Malaysian Journal of Mathematical Sciences 18(1): 107–126 (2024)
https://doi.org/10.47836/mjms.18.1.07

Malaysian Journal of Mathematical Sciences

Journal homepage: https://mjms.upm.edu.my

A Novel Image Encryption Algorithm Involving A Logistic Map and A Self-Invertible Matrix

Al-Saffar, N. F. H.*[1], Alkhayyat, H. K. H.[2], and Obaid, Z. K.[3]

[1,2]*Department of Mathematics, Faculty of Computer Science & Mathematics, University of Kufa, Iraq*
[3]*Department of Computer Science, Faculty of Computer Science & Mathematics, University of Kufa, Iraq*

E-mail: najlaa.hameed@uokufa.edu.iq
**Corresponding author*

Received: 28 October 2023
Accepted: 18 December 2023

Abstract

To prevent sensitive images shared on social media from being stolen, researchers are seeking to find and innovate image encryption algorithms. Since attackers attempt to exploit it to access encrypted images, the problem of generating keys as the first stage of any encryption algorithm is regarded as a critical problem. This issue was dealt with in this paper through the use of the logistic map. This map must be calculated using specific conditions and special parameters to maintain its chaos. The Diffie-Hellman key exchange algorithm was modified to address this problem since the shared keys now consist of a 16-element vector. The first two elements will be used for a logistic map. The entire vector was used to create a self-invertible matrix. The encryption level entails two operations: the first is matrix multiplication with a vector, and the second is a bitwise XOR operation between two matrices. A proposed encryption algorithm is capable of producing encrypted images that are challenging to decrypt through the use of stringent security tests. The outcomes of each test show how well the proposed encryption algorithm performs compared with other algorithms and how to resist the attacks.

Keywords: cryptosystems; image encryption; Diffie-Hellman key exchange; logistic map; self-invertible matrix.

1 Introduction

A cryptosystem is one of the mathematical procedures used to keep and save sensitive data, such as text, music, photos, etc., from prying eyes. Each encryption scheme has three levels: key creation, encryption, and decryption [36]. A sender encrypts plain data to create encrypted data as a result of the encryption level before sending the data to the recipient over the internet. As a result of the decryption level processing, the recipient will be able to retrieve the plain data from the encrypted data. Both levels of decryption and encryption will use the outcomes of the key creation level. Cryptosystems are divided into symmetric key cryptosystems and asymmetric key cryptosystems. In contrast, there are two types of keys -public which is for asymmetric key cryptosystems, and private key which is used for both encryption and decryption levels in symmetric key cryptosystems [26, 24].

Image encryption algorithms are crucial to image security since they shield the plain image's privacy from unauthorized individuals [28]. Diffie and Hellman established the Diffie-Hellman key exchange (DHKE) algorithm, the first asymmetric key cryptosystem in 1976 [11]. It is one of the greatest contributions to solving the problem of key exchange, it is not about establishing a shared-secret key, it is about doing it in such a way that anyone who is there at the communication between the devices does not find the key [4]. The most amazing thing about DHKE is that the communication between sender and receiver could happen over the public channel, while still being secure because it is based on the difficulty of solving the Diffie-Hellman problem [26].

In 2015, the DHKH algorithm was used to create two algorithms for image watermarking [4], it has been proven that this proposed algorithm is more secure than other algorithms. In 2017, a DHKE algorithm was improved by creating two shared keys the second one derived from the first one in such a way that made the improvement algorithm stronger than the original one [19]. In 2022, a new mathematical model for exchanging keys was introduced; it was another improvement of the DHKE algorithm [18]. The prime numbers and the possibility of using integer numbers were involved in this improvement. Indeed, this algorithm reduces the probability of attacking prime numbers. In this work, the DHKE algorithm will be modified in such a way that the shared keys will be 16 rather than one; these keys will be derived from the first shared key.

In modern image encryption methods, the chaos theory is frequently used, such as in [29, 20, 34]. Chaos theory is one of the nonlinear mathematical models for dynamical systems, and one of its advantages is that it can be handled smoothly on electronic devices. The logistic map [23] is among the simplest nonlinear chaotic maps and is therefore of particular interest to researchers. In 2018, a one-dimensional logistic map together with another chaotic map was involved in the generation of a key sequence for encrypting an image [29]. In 2019, a three-dimensional logistic map was modified to be the first step of the proposed image encryption algorithm to generate keystream by it[20].

In 2020, a sine square logistic map was one of five chaotic maps that were a basis for an image encryption algorithm, it was a tool to create the other chaotic maps [34]. In 2022, the one-dimension logistic map was a tool to build IP table of size 128×128 to be the first level of building an algorithm of image encryption [1].

In 2023, a six-dimension logistic map was a tool to generate keys to construct a secure system for encrypting images [27], as well, the logistic map was enhanced to introduce image encryption algorithm [3], the enhanced map was dependent on interbreed backwards and forward perturbation methods. Also in 2023, the logistic map was one item of a combination containing Hill ciphers, a unimodular matrix to build an algorithm of image encryption [7].

All the proposed encryption algorithms that have been referred to, which used the logistic map in their work, indicate that the proposed algorithms provide high security against several types of attacks. So, an actual tool for ensuring security is a logistics map. It will therefore be a crucial tool in this work since it will be applied at the key creation level of the suggested encryption algorithm.

This work discusses the problems that people may face who try to preserve their digital images by encrypting them. The first problem was solved by introducing a modification of the DHKE algorithm to make it 16 instead of one key. This modification adds additional time that the unauthorized persons can take to find those keys. The second problem was solved by involving the logistic map to enhance security. It will be used to create a square matrix. The third problem was solved by creating a self-invertible matrix, as calculating the inverse is considered an expensive operation. Using these matrices, a new algorithm for encrypting images was proposed as a solution for the fourth problem of security, which can encrypt grey images, to make them unpredictable images, with an extraordinary security as to other encryption image algorithms that are based on chaotic maps. It is also resistant to various known attacks.

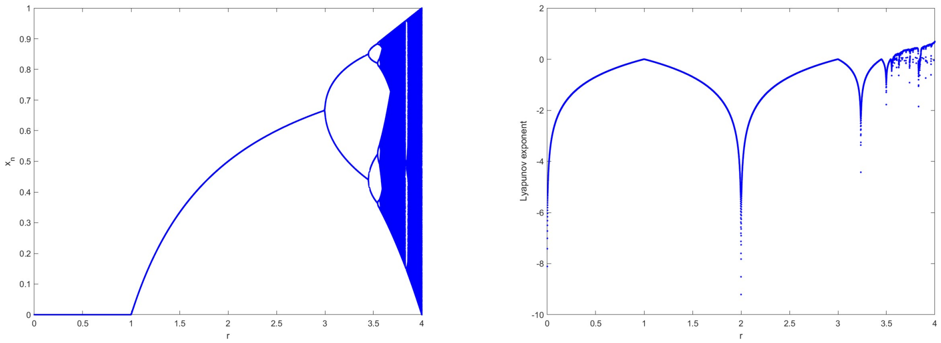
The structure of the paper was organized by introducing a brief on the logistic map with a discussion of its chaotic behaviour in Section 2. Section 3 discusses a way of generating a self-invertible matrix of dimension 8×8, in Section 4 a modification of Diffie-Hellman key exchange algorithm where the shared-secret key will be extended to be 16 shared-secret key. Section 5 contains a proposed algorithm for encrypting images. In Section 6 a series of experiments were conducted and discussed their results. Finally, Section 7 gives a conclusion of this work together with hints and ideas for future works.

2 Logistic Map

The logistic chaotic map is a one-dimension dynamical system which is nonlinear and contains the potential to behave chaotically. The uses of this map are many and different as a result of the properties that distinguish it from other maps. One of these characteristics is the one control parameter in its straightforward mathematical structure. This parameter has the ability to regulate the map's erratic behaviour [33]. The first attempt to define a logistic map was in 1845 by the scientist Pierre-FranÃ§ois de Verhulst (1804-1849). As a mathematical map, it is defined as:

$$L_\alpha(x_s) = x_{s+1} = \alpha x_s(1 - x_s), \qquad x_s \in [0, 1],$$

where x_s represents a state of the s^{th} iteration, $x_0 \in [0,1] \& \alpha \in [0,4]$ are initial and control points respectively, The behaviour of the logistic map is determined by calculating the bifurcation diagram [10]. The bifurcation of the diagram depends on the logistic map parameter. A best value of α is $3.666 \leq \alpha \leq 4$ as in Figure 1. Another way in which the behaviour of a logistic map can be determined is through computing the Lyapunov exponent [9], which measures the rate of divergence of nearby trajectories in the logistic map. A logistic map is sensitive to the initial conditions and displays chaotic behaviour when the Lyapunov exponent is positive. According to Figure 1 when $\alpha = 4$ then the logistic map has the largest Lyapunov exponent, which is nearly equal to 0.6887. Consequently, it is possible to spot a chaotic regime using both of bifurcation diagram and the Lyapunov exponent.



(a)

(b)

Figure 1: (a) Bifurcation diagrams of the logistic map, (b) Lyapunov exponents diagram of logistic map.

3 Self Invertible Matrix

A matrix $M_{n\times n}$ is called invertible if and only if $\exists\ N_{n\times n} \neq M_{n\times n}$ such that $MN = NM = I_n$, the inverse of matrix M is symbolically represented by M^{-1}. In case $N = M$ the matrix M is called the self-invertible matrix. In this section, generating a self-invertible matrix of dimension 8×8 will be discussed concerning positive integers, i.e. all operating will be considered modulo arithmetic on integer numbers. Consider $M_{8\times 8}$ be the integer value matrix;

$$M_{8\times 8} = \begin{pmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,m} \\ m_{2,1} & m_{2,2} & \cdots & m_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ m_{8,1} & m_{8,2} & \cdots & m_{8,8} \end{pmatrix} = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix},$$

where,

$$M_{11} = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} \\ m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} \\ m_{3,1} & m_{3,2} & m_{3,3} & m_{3,4} \\ m_{4,1} & m_{4,2} & m_{4,3} & m_{4,4} \end{pmatrix}, \quad M_{12} = \begin{pmatrix} m_{1,5} & m_{1,6} & m_{1,7} & m_{1,8} \\ m_{2,5} & m_{2,6} & m_{2,7} & m_{2,8} \\ m_{3,5} & m_{3,6} & m_{3,7} & m_{3,8} \\ m_{4,5} & m_{4,6} & m_{4,7} & m_{4,8} \end{pmatrix},$$

$$M_{21} = \begin{pmatrix} m_{5,1} & m_{5,2} & m_{5,3} & m_{5,4} \\ m_{6,1} & m_{6,2} & m_{6,3} & m_{6,4} \\ m_{7,1} & m_{7,2} & m_{7,3} & m_{7,4} \\ m_{8,1} & m_{8,2} & m_{8,3} & m_{8,4} \end{pmatrix}, \quad \text{and} \quad M_{22} = \begin{pmatrix} m_{5,5} & m_{6,5} & m_{5,6} & m_{5,8} \\ m_{6,5} & m_{6,6} & m_{6,7} & m_{6,8} \\ m_{7,5} & m_{7,6} & m_{7,7} & m_{7,8} \\ m_{8,5} & m_{8,6} & m_{8,7} & m_{8,8} \end{pmatrix}.$$

where $m_{i,j}$ are integer number for $i, j = 1, 2, \ldots, 8$.

The generalization states that the matrix $M_{8\times 8}$ is invertible if and only if the following criteria are met: $M_{12} = I_4 - M_{11}$, $M_{21} = I_4 + M_{11}$ and $M_{22} = -M_{11}$ [2]. Since the main purpose of this study is the process of encoding images, the image will be dealt with digitally and converted into a matrix whose highest value is 256. All operations on the matrix $M_{8\times 8}$ will be done by modulo

256. As an example suppose that

$$M_{11} = \begin{pmatrix} 18 & 3 & 13 & 4 \\ 2 & 9 & 8 & 16 \\ 4 & 12 & 16 & 3 \\ 16 & 6 & 2 & 9 \end{pmatrix},$$

then,

$$M_{12} \equiv I_4 - M_{11} \mod 256 \equiv \begin{pmatrix} 239 & 253 & 243 & 252 \\ 254 & 248 & 248 & 240 \\ 252 & 244 & 239 & 253 \\ 240 & 250 & 254 & 248 \end{pmatrix},$$

$$M_{21} \equiv I_4 + M_{11} \mod 256 \equiv \begin{pmatrix} 19 & 3 & 13 & 4 \\ 2 & 10 & 8 & 16 \\ 4 & 12 & 19 & 3 \\ 16 & 6 & 2 & 10 \end{pmatrix},$$

and,

$$M_{22} \equiv -M_{11} \mod 256 \equiv \begin{pmatrix} 238 & 253 & 243 & 252 \\ 254 & 247 & 248 & 240 \\ 252 & 244 & 238 & 253 \\ 240 & 250 & 254 & 247 \end{pmatrix}.$$

So,

$$M_{8\times 8} = \begin{pmatrix} 18 & 3 & 13 & 4 & 239 & 253 & 243 & 252 \\ 2 & 9 & 8 & 16 & 254 & 248 & 248 & 240 \\ 4 & 12 & 18 & 3 & 252 & 244 & 239 & 253 \\ 16 & 6 & 2 & 9 & 240 & 250 & 254 & 248 \\ 19 & 3 & 13 & 4 & 238 & 253 & 243 & 252 \\ 2 & 10 & 8 & 16 & 254 & 247 & 248 & 240 \\ 4 & 12 & 19 & 3 & 252 & 244 & 238 & 253 \\ 16 & 6 & 2 & 10 & 240 & 250 & 254 & 247 \end{pmatrix},$$

which is a self-invertible matrix where $(M_{8\times 8})^{-1} = M_{8\times 8}$.

4 Modified Diffie-Hellman Key Exchange Algorithm (MDHKEA)

Exchange Algorithm (MDHKEA) A Diffie-Hellman key exchange algorithm [11] for key exchange was proposed by Diffie and Hellman in 1976, and was the first algorithm in the public key cryptosystem. The computational complexity of the discrete logarithm problem serves as the foundation for this algorithm's security. Indeed, establishing a shared secret between two parties through a Diffie-Hellman key exchange paves the way for securing data transfer over internet networks. The multiplicative group of integers modulo of a prime p and a primitive root $g \, mod \, p$ has been used in this algorithm to guarantee the shared key's secrecy. In simple words, this algorithm can be summed up as follows: if $Alice$ and Bob went to exchange a key, $Alice$ would calculate her a public key, $A \equiv g^a \mod p$, a secret key, and exchange it with the public key of

Bob, $B \equiv g^b \mod p$ where b is his secret key. Now that $Alice$ and Bob have calculated the private key and discovered the shared secret key, they can obtain the shared secret key $B^a \equiv A^b \equiv g^{ab} \mod p$. In this study, the Diffie-Hellman algorithm is extended further to 16 shared-secret keys are calculated. Assuming that the shared key is k_1, then the other shared keys could be as:

$$k_i \equiv k_{i-1}^2 \mod p, \qquad \text{where } i = 2, 3, \ldots, 16.$$

For instance, $Alice$ and Bob publicly concur of prime $p = 23$ where its primitive root $g = 5$. $Alice$ calculates her public key, it is $A \equiv s^4 \mod 23 \equiv 4$ where $a = 4$ which is her secret key, she exchanges it with the public key of Bob, $B \equiv 5^3 \mod 23 \equiv 10$ where $b = 3$ is his secret key. Now that $Alice$ and Bob have calculated the private key and know they had a shared secret key, they can obtain the shared secret key $10^4 = 4^3 = 18$. Hence, the shared key is $k_1 = 18$, the rest shared key are: $k_2 = 2$, $k_3 = 4$, $k_4 = 16$, $k_5 = 3$, $k_6 = 9$, $k_7 = 12$, $k_8 = 6$, $k_9 = 13$, $k_{10} = 8$, $k_{11} = 18$, $k_{12} = 2$, $k_{13} = 4$, $k_{14} = 16$, $k_{15} = 3$ and $k_{16} = 9$.

The generation of the sixteen keys would make the modified algorithm more secure, as attackers do not have any meaningful parameters and are also unaware of what the first shared secret key is because of the discrete logarithm problem. Hence, the attacker will have a very difficult time locating the keys.

5 Proposed Image Encryption Algorithm

A novel and creative image encryption for images of size 265×256 is presented in this section. algorithm that combines a chaotic map and an 8-dimensional self-invertible matrix. As depicted in Figures 2 and 3 the algorithm has three levels: key generation, image encryption and image decryption level. At a key generation level, a modified Diffie - Hellman key exchange algorithm would be involved in sharing 16 integer numbers, these numbers will be the entries of the first part of the self-invertible matrix. Furthermore, the first and the second shared keys will be the parameters of the logistic map. One permutation round -using a generated self-invertible matrix- and one bitwise XOR operation -using a matrix generated by the logistic map- are used at the encryption level to create a high-quality cipher. The decryption level can undo the encryption and retrieve the plain image using a matrix produced by the logistic map and a self-invertible matrix that was also generated. Compared to other methods currently in use, a proposed algorithm is quick and efficient which will be thoroughly described in the following subsections.

5.1 Key generation level

At this level, two matrices will be created; the first one is self-reversible, while the second one will be constructed using the output of applying the logistics map. These matrices' entries are generated using the modified Diffie-Hellman key exchange algorithm. This level will be demonstrated by the steps below:

Step 1: Alice and Bob apply the Modified Diffie-Hellman key exchange algorithm for generating the key:

$$K = (k_1, k_2, \ldots, k_{16}).$$

Step 2: They will use k_1 and k_2 to generate initial value a_1, a_2 and control parameter α of the logistc map as:

$$\alpha = 3.99 + |a_1 - a_2|,$$

where $a_1 = \dfrac{k_1}{k_1 + k_2}$ and $a_2 = \dfrac{k_2}{k_1 + k_2}$.

Step 3: The logistic map $L_\alpha(x_k$ should apply to generate to a vector L of size $256 \cdot 256 = 65536$ as L as:

$$L = \Big(a_1 = x_1, \quad x_2 = \alpha\,(x_1)(1 - x_1), \quad x_3 = \alpha\,(x_2)(1 - x_2), \ldots, \\ x_{65536} = \alpha\,(x_{65535})(1 - x_{65535})\Big).$$

Step 4: They will apply the Greatest Integer (GI) function [8], then congruent all values *mod* 256 as follows:

$$L_1 = \Big(y_1 = [x_1] \mod 256, \quad y_2 = [x_2] \mod 256, \\ y_3 = [x_3] \mod 256, \ldots, \quad y_{65536} = [x_{65536}] \mod 256\Big).$$

Step 5: They will rewrite the vector L_1 to be a matrix Y of size 256×256 as:

$$Y = \begin{pmatrix} y_1 & y_{257} & y_{513} & \cdots & y_{65281} \\ y_2 & y_{258} & y_{514} & \cdots & y_{65281} \\ y_3 & y_{259} & y_{515} & \cdots & y_{65281} \\ \vdots & \vdots & \vdots & & \vdots \\ y_{256} & y_{512} & y_{768} & \cdots & y_{65535} \end{pmatrix}_{256\times256}.$$

Step 6: They will use the key vector K to generate a self invertible matrix A of size 8×8 as:

Step 6.1: They will rewrite the vector K to be a matrix A_{11} of size 4×4 as:

$$A_{11} = \begin{pmatrix} k_1 & k_5 & k_9 & k_{13} \\ k_2 & k_6 & k_{10} & k_{14} \\ k_3 & k_7 & k_{11} & k_{15} \\ k_4 & k_8 & k_{12} & k_{16} \end{pmatrix}_{4\times4}.$$

Step 6.2: They will produce the matrices (A_{12}), A_{21} and A_{22} by applying the algorithm in section 3 as: $A_{12} = I_4 - A_{11}$, $A_{21} = I_4 + A_{11}$ and $A_{22} = -A_{11}$.

Step 6.3: They will construct the matrix A as:

$$A_{8\times8} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}_{8\times8}.$$

The matrices Y and A will be utilized as the final set of keys.

5.2 Image encryption level

Two operations utilizing the keys Y and A, along with a plain image PI of size 256×256, will produce the cipher image with quite arbitrary values. To begin the encryption level, a sender uses a matrix A to encrypt a plain image PI, an encrypted version of the image is then used in a bitwise XOR operation with the matrix Y. The subsequent steps can be used to carry out these operations:

Step 1: Vectors of size 8×1 will be created from a pixel value of the plain image PI as:

$$(PI_1)_{8\times 1}, (PI_2)_{8\times 1}, \ldots, (PI_{8192})_{8\times 1}.$$

Step 2: Multiply the matrix A by each vector PIs to obtain the set of new vectors of size 8×1 as:

$$(C'_1)_{8\times 1} \mod 256, (C'_2)_{8\times 1} \mod 256, \ldots, (C'_{8192})_{8\times 1} \mod 256.$$

Step 3: A matrix of size 256×256 will be created by rewriting the (C'_i)'s vectors for $i = 1, 2, \ldots, 8192$ as:

$$C' = \begin{pmatrix} (C'_1) & (C'_2) & (C'_3) & \ldots & (C'_{32}) \\ (C'_{33}) & (C'_{34}) & (C'_{35}) & \ldots & (C'_{64}) \\ (C'_{65}) & (C'_{66}) & (C'_{67}) & \ldots & (C'_{96}) \\ \vdots & \vdots & \vdots & & \vdots \\ (C'_{8161}) & (C'_{8162}) & (C'_{8163}) & \ldots & (C'_{8192}) \end{pmatrix}_{256\times 256}.$$

Step 4: A bitwise XOR operation will be performed between the two created matrix Y and C' as:

$$C = Y \oplus_{xor} C'.$$

The matrix C will present the cipher image.

5.3 Image decryption level

Two operations utilizing the keys Y and A, along with a plain image PI of size 256×256, will produce a cipher image with quite arbitrary values. To begin the encryption level, a sender uses matrix A to encrypt a plain image PI, an encrypted version of the image is then used in a bitwise XOR operation with the matrix Y. This level produces an extremely safe cipher picture with uncorrelated pixels. The subsequent steps can be used to carry out these operations:

Step 1: Deriving the matrix Y of size 256×256 by applying a bitwise XOR operation between the two matrices C and Y as:

$$M' = C \oplus_{xor} Y.$$

Step 2: Vectors of size 8×1 will be created from a pixel value of the plain image M' as:

$$(M'_1)_{8\times 1}, (M'_2)_{8\times 1}, \ldots, (M'_{8192})_{8\times 1}.$$

Step 3: Multiply the matrix A by each vector M's to obtain the set of new vectors of size 8×1 as:

$$(M''_1)_{8\times 1}\, mod\, 256, (M''_2)_{8\times 1}\, mod\, 256, \ldots, (M''_{8192})_{8\times 1} \mod 256.$$

Step 4: A matrix of size 256×256 will be created by rewriting the $M''\,(_j)$'s $j = 1, 2, \ldots, 8192$ vectors as:

$$M'' = \begin{pmatrix} (M''_1) & (M''_2) & (M''_3) & \ldots & (M''_{32}) \\ (M''_{33}) & (M''_{34}) & (M''_{35}) & \ldots & (M''_{64}) \\ (M''_{65}) & (M''_{66}) & (M''_{67}) & \ldots & (M''_{96}) \\ \vdots & \vdots & \vdots & & \vdots \\ (M''_{8161}) & (M''_{8162}) & (M''_{8163}) & \ldots & (M''_{8192}) \end{pmatrix}_{256\times 256}.$$

The matrix M'' is the plain image PI.

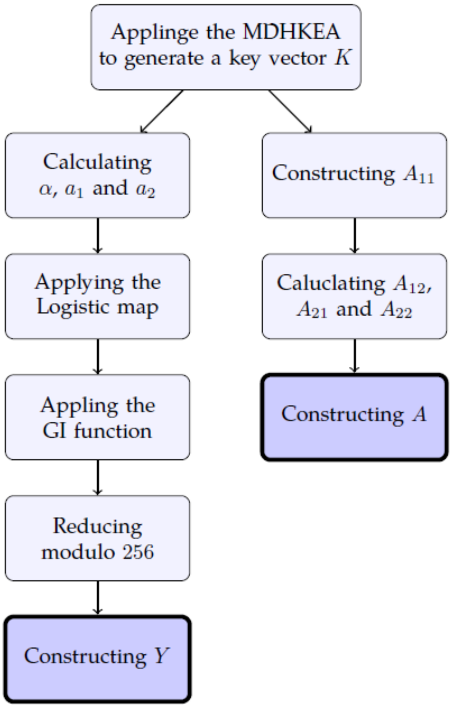

Applinge the MDHKEA to generate a key vector K
Calculating α, a_1 and a_2
Applying the Logistic map
Appling the GI function
Reducing modulo 256
Constructing Y
Constructing A_{11}
Caluclating A_{12}, A_{21} and A_{22}
Constructing A

Figure 2: Framework of key generation level of a proposed image encryption algorithm.

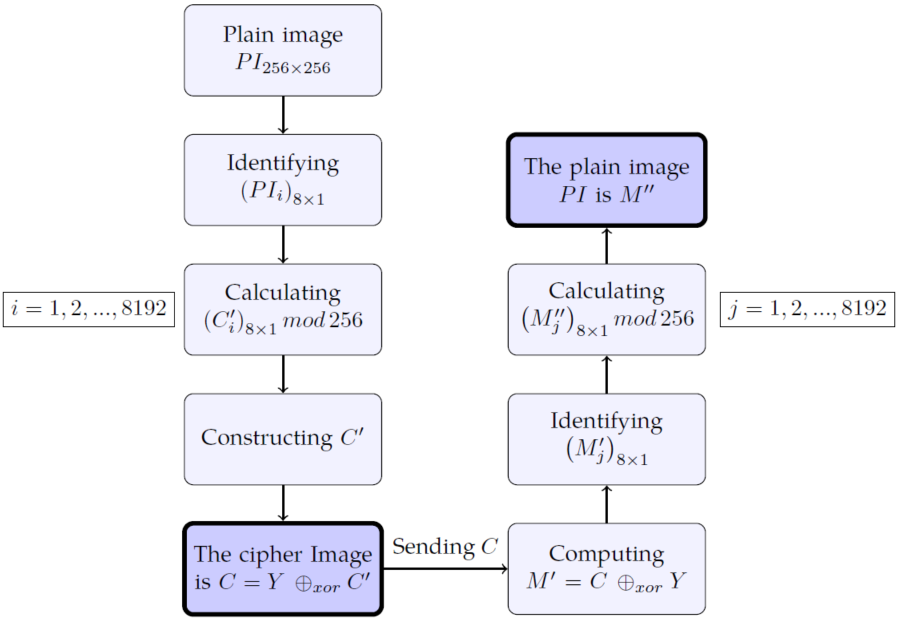

Plain image $PI_{256\times256}$
Identifying $(PI_i)_{8\times1}$
$i = 1, 2, ..., 8192$
Calculating $(C'_i)_{8\times1}\ mod\ 256$
Constructing C'
The cipher Image is $C = Y \oplus_{xor} C'$
Sending C
Computing $M' = C \oplus_{xor} Y$
Identifying $(M'_j)_{8\times1}$
Calculating $(M''_j)_{8\times1}\ mod\ 256$
$j = 1, 2, ..., 8192$
The plain image PI is M''

Figure 3: Framework of image encryption and decryption levels of a proposed image encryption algorithm.

6 Experimental Results and Performance Evaluation

In this section, a series of experiments will be conducted on grey images of size 256×256. Tests will be performed using Matlab $R2023a$ working on a computer with a Windows $11Pro$ operating system with processor: 12^{th} Gen Intel(R) Core(TM) i7-12650H 2.70 GHz and 16 GB RAM. In these experiments, the Baboon, Cat, Goose, Grass, House and Lena images of size 256×256 will be involved as plain images. These images have been used in a number of earlier studies such as [22, 21, 5] to evaluate the level of security of image encryption techniques.

By choosing effective and recent algorithms, and using the results from their published papers, a fair comparison with image encryption techniques based on chaos was made in this study.

6.1 Histogram analysis and chi$-$square test

Similar to how each person's thumbprint is unique, each image's pixel distribution is distinct and does not repeat. In fact, the number of colours in a single channel is related to the 256 levels of grey, by computing the frequency of every single grey level, the histogram will be determined; thus, a histogram is a visual representation based on the size of the image. The flat histogram indicates a uniform distribution of grey levels and prevents any attempt by unauthorized individuals to retrieve any information that may help to know the plain image [6]. For the proposed technique, Figure 4 displays the histograms of the plain images, encrypted images, and decrypted images for a set of selected images. Analyzing an encrypted image will be very challenging because it can be seen that the histogram of an encrypted image is almost uniform and different from the histogram of a plain image. Another tool that is used to measure the monotony of the histogram quantitatively is the chi$-$square test, which can be calculated using the following equation [25]:

$$\chi^2 = \sum_{i=1}^{255} \frac{E_i - Z}{Z},$$

where E_i is the value of the current pixel, and Z is the expected occurrence frequencies of each pixel. For experimental purposes if the value of the Chi$-$square of the encrypted image is lower than 293.2478, then the encrypted image passed the chi$-$square assessment, qucecuntly the histogram of the encrypted image is uniform [5]. From the test results shown in Table 1, all encrypted images - that have been used in this work- have passed the chi-square test. which means that the pixel value distributions of these encrypted images are uniform. Therefore, the proposed encryption algorithm has high security.

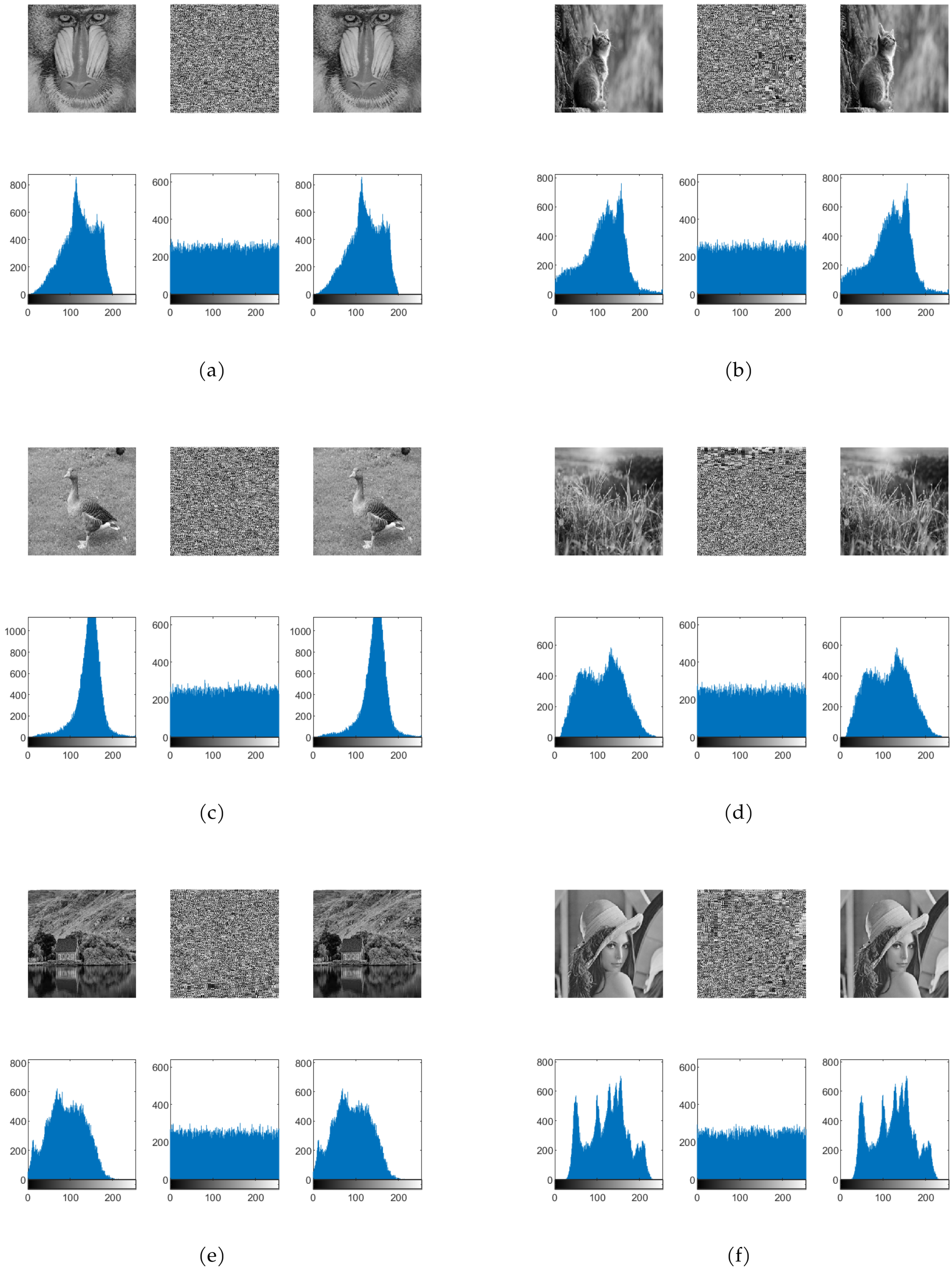


Figure 4: Plain image, encrypted image and decrypted image with their histograms: (a) Baboon, (b) Cameraman, (c) Cat, (d) Goose, (e) Lena, (f) Pepper.

Table 1: Chi−square values of encrypted images.

Images	Chi−square	
	Encrypted Image	Passed?
Baboon	277.3358	Yes
Cat	276.3348	Yes
Goose	281.2432	Yes
Grass	279.5231	Yes
House	275.6521	Yes
Lena	209.5463	Yes

6.2 Correlation analysis

In plain images, there is a strong correlation between adjacent pixels in different directions. Attackers may take advantage of breaking the proposed encryption algorithm by analysing this correlation [25]. In this research, the logistic map implementation will be one approach to removing this correlation. To put it another way, if two images are closely related, the correlation coefficient is close to 1, while the two images are not connected, if the coefficient is close to 0 [17]. The correlation coefficients between the pixels of two images −the plain image and the encrypted image− are assessed in three directions (horizontally, vertically and diagonally). It can be calculated as:

$$c_{xy} = \frac{\frac{1}{n}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y))}{\sqrt{\frac{1}{n}\sum_{i=1}^{N}(x_i - E(x))^2}\sqrt{\frac{1}{n}\sum_{i=1}^{N}(y_i - E(y))^2}},$$

where N is the total number of image pixels, x and y are values of adjacent pixels of the plain image and encrypted image respectively, $E(x)$ and $E(y)$ are their average values [16]. The results c_{xy} of three directions: horizontal, vertical, and diagonal of the proposed encryption algorithm on an implemented image are presented in Table 2. This calculation was done for both of a plain image and the corresponding pixels in its encrypted image. Figure 5 shows the values of the distribution values of c_{xy}. The comparison values of c_{xy} of the proposed encryption algorithm to those of other algorithms for the Lena image are shown in Table 3. The results show the proposed encryption algorithm reduces the correlation between adjacent pixels more effectively than the other algorithms.

Table 2: Correlation coefficient between plain and encrypted images.

Image	Plain Image			Encrypted Image		
	H	V	D	H	V	D
Baboon	0.8741	0.8356	0.7948	0.0049	0.0005	-0.0014
Cat	0.9340	0.9624	0.9336	0.00319	0.2071	0.0230
Goose	0.8017	0.8639	0.7342	0.0035	-0.0035	0.0051
Grass	0.8834	0.9562	0.8521	0.0486	0.0613	0.0266
House	0.8385	0.8565	0.7509	0.0137	0.0301	0.00085
Lena	0.9456	0.9727	0.9214	0.0005	0.0022	0.0001

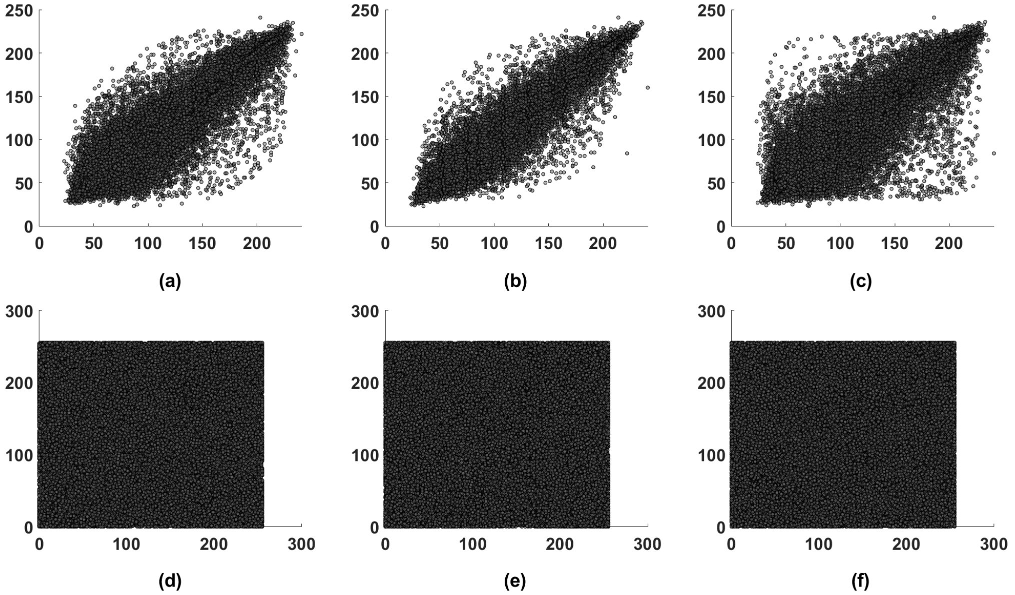


Figure 5: Correlation coefficient of Lena image (a) Horizontal encrypted image, (b) Vertical encrypted image, (c) Diagonal encrypted image, (d) Horizontal plain image, (e) Vertical plain image, and (f) Diagonal plain image.

Table 3: Correlation coefficient of the encrypted Lena image with other algorithms.

Algorithms	Encrypted Images		
	H	V	D
Proposed Algorithm	0.0005	0.0022	0.0001
[22] 2022	-0.0003	-0.0037	0.0020
[21] 2021	-0.0023	-0.0017	-0.0022
[5] 2019	0.0027	0.0012	0.0003
[35] 2018	0.0467	-0.0173	-0.0078
[30] 2018	-0.0029	-0.0017	0.0004

6.3 Differential attack analysis

It is one of the branches of statistics, which has an important role in distinguishing between two encrypted images calculated of two plain images have a difference of one bit in the pixels [3]. The rate of pixel change ($NPCR$) and the unified average change intensity ($UACI$) can be used to determine whether or not the algorithms can withstand this type of attack [31]. It is obvious that a minor difference between the two plain images results in a significant change between cipher images created with the same secret key. The proposed encryption algorithm in this paper proved that it could be repelling various types of attacks by computing $NPCR$ and $UACI$ for selected images, where the perfect values for $NPCR$ and $UACI$ to demonstrate resistance to differential attack are 99% and 33% respectively [25]. The mathematical formula for $NPCR$ and $UACI$ is:

$$NPCR = \sum_{i,j} \frac{R(i,j)}{N \times M} \times 100\%,$$

$$UACI = \frac{1}{N \times M} \sum_{i,j} \frac{En_1(i,j) - En_2(i,j)}{255} \times 100\%.$$

since, En_1 and En_2 are various encrypted images that have been generated using various keys while $D(i,j)$ is 0 or 1 when $C_1(i,j), C_2(i,j)$ are not equal or $C_1(i,j), C_2(i,j)$ are equal respectively. The results of $NPCR$ and $UACI$ for the proposed encryption algorithm on implemented images with a slight change in secret keys of MDHKEA are presented in Table 4. The comparison between the values of $NPCR$ and $UACI$ of a planned encryption algorithm to those of other algorithms for the Lena image is shown in table 5. The results in the tables demonstrate the resistance of the proposed algorithm for differential attack.

6.4 Local shannon entropy (LSE)

The disorder's numerical value for digital image [32] is calculated with a statistical measure called Local Shannon Entropy. This measure is based on evaluating the disturbance of each pixel value in a digital image based on a probability distribution of grey levels of its immediate surroundings [13]. Hence LSE for an image could be calculated as:

$$LSE(d) = -\sum_{i=0}^{P-1} p(d_i) \log_2 p(d_i).$$

where P is a grey level of image, and $p(d_i)$ is a discrete density probability function, d_i is a value of pixel from 0 to 255 [3]. So, the LSE value is higher for pixels with high level unpredictability, while it is lower for pixels with a lower level of randomness. A proposed encryption algorithm was tested on 6 plain images, the results are shown in Table 4. On the other hand Table 5 compares the values of LSA of the proposed encryption algorithm to the values of LSA resulting from other algorithms for Lena. The results indicate that a proposed encryption algorithm provides a high degree of randomness, this is due to the involvement of a logistic map in the encryption level, which contributes to the high randomness in an encrypted image.

Table 4: $NPCR, UACI$ and LSE for encrypted images .

Images	Tests		
	$NPCR$	$UACI$	LSE
Baboon	99.5224	33.2701	7.9971
Cat	99.3759	32.2630	7.9972
Goose	99.5605	33.3352	7.9970
Grass	99.3988	31.8138	7.9968
House	99.3912	33.0179	7.9972
Lena	99.6572	33.4922	7.9975

Table 5: $NPCR, UACI$ and LSE of encrypted Lena with other proposed algorithms.

Algorithms	Tests		
	$NPCR$	$UACI$	LSE
Proposed Algorithm	99.6572	33.4922	7.9975
[22] 2022	99.59	33.27	7.9971
[21] 2021	99.6124	33.4600	7.9971
[5] 2019	99.6368	33.4724	7.9974
[35] 2018	99.61	33.46	7.9970
[30] 2018	99.5986	33.4561	7.9971

6.5 Key security

The private keys used to encrypt and decrypt the information are made up of two significant parts: keyspace and key sensitivity.

The first part (keyspace) must be within specifications capable of repelling attacks directed at the proposed encryption image techniques, so according to the literature, it will be perfect if it is equal to or larger than 2^{128} [12]. In the proposed encryption image, the logistic map parameters are a_1, a_2 and α which there are real numbers. Therefore, if the precision is 10^{-15} will make the key space is $(10^{15})^3 = 10^{45}$, which is greater than 2^{128}. so, in terms of keyspace, the proposed encryption image is capable of repelling attacks.

The second part (key sensitivity) measures the effectiveness of that key in achieving randomness in encrypted images, the unpredictability of finding patterns in the logistic map will ensure this partial. In fact, any modification of the logistic map's parameters will alter the chaotic variables, which in turn will alter the chaotic sequence. In the implementation of the proposed encryption, the key sensitivity will be measured using two keys:

$$K = (k_1, k_2, \dots, k_{16}), \text{ and } K' = (k'_1, k'_2, \dots, k'_{16}),$$

to encrypt Lena image. where $k'_i = k_i + 1$, $i = 1, 2, \dots, 16$. The first encrypted image will be decrypted with $K' = (k'_1, k'_2, \dots, k'_{16})$ and the second encrypted image will be decrypted with $K = (k_1, k_2, \dots, k_{16})$, Figure 6 will show the results of these experiments, this Figure 6 indicates that the proposed encryption image has a high sensitivity if the keys have been changed, which means that the plain image can not be retrieved from the encrypted image. So, the chaotic sequence together with the bitxoring operation has a significant impact on the digital plain which gives an advantage to the proposed encryption algorithm.

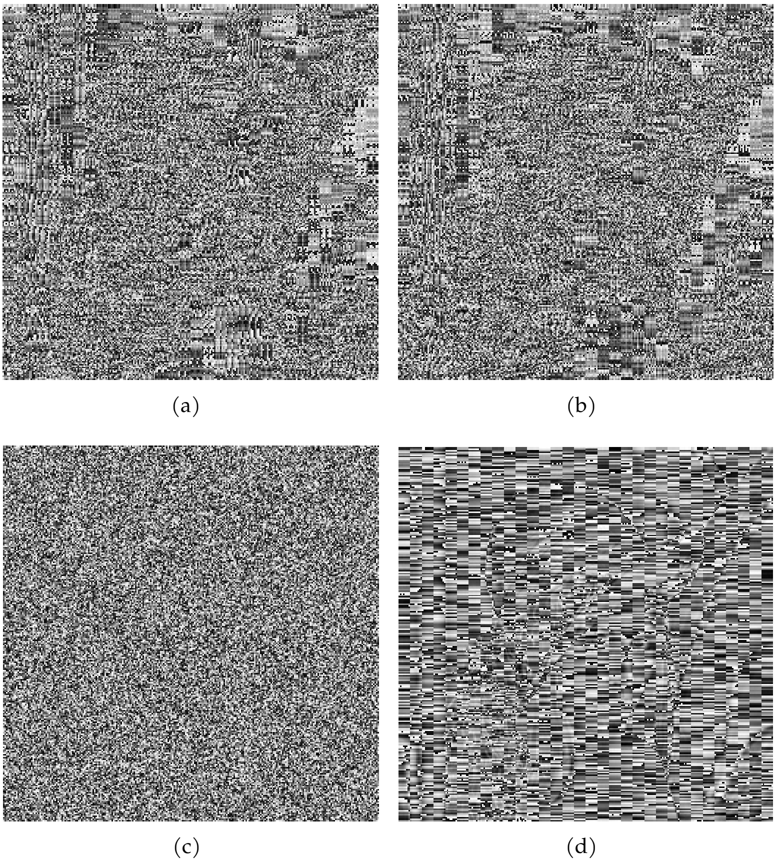


Figure 6: Key sensitivity analysis; (a) Encrypted Lena with K, (b) Encrypted Lena image with K', (c) Decrypt (a) with K', (d) Decrypt (b) by K.

6.6 Implementation time

Calculating the time of implementation of the proposed encryption algorithm is an important measure. Using a large amount of data with a faster algorithm will give the encryption algorithm an advantage. As a result, a high-performance algorithm must consider its security and implementation time. Table 6 shows the results for implementing of proposed encryption algorithm of 6 plain images. According to Table 6, a proposed encryption algorithm is capable of encrypting and decrypting images so fast that the total time for encryption and decryption for all selected images does not exceed the 0.03 seconds, where the implementation time of encryption and decryption for Lena's image was 2.9 Seconds using proposed algorithms in [5], 0.4313 seconds and 0.25 seconds for only encryption level using proposed algorithms in [21] and [3] respectively.

Table 6: Implemention time of selected images.

Algorithms	Implementation time in seconds		
	Encryption	Decryption	Total
Baboon	0.0170	0.0116	0.0286
Cat	0.0160	0.0122	0.0282
Goose	0.0149	0.0087	0.0237
Grass	0.0143	0.0087	0.0231
House	0.0141	0.0094	0.0235
Lena	0.01375	0.0082	0.0219

7 Conclusion and Future Works

Tables 3 and 5 compare performance for many proposed image encryption algorithms using computing the correlation coefficient, LSE, $NPCR$ and $UACI$ respectively. Lena's image was the plain image in all these comparisons. A proposed encryption algorithm had a correlation coefficient is 0.0009, $LSE = 79975$, $NPCR = 99.6572$ and $UACI = 33.4922$. On the other hand, a keyspace is $10^{45} > 2^{128}$ and implementation time for encryption and decryption was 2.9 seconds. The values $NPCR$ and $UACI$ for implementing the proposed algorithm are better than existing algorithms. However, the implementation time of a proposed algorithm is the fastest in comparison with the existing algorithms. In fact, this study has introduced a new image encryption algorithm based on a logistic map and self-invertible matrix. This algorithm modified DHKE algorithm for generating a key vector. At the encryption level, two operations were done (contracting C' and bitwise XOR operation) to generate an encrypted image. The proposed encryption algorithm provides a high level of security and a low implementation time making it capable of repelling attacks. Overall, this work offers a secure and reliable method to transfer digital images.

The researcher may consider studying other sources in the future such as texts, sounds or videos. Another chaotic map could be involved such as Gaussian map [15] or Henon map [14].

Acknowledgement I would like to extend my heartfelt thanks to all those who have assisted me during the course of writing this paper.

Conflicts of Interest The authors declare no conflict of interest.

References

[1] A. A. Abdallah & A. K. Farhan (2022). A new image encryption algorithm based on multi chaotic system. *Iraqi Journal of Science*, *63*(1), 324–337. https://doi.org/10.24996/ijs.2022.63.1.31.

[2] B. Acharya, G. S. Rath, S. K. Patra & S. K. Panigrahy (2007). Novel methods of generating self-invertible matrix for Hill Cipher algorithm. *International Journal of Security*, *1*(1), 14–21.

[3] M. Alawida (2023). A novel chaos-based permutation for image encryption. *Journal of King Saud University-Computer and Information Sciences*, *35*(6), Article ID: 101595. https://doi.org/10.1016/j.jksuci.2023.101595.

[4] J. R. Aparna & S. Ayyappan (2015). Image watermarking using Diffie Hellman key exchange algorithm. *Procedia Computer Science*, *46*, 1684–1691. https://doi.org/10.1016/j.procs.2015.02.109.

[5] A. Arab, M. J. Rostami & B. Ghavami (2019). An image encryption method based on chaos system and AES algorithm. *The Journal of Supercomputing*, *75*, 6663–6682. https://doi.org/10.1007/s11227-019-02878-7.

[6] J. Arif, M. A. Khan, B. Ghaleb, J. Ahmad, A. Munir, U. Rashid & A. Y. Al-Dubai (2022). A novel chaotic permutation-substitution image encryption scheme based on logistic map and random substitution. *IEEE Access*, *10*, 12966–12982. https://doi.org/10.1109/ACCESS.2022.3146792.

[7] S. Arifin, A. Nicholas, Suwarno, H. Baskoroputro, Faisal, A. S. Prabowo, M. A. Ibrahim & A. Rahayu (2023). Algorithm for digital image encryption using multiple hill ciphers, a unimodular matrix, and a logistic map. *International Journal of Intelligent Systems and Applications in Engineering*, *11*(6s), 311–324. https://ijisae.org/index.php/IJISAE/article/view/2858.

[8] D. Burton (2010). *Elementary Number Theory*. McGraw Hill, New York 7th edition.

[9] J. S. Cánovas & H. E. Rezgui (2023). Revisiting the dynamic of q-deformed logistic maps. *Chaos, Solitons & Fractals*, *167*, Article ID: 113040. https://doi.org/10.1016/j.chaos.2022.113040.

[10] M.-F. Danca (2022). Fractional order logistic map: Numerical approach. *Chaos, Solitons & Fractals*, *157*, Article ID: 111851. https://doi.org/10.1016/j.chaos.2022.111851.

[11] W. Diffie & M. E. Hellman (1976). New directions in cryptography. *IEEE Transactions on Information Theory*, *22*(6), 644–654. https://doi.org/10.1109/TIT.1976.1055638.

[12] M. François, T. Grosges, D. Barchiesi & R. Erra (2012). A new image encryption scheme based on a chaotic function. *Signal Processing: Image Communication*, *27*(3), 249–259. https://doi.org/10.1016/j.image.2011.11.003.

[13] U. Hayat, I. Ullah, N. A. Azam & S. Azhar (2022). A novel image encryption scheme based on elliptic curves over finite rings. *Entropy*, *24*(5), Article ID: 571. https://doi.org/10.3390/e24050571.

[14] M. Hénon (1976). A two-dimensional mapping with a strange attractor. *Communications in Mathematical Physics*, *50*(1), 69–77. https://doi.org/10.1007/BF01608556.

[15] R. C. Hilborn (2000). *Chaos and nonlinear dynamics: an introduction for scientists and engineers*. Oxford University Press, United Kingdom 2nd edition.

[16] P. Kavitha & P. Vidhya Saraswathi (2019). Color image encryption: A new public key cryptosystem based on polynomial equation. In *Proceedings of the International Conference on ISMAC in Computational Vision and Bio-Engineering 2018 (ISMAC-CVB)*, pp. 69–78. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-030-00665-5_8.

[17] M. A. A. Khodher & A. Alabaichi (2021). Concealing a secret message in a colour image using an electronic workbench. *Iraqi Journal of Science*, *62*(12), 4964–4977. https://doi.org/10.24996/ijs.2021.62.12.33.

[18] Z. N. Khudhair, A. Nidhal & N. K. El Abbadi (2022). Text multilevel encryption using new key exchange protocol. *Baghdad Science Journal*, *19*(3), 619–630. https://doi.org/10.21123/bsj.2022.19.3.0619.

[19] C. Kumar & P. M. D. R. Vincent (2017). Enhanced diffie-hellman algorithm for reliable key exchange. In *IOP Conference Series: Materials Science and Engineering*, volume 263 pp. Article ID: 042015. https://dx.doi.org/10.1088/1757-899X/263/4/042015.

[20] C. Li, G. Luo & C. Li (2019). An image encryption scheme based on the three-dimensional chaotic logistic map. *International Journal of Network Security*, *21*(1), 22–29. https://doi.org/10.6633/IJNS.201901_21(1).04.

[21] H. Liang, G. Zhang, W. Hou, P. Huang, B. Liu & S. Li (2021). A novel asymmetric hyperchaotic image encryption scheme based on elliptic curve cryptography. *Applied Sciences*, *11*(12), Article ID: 5691. https://doi.org/10.3390/app11125691.

[22] M. Maazouz, A. Toubal, B. Bengherbia, O. Houhou & N. Batel (2022). FPGA implementation of a chaos-based image encryption algorithm. *Journal of King Saud University-Computer and Information Sciences*, *34*(10), 9926–9941. https://doi.org/10.1016/j.jksuci.2021.12.022.

[23] R. M. May (1976). Simple mathematical models with very complicated dynamics. *Nature*, *261*, 459–467. https://doi.org/10.1038/261459a0.

[24] A. J. Menezes, P. C. Van Oorschot & S. A. Vanstone (2018). *Handbook of Applied Cryptography*. CRC Press, United States 1st edition.

[25] A. A. Neamah & A. A. Shukur (2023). A novel conservative chaotic system involved in hyperbolic functions and its application to design an efficient colour image encryption scheme. *Symmetry*, *15*(8), Article ID: 1511. https://doi.org/10.3390/sym15081511.

[26] A. Nitaj (2017). Post quantum cryptography. *Malaysian Journal of Mathematical Sciences*, *11*(S), 1–28.

[27] A. A. Rashid & K. A. Hussein (2023). Image encryption algorithm based on the density and 6D logistic map. *International Journal of Electrical and Computer Engineering*, *13*(2), 1903–1913. http://doi.org/10.11591/ijece.v13i2.pp1903-1913.

[28] H. Stanley & A. Ramachandran (2022). Extended logistic map for encryption of digital images. *International Journal of Nonlinear Sciences and Numerical Simulation*, *23*(7–8), 985–1000. https://doi.org/10.1515/ijnsns-2022-0028.

[29] I. A. Taqi & S. M. Hameed (2018). A new color image encryption based on multi Chaotic Maps. *Iraqi Journal of Science*, *59*(4B), 2117–2127. https://ijs.uobaghdad.edu.iq/index.php/eijs/article/view/451.

[30] X. Wang, X. Zhu & Y. Zhang (2018). An image encryption algorithm based on josephus traversing and mixed chaotic map. *IEEE Access*, *6*, 23733–23746. https://doi.org/10.1109/ACCESS.2018.2805847.

[31] Y. Wu, J. P. Noonan & S. Agaian (2011). NPCR and UACI randomness tests for image encryption. *Cyber journals: multidisciplinary journals in science and technology, Journal of Selected Areas in Telecommunications*, *1*(2), 31–38.

[32] Y. Wu, Y. Zhou, G. Saveriades, S. Agaian, J. P. Noonan & P. Natarajan (2013). Local shannon entropy measure with statistical tests for image randomness. *Information Sciences*, *222*, 323–342. https://doi.org/10.1016/j.ins.2012.07.049.

[33] B. Yang & X. Liao (2018). Some properties of the logistic map over the finite field and its application. *Signal Processing*, *153*, 231–242. https://doi.org/10.1016/j.sigpro.2018.07.011.

[34] B. Yousif, F. Khalifa, A. Makram & A. Takieldeen (2020). A novel image encryption/decryption scheme based on integrating multiple chaotic maps. *AIP Advances*, *10*(7), Article ID: 075220. https://doi.org/10.1063/5.0009225.

[35] Y. Zhang (2018). Test and verification of AES used for image encryption. *3D Research*, *9*(3), 1–27. https://doi.org/10.1007/s13319-017-0154-7.

[36] Z. A. Zukarnain, A. Buhari, N. Z. Harun & R. Khalid (2019). QuCCs: An experimental of quantum key distribution using quantum cryptography and communication simulator. *Malaysian Journal of Mathematical Sciences*, *13*(S), 127–140. https://api.semanticscholar.org/CorpusID:215954342.