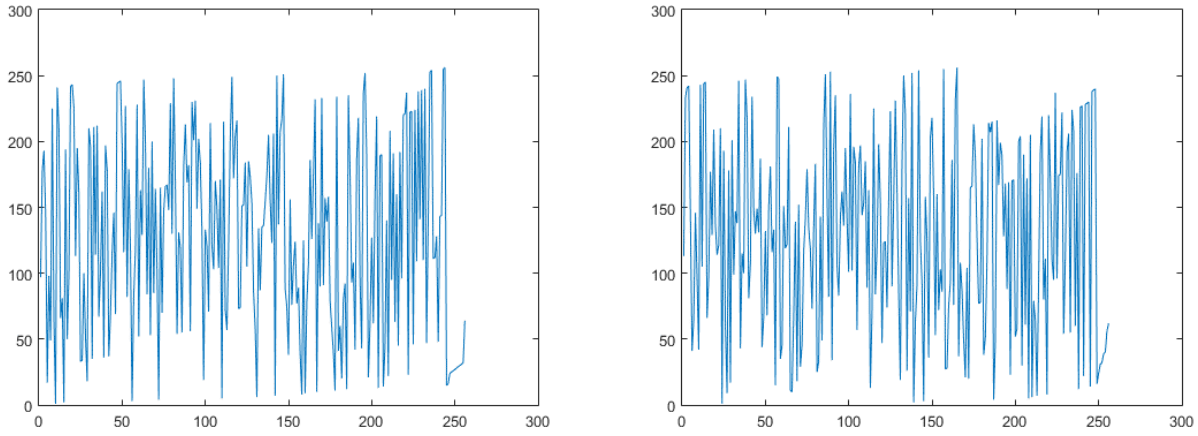


Figure. 1 Image encryption and decryption with a median filter to retrieve an enhanced decrypted image



(a) (b)

Figure. 2 Clarify the high randomness positions generated: (a) x_{new} and (b) y_{new}

applying the new proposed Eq. (10) after 400 iterations to increase the randomness and the complexity of the 1D logistic map.

$\forall k, 1 \le k \le w \times h$ and $\forall n, 0 \le n \le (w \times h) - 1$

$$x_k = int\left(floor\left(mod\left((x_{400+n} \times 10^{16}), 256\right)\right)\right) \quad (10)$$

Where

w *and* h are the width and the height of the image

4.2 The proposed system steps

The proposed method must satisfy the two important cryptography properties confusion and diffusion. First, transform the color image $\boldsymbol{I}$ to grayscale and then to binary. After that, shuffle the binary image $\boldsymbol{Sh_b}$ using a 2D logistic map as a diffusion step. Then, the $\boldsymbol{XOR}$ operation applies between the key generated using Eq. (10) and the shuffled image to generate the cipher image using Eq. (11) as a confusion step.

$$C = Sh_b \oplus key_b \quad (11)$$

Finally, applying the median filter to enhance the decrypted image after adding noise. Fig. 1 shows the proposed system for encryption and decryption with post-processing smoothing using the median filter (3×3).

5. Results and discussion

The suggested system used data set from the signal and image processing institute (SIPI) affiliated with the University of Southern California (USC) for evaluating the performance of the proposed system [19]. The system applied in HP computer Core i7-10510U CPU 1.80-2.30 GHz 16GB RAM and Matlab R2020a.

Algorithm 1: The proposed system

Input: Plain Image $\boldsymbol{I(w,h)}$
$\boldsymbol{w,h}$**:** Image width and height
$\boldsymbol{r_1 = 0.98, r_2 = 0.66, s_1 = 0.18,\ s_2 = 0.15}$
Output: Encrypted Image $\boldsymbol{C(w,h)}$**.**
Step 1: Compute the initial parameter of 1D and 2D logistics:
- $\boldsymbol{x_0 = 0.9375}$ using **Eq. (4)**
- $\boldsymbol{y_0 = 0.3906}$ using **Eq. (7)**
- $\boldsymbol{r = 3.814723686393179}$ using **Eq. (8)**

Step 2:: Transform the $\boldsymbol{RGB}$ plain image $\boldsymbol{I}$ into grayscale $\boldsymbol{Gray(w,h)}$**.**
Step 3: Convert the contents of $\boldsymbol{Gray}$ matric that contain numbers in the range [1,256] to binary form $\boldsymbol{G_b}$ $\boldsymbol{(w,h)}$ as a confusion step.
Step 4: Generate new positions x_{new} and y_{new} by **2D** logistic map using **Eq. (2)** as a diffusion step. Where $[\boldsymbol{x_{new}, y_{new}}]$ the new positions as clarified in **Fig 2**.
Step 5: Shuffle the binary Gray image with the new positions generated in the previous step.$\boldsymbol{G_b}$ is shuffled as follows: $\forall i, 1 \le i \le w$ and $\forall j, 1 \le j \le h$ and $\boldsymbol{Sh_G(i,j) = G_b(x_{new}, y_{new})}$
Step 6: Apply **Algorithm 2** to generate a random complicated key, $\boldsymbol{Key = \{k_1, k_2, \dots, k_{w*h}\}}$, as in **Eq. (10)** using a 1D logistic chaotic map.
Step 7: Convert the 1D key vector to 2D key image .Then into binary form to generate $\boldsymbol{key_b(w,h)}$
Step 8: Apply $\boldsymbol{XOR}$ operation between the shuffled image and the secret key to produce the cipher image as obtained in **Eq. (11)**.
Step 9: Compute the metrics between the plain and encrypted image before adding any attacks.
Step 10: The decryption process reversed the encryption steps to retrieve the decrypted image.
Step 11: Compute the metrics between the plain and decrypted image before adding any attacks.

Algorithm 2: Secret Key Generation

Input:$\boldsymbol{w,h}$ width and height of the image
Output: Secret key as 2D key Image $\boldsymbol{Key[w,h]}$
Step 1: Compute the Initial Parameter of 1D logistic map $\mathbf{x_0}$ and **r** using **Eq. (4)** and **Eq. (8)** respectively. i = 1; j = 1;
Step 2: for loop = 0 to ((w ∗ h) − 1) + 400
begin
Step 3: Compute the random sequence $\mathbf{x_k}$ using **Eq. (1)**, and then **Eq. (10)**

$$x_{new} \leftarrow r x_{old}(1 - x_{old})$$

$$x_{new2} \leftarrow int\left(floor\left(mod\left((x_{new} \times 10^{16}), 256\right)\right)\right)$$

Step 4: **If** (loop >= 400)
save the x_{new2} in Key[i, j] as
$Key[i,j] \leftarrow x_{new2}$; i + +;
end if
Step 5: if (i > w)
j + +; i = 1;
end if
Step 6: if (j > h)
break;
end if
Step 7: $x_{old} = x_{new2}$; **Goto Step1**
end for
Step 8: Convert the secret key **Key**[**w**, **h**] into a binary form $\mathbf{Key_b}$[**w**, **h**]

Algorithm 3: Add Noise and Median Filter smoothing

Input: Encrypted Image $\boldsymbol{C(w,h)}$
$\boldsymbol{w,h}$: Image width and height
Output: Smoothly-noisy Image $\boldsymbol{Smooth(w,h)}$
Step 1: Add noise to the encoded image to check the performance of the suggested system with two kinds of noise like salt and pepper and Gaussian noise.
Step 2: Decrypted the noisy image $\boldsymbol{denoise(w,h)}$
Step 3: Compute the metrics between the plain and decrypted noisy image.
Step 4: Apply median filter (3×3) to enhance the decrypted image as calculated in the following equation:

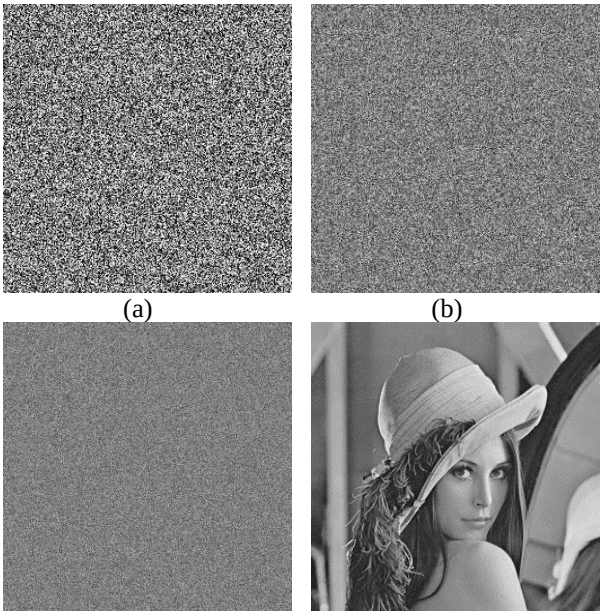
$$Smooth(w,h) = midfilt3(Denoisy, [3\ 3])$$

Step 5: Compute the metrics between the plain and decrypted smoothly-denoise image.
Step 6: Compare our results with other works.

Fig. 3 presented Lena's encrypted and decrypted image in different sizes. The figure clarifies that the proposed system provides good encryption because the cipher image cannot be detected.

5.1 Key-space test

The key-space size demonstrates the power of the system to confront the brute-force attack. The secret key of the proposed system is formed from 1D and 2D parameters $r, x_0, x_1, x_2, r_1, r_2, s_1, s_2,$ and y_0. The suggested method key precision is 10^{16} and key space is approximately equal to $\mathbf{2^{317}}$ $(\mathbf{(10^{16} > 2^{53})} \times \mathbf{2^8 \times 2^{256}})$. In [8, 11] key precision is equal to 10^{14}. In [15, 16] equal to 10^{12} and 10^{15} respectively. The proposed system precision is higher than the other



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

Figure. 3 Encrypted Lena's image sizes of the proposed method: (a) 256×256, (b) 512×512, (c) 1024×1024, and (d) Decrypted Lena's image in all sizes without any loss of data

works, which indicates higher security against brute force attacks than the other works.

5.2 Differential attack test

MSE, $PSNR$, and $NPCR$, metrics are illustrated to check the efficiency of the proposed encryption system in terms of calculating the mean squared error, the differences between the original image and the image after decrypted. MSE illustrated in Eq. (12) [20-24]:

$$MSE = \frac{1}{wh}\sum_{i=0,j=0}^{w,h}(I(i,j) - C(i,j))^2 \quad (12)$$

Where $I(i,j)$ is the original image, $C\,(i,j)$ is an encrypted or decrypted image, h and w are the dimensions of the image.

PSNR (Peak signal to noise ratio) is a logarithmic quantity clarify the ratio of the maximum strength of the signal and the corrupted noise that affect the accuracy of its representation in a decibel scale (dB), as defined in Eq. (13) [20-24]:

$$PSNR = 20\ log_{10}\left(\frac{255}{\sqrt{MSE}}\right) \quad (13)$$

NPCR (Number of Pixel Change Rate) is the pixel difference ratio between two different images, as calculated in Eq. (14) [20-21, 23, 25-26].

$$NCPR(C1,C2) = \frac{\sum_{i=1}^{w}\sum_{j=1}^{h} D(i,j)}{w \times h} \times 100 \quad (14)$$

Table 1. MSE, PSNR, and NPCR of the proposed method between plain and decrypted image

Plain Image	MSE	PSNR	NPCR
$Lena_{(1024*1024)}$	0	infinity	0
$Lena_{(512*512)}$	0	infinity	0
$Lena_{(256*256)}$	0	infinity	0
Mandrill	0	infinity	0
Peppers	0	infinity	0

Table 2. MSE, PSNR, and NPCR of the proposed method between plain and cipher image

	Plain Image	MSE	PSNR	NPCR
Proposed Method	$Lena_{(1024*1024)}$	7768.40	9.22749	99.6080
	$Lena_{(512*512)}$	7762.42	9.23083	99.6078
	$Lena_{(256*256)}$	7844.52	9.18514	99.6201
	$Mandrill_{(1024*1024)}$	7184.54	9.56682	99.6222
	$Mandrill_{(512*512)}$	7239.39	9.53378	99.6063
	$Mandrill_{(256*256)}$	7359.22	9.46249	99.6353
	$Peppers_{(1024*1024)}$	8420.17	8.87759	99.6147
	$Peppers_{(512*512)}$	8386.89	8.89480	99.6132
	$Peppers_{(256*256)}$	8455.80	8.85925	99.6307
[15]	$Lena_{(512*512)}$	7747.30	9.23929	-
[16]	$Mandrill_{(256*256)}$	7188.3	9.5645	-
	$Peppers_{(256*256)}$	8199.6	8.9929	-

Where

$$D(i,j) = \begin{cases} 0, & if\ Img1(i,j) = Img2(i,j) \\ 1, & if\ Img1 \neq Img2(i,j) \end{cases} \quad (15)$$

Img1 and Img2 are plain and decrypted images

Table 1 shows that the proposed method achieves MSE=0, PSNR= ∞, NPCR=0 between the plain image and decrypted image for Lena with different sizes. The results show that the schema more secure, lossless method, which retrieves data without any loss. Table 2 shows that the proposed scheme achieved larger MSE, smaller PSNR between the plain and the cipher image, and encode a larger image size compared with [15, 16], and high NPCR reached up to 99.63.

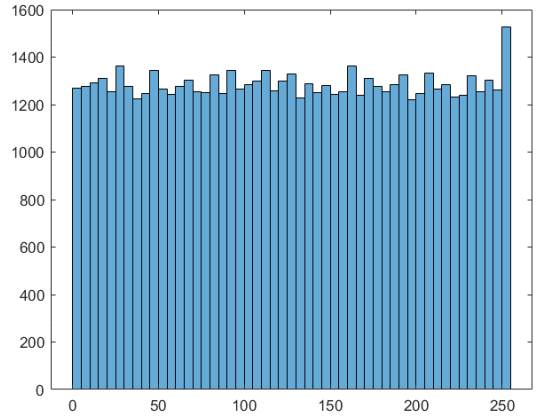
5.3 Statistical attack test

Entropy (H), correlation coefficient, and histogram analysis are information randomness degrees that are used to check the strength of the suggested method to withstand statistical attack**.** The entropy is the probability of the occurrence of each value in the encrypted image. The most efficient method must achieve the highest entropy closer to 8, as illustrated in Eq. (16) [20, 26]

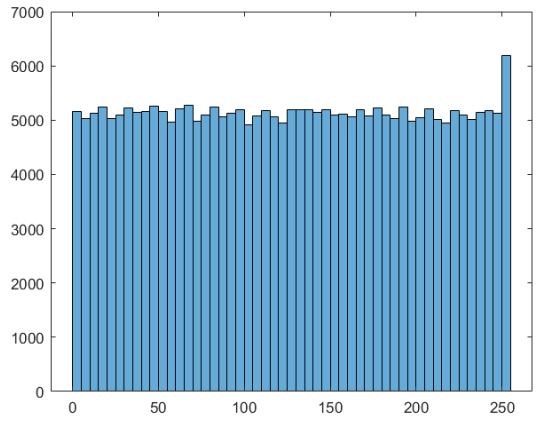
$$H(m) = -\sum_{i=1}^{L} p(m_i)\ log_2\ p(m_i) \quad (16)$$

Where

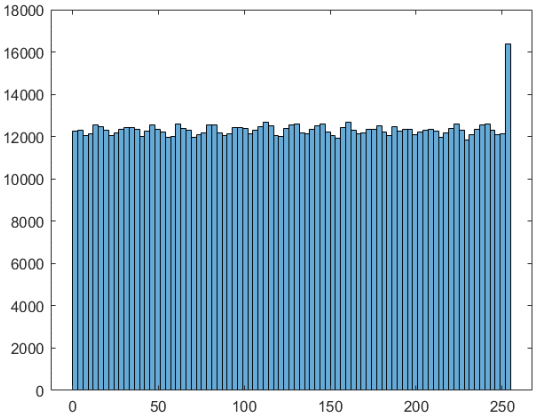
Received: March 18, 2023. Revised: May 6, 2023.



(a)

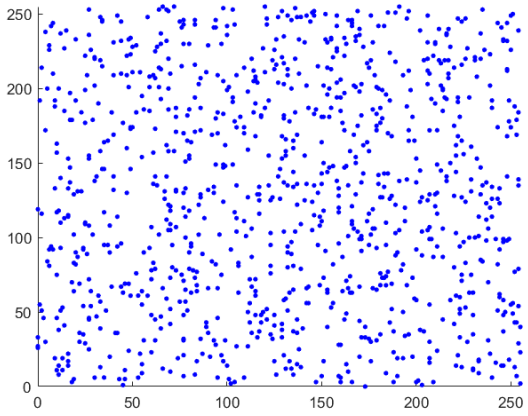


(b)

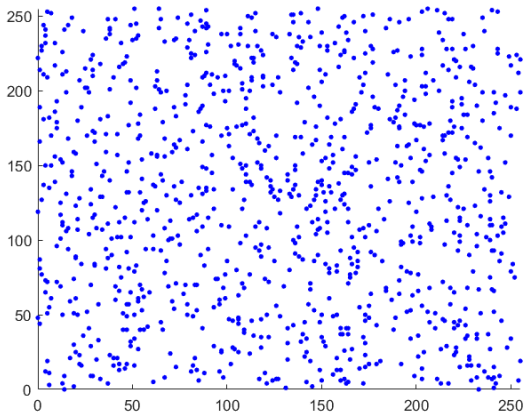


(c)

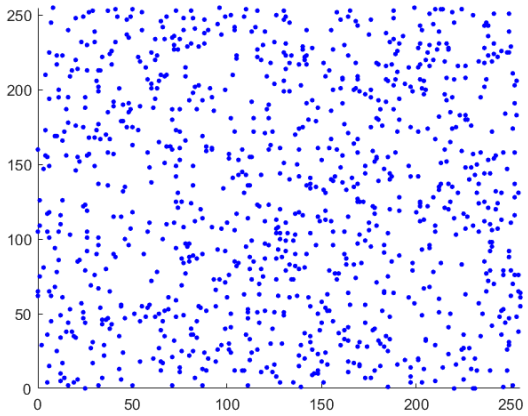
Figure. 4 The histogram of Lena's cipher with different sizes: (a) 256×256, (b) 512×512, and (c) 1024×1024



(a)



(b)



(c)

Figure. 5 Correlation coefficient between neighboring pixels for Lena (256×256) encrypted image in different directions: (a) Vertical (x,y+1), (b) Horizontal (x+1,y), and (c) Diagonal (x+1,y+1)

m_i is the i^{th} pixel for the image, and $p(m_i)$ is the probability of m_i, L=256

The correlation is counted in Eq. (17), which calculates the relationship of the close neighboring pixels in vertical, horizontal, and diagonal directions [21, 23, 25-26].

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)D(y)}} \tag{17}$$

Where

Table 3. Vertical, horizontal, and diagonal correlation, the entropy of the proposed method

	Image Name	V	H	D	Entropy
	Lena$_{(1024\times1024)}$	**-0.0047**	**-0.0076**	**-0.0048**	**7.99985**
	Lena$_{(512\times512)}$	-0.0275	-0.0262	0.0003	7.99920
	Lena$_{(256\times256)}$	-0.0042	-0.0052	-0.0010	7.99744
	Mandrill$_{(1024\times1024)}$	**-0.0080**	**-0.0057**	**-0.0067**	**7.99984**
	Mandrill$_{(512\times512)}$	-0.0091	-0.0049	-0.0066	7.99928
	Mandrill$_{(256\times256)}$	-0.0009	-0.0620	-0.0011	7.99694
	Peppers$_{(1024\times1024)}$	**-0.0089**	**-0.0012**	**-0.0006**	**7.99985**
	Peppers$_{(512\times512)}$	-0.0256	-0.0515	-0.0024	7.99933
	Peppers$_{(256\times256)}$	0.0004	-0.0178	0.0009	7.99734
[1]	Lena$_{(755*755)}$	-	-	-	7.9980
	Mandrill$_{(560*560)}$	-	-	-	7.9982
	Peppers$_{(256*256)}$	-	-	-	7.9977
[8]	Lena$_{(256\times256)}$	-0.0173	0.0118	0.0080	7.99705
	Peppers$_{(256\times256)}$	0.0350	-0.0159	-0.0096	7.99727
[10]	Lena$_{(512\times512)}$	0.0039	0.0059	-0.0050	7.9994
[11]	Lena$_{(512\times512)}$	-0.0017	-0.0010	0.0002	7.9994
	Lena$_{(256\times256)}$	0.0073	0.0072	0.0015	7.9974
[15]	Lena$_{(512*512)}$	-0.0391	0.0004	0.0030	7.9993
	Mandrill$_{(512*512)}$	0.0028	0.0031	-0.0029	7.9993
	Peppers$_{(512*512)}$	0.0432	0.0019	0.0254	7.9994

$$E(x) = \frac{1}{S}\sum_{i=1}^{S} x_i$$

$$D(x) = \frac{1}{S}\sum_{i=1}^{S}\left(x_i - E(x_i)\right)^2$$

$$cov(x,y) = \frac{1}{S}\sum_{i=1}^{S}\left(x_i - E(x_i)\right)\left(y_i - E(y_i)\right)$$

x and y are adjacent pixels, S is equal to 1000 which means the number of selected adjacent pixels of the image, and $D\ (x)$ is the variance.

Fig. 4, Fig. 5, and Table 3 show that the proposed schema can withstand statistical attack according to Lena's flat, stable histogram with different sizes, high entropy, and small correlation coefficient between the adjacent pixels in vertical, horizontal, and diagonal directions respectively. The proposed method encrypts a higher image size **(1024×1024)** and achieves a higher entropy equal to **7.99985** compared with other works [1, 8, 10, 11, and 15].

5.4 Noise attack test

High-security encryption systems must resist all types of noise during transmissions over a channel such as salt and pepper, and Gaussian. Different densities of salt and pepper noise $\boldsymbol{d} = \{\mathbf{5}\%, \mathbf{10}\%, \mathbf{20}\%, \mathbf{50}\%\}$ and vary variances of Gaussian noise $\boldsymbol{v} = \{\mathbf{0.001}, \mathbf{0.01}, \mathbf{0.1}\}$ with no mean are appended to the Lena cipher image.

Table 4 clarified MSE, PSNR, and Mean Absolute error (MAE), or the variation between the Lena authentic image and decoded under two types of noise before and after smoothing the decrypted image. The results show that the proposed method is stronger than the other works [9, 14] against two types of noise that achieve too small MAE and MSE between the plain image and the decrypted image.

Table 4. MSE, PSNR after smoothing**,** and MAE (differences) between plain and decrypted images for Lena (256×256) under noise

Noise Type	Ratio	MSE	PSNR	MAE before smooth	MAE after smooth
Salt & pepper	**5%**	91.174	28.5321	3.67114	1.5044
	10%	103.400	27.9856	7.27275	4.79283
	20%	139.402	26.6881	14.5484	5.65993
	50%	644.512	20.0385	37.2704	13.3160
[9]	**5%**	406.60	22.00	-	-
	10%	782.00	19.20	-	-
[14]	**5%**	464.05	21.47	-	-
Gaussian	**0.001**	337.411	22.8492	29.7063	13.3787
	0.01	350.869	22.6794	29.8753	13.5537
	0.1	533.293	20.8611	37.1993	17.4006
[9]	**0.01**	1967.8	15.2	-	-
	0.1	4665.3	11.4	-	-
[14]	**0.001**	840.07	18.89	-	-

Table 5. MSE, PSNR after smoothing, and MAE (differences) between plain and decrypted images for Lena (1024×1024) under noise

Noise Type	Ratio	MSE	PSNR	MAE before smooth	MAE after Smooth
Salt & pepper	**5%**	3.09279	43.2273	3.63785	0.87167
	10%	4.95393	41.1813	7.34084	1.04906
	20%	16.7121	35.9005	14.5949	1.53665
	50%	401.087	22.0984	36.4513	**8.15541**
Gaussian	**0.001**	246.526	24.2122	29.6645	11.6910
	0.01	247.423	24.1964	29.7516	11.7139
	0.1	447.417	21.6237	37.0724	**16.0991**

Figs. (6-9) clarify that the proposed system achieves excellent results against the two kinds of noise tested on Lena (256×256) and (1024×1024) respectively when compared with [10, 11, 13-16]. The schema in [10, 11, 13-16] resists 20%, 10%, and 5% salt and pepper noise respectively on Lena (256×256). In [16] resists 0.02 Gaussian noise on Lena (256×256).

From Table 5, MAE of 8.1 indicates the proposed method resists a higher ratio of noise and retrieves approximately 92% of the image with 50% salt and pepper noise. MAE of 16.0 retrieves 84% with 0.1% Gaussian noise, which indicates the proposed method is higher security against noise attacks on Lena with a larger size (1024×1024) than other works.

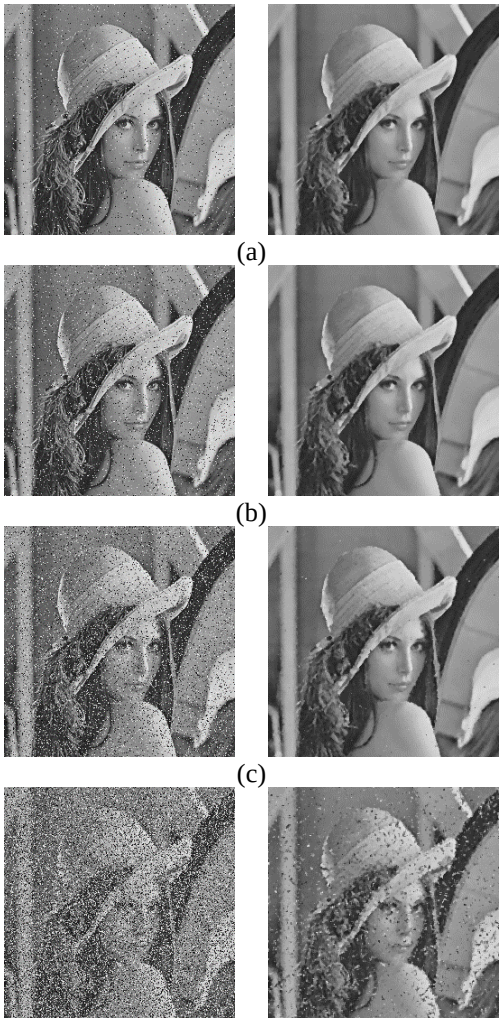


Figure. 6 Lena (256×256) results under salt and pepper noise with various densities before and after smoothing: (a) 0.05, (b) 0.10, (c) 0.20 and (d) 0.50

Table 6. Speed analysis

Method	Image	Encrypt Time (s)	Decrypt Time (s)
Proposed	$\text{Lena}_{(1024\times1024)}$	1.7856	2.0104
	$\text{Lena}_{(512\times512)}$	0.4487	0.4383
	$\text{Lena}_{(256\times256)}$	0.1623	0.1290
[8]	$\text{Lena}_{(256\times256)}$	2.4432	-
[13]	$\text{Lena}_{(256\times256)}$	0.32	-
[14]	$\text{Lena}_{(256\times256)}$	4.4998	-
[16]	$\text{Lena}_{(256\times256)}$	0.6007	0.3804

5.5 Speed analysis test

The efficient method should take a small execution time in encryption and decryption. The proposed approach is faster than the other works [8, 13, 14, 16], and Lena's encryption and decryption time are presented in Table 6.

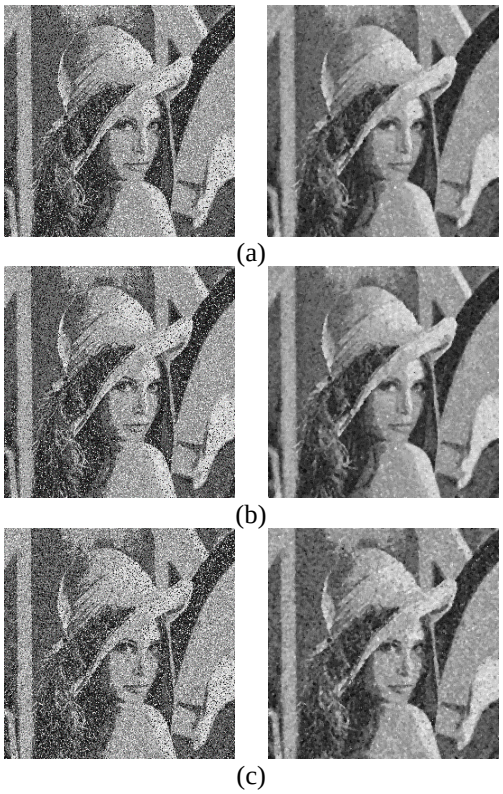


Figure. 7 Lena (256×256) result under Gaussian noise with various variances before and after smoothing: (a) 0.001, (b) 0.01, and (c) 0.1

5.6 Key sensitivity test

The suggested system is sensitive to a trivial change in key parameters, which indicates that the proposed schema is more secure when changing one parameter at a time. Consider $\boldsymbol{K}$ the secret key, $\boldsymbol{r} = \mathbf{3.814723686393179}$ $\boldsymbol{and}$ $\boldsymbol{x_0} = \mathbf{0.937}$ are two chaotic parameters that are used to encrypt Lena's image by the proposed system. **Fig. 10a** clarifies the decrypted Lena image when the key $(\boldsymbol{r})$ is changed (only one bit LSB) to $(\boldsymbol{r'})$ while keeping the other keys unchanged. Fig. 10 (b) clarifies the decryption of the encrypted Lena image when the secret key $(\boldsymbol{x_0})$ is changed (only one bit LSB) to $(\boldsymbol{x_0'})$ while keeping the other keys unchanged.

Fig. 10 clarifies that the decoding process is unable to restore the authentic image when a very small change is made to the secret key, indicating very sensitivity to secret keys.

6. Conclusions

This paper proposes a new encoding method for the grayscale image. Two logistic chaotic maps are used, a 1D logistic chaotic map to produce complex, high randomness, secure key and a 2D chaotic map to change the pixels locations of the original image

Figure. 8 Lena (1024×1024) results under salt and pepper noise with various densities before and after smoothing: (a) 0.05, (b) 0.10, (c) 0.20 and (d) 0.50.

randomly. A median filter applies to get an efficient, highly secure cryptosystem that retrieves the most important information with too low lost information from decrypted images. From the figures, tables, and comparison presented, the results show that the proposed system encodes an image size of 1024×1024 larger than the other works and resists a higher ratio of two types of noise attack equal to 50% salt and pepper noise, and 0.1 Gaussian noise while the other works can't. The suggested system is faster than the other works as shown in Table 6, sensitive to a tiny change in the secret key, and ability to resist several types of attacks like differential, brute force, and statistical attacks.

Conflicts of interest

The authors declare no conflict of interest. We have no known competing financial or personal interests. The relationships presented in this paper.

Author contributions

"conceptualization, Ibtisam A. Taqi and Mela G. Abdul-Haleem; methodology, Ibtisam A. Taqi; software, Ibtisam A. Taqi; validation, Ibtisam A. Taqi, Mela G. Abdul-Haleem; formal analysis, Ibtisam A. Taqi; investigation, Ibtisam A. Taqi; resources, Ibtisam A. Taqi; data curation, Ibtisam A. Taqi; writing—original draft preparation, Ibtisam A. Taqi; writing—review and editing, Ibtisam A. Taqi; visualization, Ibtisam A. Taqi; supervision, Ibtisam A. Taqi; project administration, Ibtisam A. Taqi; funding acquisition, Ibtisam A. Taqi and Mela G. Abdul-Haleem".

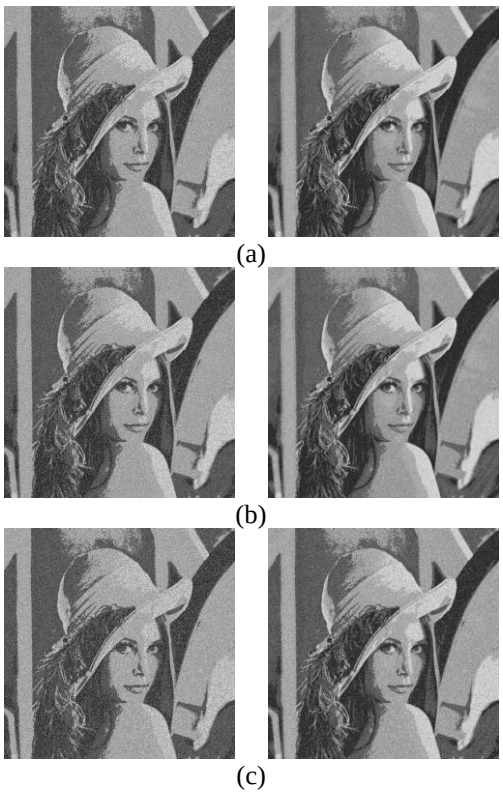


Figure. 9 Lena (1024×1024) results under Gaussian noise with various variances before and after smoothing: (a) 0.001, (b) 0.01, and (c) 0.1

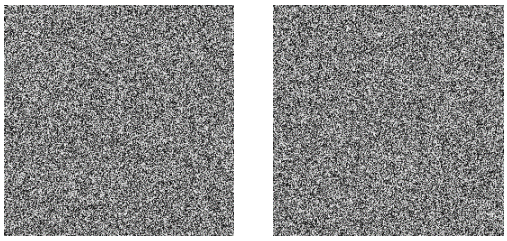


Figure. 10 Suggested system results for the key sensitivity to Lena: (a) The image after decryption with r' and (b) The image after decryption with ${x_0}'$

References

[1] A. A. Abdallah, and A. K. Farhan, "A New Image Encryption Algorithm Based on Multi Chaotic System", *Iraqi Journal of Science*, Vol. 6, No. 1, pp. 324-337, 2022.

[2] Y. Tao, W. Cui, Z. Zhang, and T. Shi, "An Image Encryption Algorithm Based on Hopfield Neural Network and Lorenz HyperChaotic System", *IAENG International Journal of Computer Science*, Vol. 49, No. 4, 2022.

[3] Y. Zha, R. Meng, Y. Zhang, and Q. Yang, "Image encryption algorithm based on a new chaotic system with Rubik's cube transform and Brownian motion model", *Optik*, Vol. 273, pp. 170342, 2023.

[4] Y. Wu, G. Yang, H. Jin, and J. P. Noonan, "Image encryption using the two dimensional logistic chaotic map", *Journal of Electronic Imaging*, Vol. 21, No. 1, 2012.

[5] Z. Tang, X. Zhang, and W. Lan, "Efficient image encryption with block shuffling and chaotic map", *Multimedia Tools and Applications*, Vol. 74, pp. 5429–5448, 2014.

[6] L. Liu and S. Miao, "A new simple one-dimensional chaotic map and its application for image encryption", *Multimedia Tools and Applications*, Vol. 77, pp. 21445–21462, 2018.

[7] S. Zhu, G. Wang, and C. Zhu, "A Secure and Fast Image Encryption Scheme Based on Double Chaotic S-Boxes", *Entropy*, Vol. 21, No. 8, 2019.

[8] M. A. B. Farah, A. Farah, and T. Farah, "An image encryption scheme based on a new hybrid chaotic map and optimized substitution box", *Nonlinear Dynamics*, Vol. 99, pp. 3041–3064, 2020.

[9] M. Z. Talhaoui and X. Wang, "A new fractional one dimensional chaotic map and its application in high speed Image Encryption", *Information Sciences*, Vol. 550, pp. 13-26, 2021.

[10] M. Z. Talhaoui, X. Wang, and M. A. Midoun, "Fast image encryption algorithm with high security level using the Bülban chaotic map", *Journal of Real-Time Image Processing*, Vol. 18, pp. 85–98, 2021.

[11] A. Elghandour, A. Salah, and A. Karawia, "A new cryptographic algorithm via a two-dimensional chaotic map", *Ain Shams Engineering Journal*, Vol. 13, No. 1, p. 101489, 2022.

[12] Q. Lu, L. Yu, and C. Zhu, "Symmetric Image Encryption Algorithm Based on a New Product Trigonometric Chaotic Map", *Symmetry*, Vol. 4, No. 2, p. 373, 2022.

[13] J. Zheng and T. Lv, "Image encryption algorithm based on cascaded chaotic map and improved Zigzag transform", *IET Image Processing*, Vol. 16, No. 14, pp. 3863-3875, 2022.

[14] S. Wang, L. Hong, and J. Jiang, "An image encryption scheme using a chaotic neural network and a network with multistable hyperchaos", *Optik*, Vol. 268, p. 169758, 2022.

[15] I. Yasser, F. Khalifa, M. A. Mohamed, and A. S. Samrah, "A New Image Encryption Scheme Based on Hybrid Chaotic Maps", *Complexity*, Vol. 2020, p. 23, 2020.

[16] S. Zhu, X. Deng, W. Zhang, and C. Zhu, "Image Encryption Scheme Based on Newly Designed Chaotic Map and Parallel DNA Coding", *Mathematics*, Vol. 11, No. 1, p. 231, 2023.

[17] Y. Xiong, A. He, and C. Quan, "Security analysis of a double-image encryption technique based on an asymmetric algorithm", *Journal of the Optical Society of America A*, Vol. 35, pp. 320-326, 2018.

[18] M. I. F. Allah and M. M. Eid, "Chaos based 3D color image encryption", *Ain Shams Engineering Journal*, Vol. 1, No. 1, pp. 67-75, 2020.

[19] A. G. Weber, "The USC-SIPI image database: Version 5", *http://si pi. usc. edu/database/ (2006)*. [Accessed 10 1 2021].

[20] F. Masood, J. Ahmad, S. A. Shah, S. S. Jamal, and I. Hussain, "A novel hybrid secure image encryption based on Julia set of fractals and 3D Lorenz chaotic map", *Entropy*, Vol. 22, No. 3, p. 274, 2020.

[21] B. Yousif, F. Khalifa, A. Makram, and A. Takieldeen, "A novel image encryption decryption scheme based on integrating multiple chaotic maps", *AIP Advances*, Vol. 10, No. 7, p. 075220, 2020.

[22] J. Toama and N. H. M. Ali, "A secure cipher for the gray images based on the Shamir secret sharing scheme with discrete wavelet haar transform", *Journal of Mechanics of Continua and Mathematical Sciences*, Vol. 15, No. 6, pp. 2454 -7190, 2020.

[23] Y. Xian and X. Wang, "Fractal sorting matrix and its application on chaotic image encryption", *Information Sciences*, Vol. 547, pp. 1154-1169, 2021.

[24] A. Yousif and A. H. Kashmar, "Key Generator to Encryption Images Based on Chaotic Maps", *Iraqi Journal of Science*, Vol. 60, No. 2, pp. 362–370, 2019.

[25] S. Jassim and S. M. Hameed, "A Modified Advanced Encryption Standard for Color

Images", *Iraqi Journal of Science*, Vol. 63, No. 1, pp. 294-312, 2022.

[26] Z. Hua, Z. Zhu, Y. Chen, and Y. Li., "Color image encryption using orthogonal Latin squares and a new 2D chaotic system", *Nonlinear Dynamics*, Vol. 104, pp. 4505-4522, 2021.