A Novel Back Propagated Image Encryption Method with PGP Public Key and 1D Logical Map

Talha Çelik [1], Nurettin Doğan [1,*], Adem Alpaslan Altun [1,*]

Abstract

Encrypted communication plays a significant role in a wide range of areas today, both in protecting personal data and ensuring secure data exchange between governments. Encryption has been used for centuries to protect data generated by humans. In the past, data that needed to be encrypted has evolved from text-based writings to image and sound-based multimedia messages, depending on the development of technology. The high data density and abundance of similar data contained in multimedia messages have been seen to render the classic text-based encryption methods used to encrypt this data time-consuming and insecure in terms of encryption security. Therefore, this study examines recent work on encrypting multimedia messages and proposes a new hybrid encryption method. The proposed method consists of three stages. In the first stage, the rows and columns of the plain image are shifted using the classical Affine encryption algorithm. In this stage, the width and height values of the image are used to create the Affine key. In the second step, the recipient's Pretty Good Privacy (PGP) key is obtained. The resulting PGP key creates the key sequence K to be used in the encryption process. The values are used to obtain the μ value, which is the input parameter of the One-Dimensional (1D) logistic map, and the initial value x_0. In the last step, the XOR operation is applied to the image using the data obtained from the 1D logistic map, and all pixels of the plain image are encrypted. In the proposed work, experiments were conducted on greyscale and RGB images of different sizes. The NPCR value of 99.6078 and the UACI value of 33.5323 demonstrate the applicability of the proposed method.

Keywords: cryptography, chaos, logistic map, image encryption, public key.

Received: 26.10.2025 Accepted: 18.12.2025 Research Article

[1] Computer Engineering Department, Selçuk University, Konya, Türkiye
* Corresponding authors.
✉ nurettin.dogan@selcuk.edu.tr, altun@selcuk.edu.tr

Cite this article as: Çelik, T., Doğan, N., & Altun, A. A. (2026). A novel back propagated image encryption method with PGP public key and 1D logical map. *Journal of Defence and Security Industry: Strategy and Technology 1*, 116-139.

1. Introduction

While rapidly developing and expanding technology has brought convenience, it has also created new privacy issues. Sharing multimedia messages and sending data between individuals has become quite easy today. However, this leads to a violation of personal information privacy. Violation of information privacy can also lead to serious problems in interstate and military communications. Since Julius Seaser, many new methods have been developed for creating and transmitting encrypted messages. In this rapidly developing technological world, multimedia messages have begun to replace text-based messages. Multimedia messages can be both single-frame still images and multi-frame videos. The military devices transmitting these image sequences must encrypt the image during transmission and decrypt it only by the unit managing the military device. These situations are examples of today's information security and encryption needs.

Many new methods have been introduced for use in encryption processes. Some of these are symmetric encryption methods, while others are asymmetric (Althamir et al., 2023). DES, AES, and RSI algorithms are among the most commonly used symmetric encryption methods in today (Diffie, 2022). These methods are quite secure and fast for encrypting text-based messages. However, when used in image encryption, they are both slower than asymmetric encryption methods in terms of encryption speed and inadequate in ensuring the security of the encrypted data (Geetha et al., 2025). Using asymmetric encryption methods in their raw form for image encryption reduces the reliability of the encryption process, and the encrypted image can be decrypted with a number of decryption methods. Considering these circumstances, the need to develop new, more secure and faster methods for encrypting messages containing images or multimedia information has emerged.

RGB and grayscale image contains a high density of data. The similarity between this data and the large number of data to be encrypted are fundamental characteristics of an image. The similarity of the numerical values of two adjacent pixels in an image defines high similarity. The concept of data density and the large number of data to be encrypted describe the number of pixels in the image. An RGB image with 1024 horizontal and 768 vertical pixels requires a total of 2,359,296 data points to be encrypted. Due to these features of the image, encrypting an image with traditional text encryption methods causes both waste of time and reduces reliability because the encryption process takes longer than chaotic methods (Althamir et al., 2023).

Permutation and diffusion methods were developed to eliminate high pixel correlation in images (Karthik et al., 2025). These methods also utilized the randomness provided by the chaos architecture. Chaotic maps are frequently preferred in image encryption due to their unpredictability, sensitive variation depending on the initial condition, and ergodicity (Braverman & Nelson, 2025). Fridrich (1998) first proposed encrypting an NxN image using a two-dimensional (2D) Baker map. Also in that year, Baptista proposed a method for encrypting text using a one-dimensional (1D) logistic map using a chaos architecture.

In the following years, chaotic maps continued to be frequently used for image encryption. However, the initial values used in chaotic maps and the input parameters of the chaotic map were fixed at each encryption step. Consequently, when the chaotic values obtained from the chaotic map iterations are identical, decryption becomes considerably easier by estimating the initial value of the chaotic map. Several new methods have been proposed to eliminate this problem and to dynamically obtain the initial parameters of the chaotic map. Initial values and parameters for chaotic maps have been generated using data from biological generators such as electrocardiography (ECG), voice, iris, fingerprint, and similar data. This makes it difficult to predict the initial values and the result obtained for the first of the chaotic values to be used for encryption.

As an example of these methods for audio data, Lyle et al. (2022) applied the Discrete Fourier Transform (DFT) transform to recorded human voice. They generated the initial value of a 1D logistic map using the values obtained from the DFT analysis. They then encrypted the image by iterating the 1D logistic map 3000 times and applying a diffusion operation on each pixel using the data obtained starting from the 3001st value. The main purpose of advancing the 1D chaotic map 3000 steps after generating the initial chaotic value is to make it difficult to estimate the input parameter μ and the initial value x_0 of the 1D logistic map. In a similar study, Lyle et al. (2022) aimed to increase the initial value of the chaotic map and the unpredictability of its parameters by expanding the key space. They determined the initial values using a pseudo-random generator. They then examined the differences between the Zaslavsky Web Map and the New Adaptive Zaslavsky Web Map using a bifurcation diagram and key space analysis. The New Adaptive Zaslavsky Web Map was concluded to be more secure and usable. It was suggested that this chaotic map could be used in encryption operations (Lyle et al., 2022). This study proposed image encryption using a 1D logistic map for data obtained by digitally scanning the iris layer of the human eye, which allows sufficient light to enter the eye. Similar to other studies, data obtained from digitally scanning the iris layer was used to determine the initial values of the chaotic map (Archana et al., 2022). Archana et al. (2022) proposed encryption using digital data obtained from electrocardiography (ECG). The ECG values used in the encryption process were used to replace the input parameters α, β, and ν of the 3D logistic map. This study proposes to encrypt text-based data such as card numbers instead of image encryption. However, the methods and recommendations used, similar to other studies, were developed to ensure that the initial value and input parameters of the chaotic map are constantly changing (Eldesouky et al., 2022).

Based on the information obtained through literature review and research, this study proposes a new three-step image encryption method. This proposed method aims to eliminate the problem encountered in other studies, which requires constant generation of input parameters and initial values for chaotic maps. This aims to make the encryption process more secure and faster. Furthermore, the PGP public key is used to address the problem of sharing private keys, which is often the case in encryption methods.

In this study, the literature review examines studies on image encryption in the literature. The use of chaotic maps in image encryption was investigated, and the parameter and input value generation methods for chaotic maps were obtained. The back propagated image encryption process proposed in this study is explained in detail in three separate steps in the method section. In the performance analysis section, images encrypted with the proposed method are compared with other studies in the literature. The findings and results are detailed in tables. The final section outlines the contribution of the method developed in this study to the literature, along with its advantages and disadvantages. Considering its potential contribution to future studies, recommendations are made.

1.1. 1D Logistic Map

Chaotic systems are built on mathematical functions that depend on initial conditions and input parameters. Within the limits of the appropriate values used for the input parameter of the chaotic map to exhibit chaotic behaviour, the chaotic map function produces non-repetitive, unpredictable outputs. In this study, a 1D logistic map was used to encrypt RGB and grayscale images. The first use of a 1D logistic map began in 1998. Baptista used a 1D logistic map to encrypt a simple text. In his first study, he succeeded in encrypting and decrypting the word "hi." In the following years, 1D logistic maps were frequently used in image encryption due to their ease of use, reliability, and speed. The 1D logistic map is derived from Equation (3).

Figure 4 shows the Lyapunov exponent (a) and bifurcation graph (b) for a 1D logistic map. As can be seen from this graph, the values within the [3.47, 4.0] range represent the value range in which the 1D logistic map exhibits chaotic behaviour. In developing the algorithm proposed in this study, the range in which the 1D logistic map exhibits chaotic behaviour was taken as the reference for manipulating the input parameter μ of the 1D logistic map. Furthermore, the input parameter μ was generated using the PGP key belonging to the recipient of the encrypted message, making it unique to the recipient.

In this study, the 1024 characters corresponding to the first 2^{13} bits of the PGP public key were used to determine the input parameter μ and the initial value x_0 of the 1D logistic map at the beginning of the encryption process. The decimal values of these characters, found in the ASCII code table, were used for the numerical values. The 1D logistic map was iterated as many times as the number of pixels in the grayscale Lena image, and a key sequence equal to the total number of pixels was obtained. The resulting key sequence was then propagated by applying the XOR operation, with each element corresponding to each pixel in the grayscale Lena image.

1.2. Pretty Good Privacy Key (PGP)

RSA, one of the asymmetric encryption methods, was developed in 1977 at the Massachusetts Institute of Technology (MIT) laboratories (Susanti et al., 2025). In this method, each user has a private and a public key. The sender encrypts a file with the recipient's public key. The recipient decrypts the encrypted message with their private key. The term "key" in this method

refers to the public and private key pair generated by the PGP software. The PGP software, developed by Philip Zimmermann in 1991, was developed by rearranging the public key and private key pair used in the RSA method for file and email encryption (Heinrich, 2025). In this study, the public key generated by the PGP software was generated using the function shown in Equation (4) as the input parameter μ of the chaotic map. The input parameter was generated to be between the values at which the 1D logistic map exhibits chaotic behaviour. The developed algorithms and methods are discussed in detail in the section titled generation of chaotic values. The obtained input parameter μ was used in the iterations of the 1D logistic map.

The keys in the public and private key pairs generated via the PGP software consist entirely of ASCII characters. These keys can be 512, 1024, and 2048 characters long (Heinrich, 2025). The lengths of these key pairs can be increased. However, a longer key does not necessarily imply increased security. While a 1024-character public key requires $3 * 10^{11}$ human years to be decrypted using brute force on modern computers, a 2048-character public key requires $4 * 10^{27}$human years. As can be seen from the obtained values, the time required for a brute-force attack to crack a password is quite long. Therefore, it is preferred due to its performance during both encryption and decryption. Furthermore, the use of 1024-character key pairs have become widespread due to its security against brute-force attacks (Saif, 2025). Studies have been conducted on the commonly used 1024-character PGP public key, taking into account both performance and security criteria. Therefore, in the proposed algorithm, 1024 characters corresponding to the first 2^3 bits of the public key were used in the algorithm used to generate the input parameter of the 1D logistic map.

For example, "mQINBGO133gBEADd" represents the first 16 characters of a public key. The μ parameter generated by the method specified in Equation (4) will be 3.62306202500612 based on the first 16 characters. All μ values obtained are in the range [3.47, 4.0], where the 1D logistic map exhibits chaotic behaviour. The number of digits after the decimal point in the generated values is kept constant at 15 in IEEE floating point standards (Boldo et al., 2023).

1.3. Affine Encryption

Affine encryption is a reversible encryption algorithm that uses modular arithmetic by establishing a relationship between numbers using the greatest common divisor. While it is generally used to manipulate values in text encryption, it is used to manipulate pixel positions in image encryption. Affine is a frequently used encryption method (Ihsan & Doğan, 2023). Unlike single-key encryption methods, Affine encryption uses two-key encryption (Ali et al., 2025). This aims to increase security, but using Affine alone in an encryption process is insecure. Therefore, other methods have been utilized in addition to Affine encryption.

$$E(x) = (ax + b) mod\ m, \quad a, b \in Z, \quad \gcd(a, m) = 1 \tag{1}$$

$$D(y) = a - 1(y - b) mod\ m, \quad a - 1a \equiv 1(mod\ m) \tag{2}$$

$$x, y \in \{0,1,2, \dots, m - 1\}$$

The x value in Equation (1) and Equation (2) represents the value to be used in the encryption process. The $E(x)$ method represents the encryption process. M and N values are representing plain image sizes. In the Affine method, the values a and b are specified as two keys. To determine the value of a, the greatest common divisor must be determined, such that $(a, N) = 1$. No such requirement is required for the other Affine key used, but the value of b must be specified between 0 and $N - 1$.

2. Method

In this study, the method used to encrypt RGB and grayscale Lena images consists of three stages. In the first stage, the a and b values used in the Affine algorithm are determined by the width and height of the image. In this process, the a value of the Affine method is calculated by decreasing the image width from halfway up to $(a, W) = 1$. For example, for an image with a width of 1024 pixels, $N = 1024$ and $a = n/2 - step_n$. Based on this example, a value will be 511 and the b value will be 256. After finding the Affine keys, row and column shifting operations are applied to the Lena image. In the second stage, the first 1024 characters of the PGP public key are converted to numerical values to produce chaotic values in the 1D logistic map and processed with the normalization function in Equation (4). This normalization function normalizes the numerical values obtained from the PGP public key between the values $[3.47, 4.0]$, which is the region where the 1D logistic map will exhibit chaotic behaviour. Then, the appropriate input parameters and initial values for the 1D logistic map are obtained. The 1D logistic map is iterated as many times as the number of pixels in the image to be encrypted. Each chaotic value obtained at this stage is transferred to the key array K, which will be used in the diffusion process. In the third and final stage, a dynamic diffusion operation is applied using the XOR operator to each element in the key array K, each pixel in the gray-level Lena image (P_i), and the pixel value in the previous encrypted gray-level Lena image (C_i). Thus, the last encrypted pixel value establishes a close connection with the first pixel value in the image.

In the first step, the a and b values generated using the affine algorithm were used to perform the row and column shifting operation. The a and b values used in the Affine algorithm are explained in detail in the section affine encryption step.

In the second stage, the input parameter and initial value of the 1D logistic map are generated using the PGP public key. The input parameter and initial value generated using the PGP public key are within the region where the 1D logistic map exhibits chaotic behaviour. Therefore, the data obtained from the chaotic map is unpredictable. Next, a sequence of K keys, corresponding to the number of pixels in the Lena image, is generated using the 1D logistic map function Equation (3) and the K key generation function Equation (5). The K key sequence contains the chaotic outputs generated by the 1D logistic map. These values are stored for use in the next stage, the diffusion stage.

$$x_{n+1} = f(x_n) = b\, x_n(1 - x_n), [b = \mu]\ in\ [3.47,4] \tag{3}$$

In the third and final step, a dynamic XOR operation is applied between each element (K_i) of the key sequence K obtained from the iterations of the chaotic map and each pixel value (P_i) of the gray level Lena image. To increase the reliability of the encryption algorithm during the propagation phase and to obtain a stronger encryption method, the result of the previous XOR operation C_{i-1} is included in the next operation. This operation is shown in Equation (9). In this operation, because the result of the previous XOR value is obtained, the previous value is not generated in the first iteration. To eliminate this problem, the C_0 value in the first iteration is fixed to 1. The other C_i values are determined by the XOR operation and used in the next calculation. The diagram of the method developed within the scope of this study is shown in Figure 1.

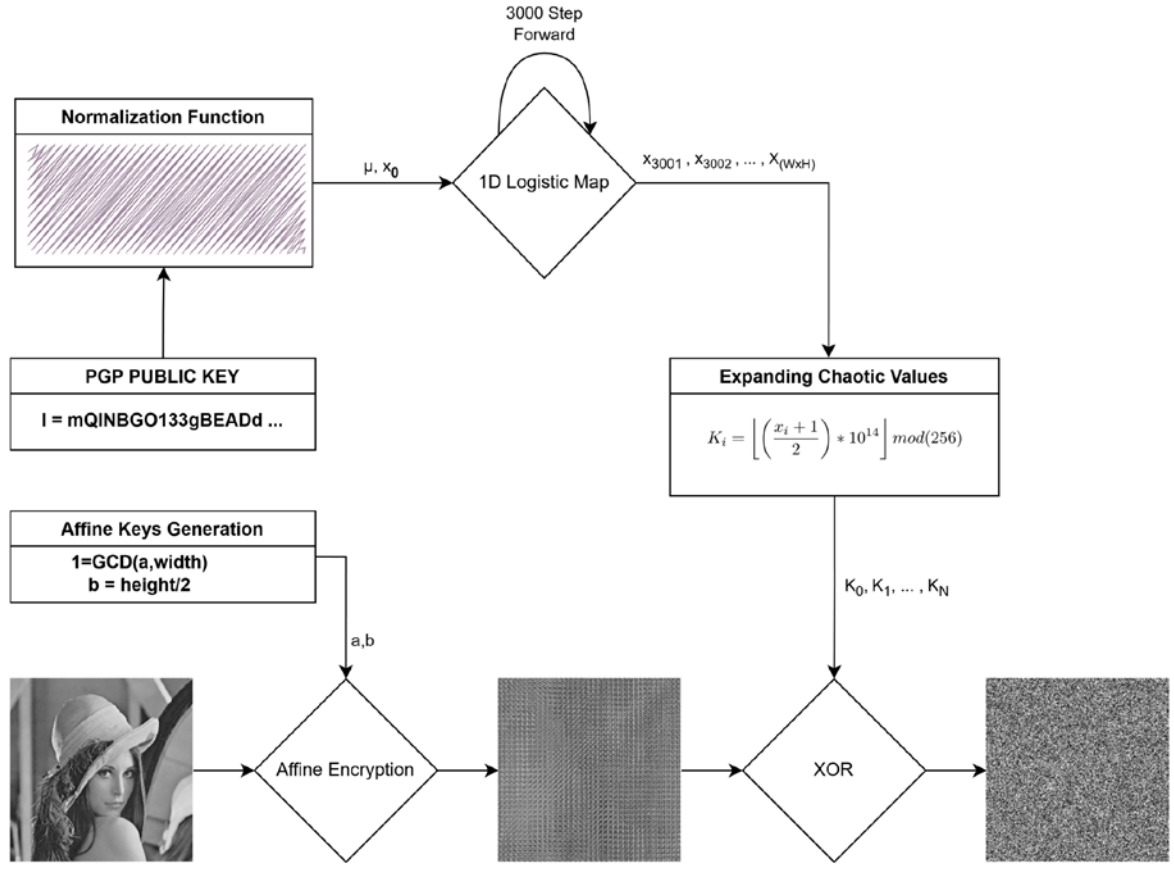

Figure 1. Diagram of the proposed method.

2.1. Affine Encryption Step

In the affine method, two keys are selected, a and b. These keys should be determined such that Greatest Common Divisor (GCD) $(a, N) = 1$ for a key space of length N. The Euclidean division algorithm is used to determine this key. This algorithm generates the value a by calculating $1 = \gcd(a, N)$. The other key, b, can be chosen to lie between the values $[0, N]$. In the proposed algorithm, the value b is set to $N/2$. $E(x)$ represents the encryption function, and $D(y)$ represents the decryption function, shown in the Equation (1) and Equation (2).

A 256x256 pixel grayscale Lena image, in which row and column shifting was applied using the Affine method, is shown in Figure 2. In this image, a = 511, b = 256 was set for the Affine encryption algorithm. As can be seen from the encrypted image (c), its pixels are not homogeneously spread. Therefore, the encrypted image obtained by using the Affine encryption algorithm alone can be easily decrypted using frequency analysis and similar methods. Therefore, in this study, in addition to the Affine method, a diffusion process was also used. A 1D chaotic map was utilized to ensure that the values used in the diffusion process were unpredictable. Thus, the reliability and encryption performance of the system were increased. As a result of the experiments, the encrypted images obtained with this method were examined under the reliability section. The Euclidean function used to determine the a and b parameters of the Affine method is shown in Algorithm 1. The K_a value obtained with the Euclidean function was used in the row and column shifting step.

Algorithm 1: An algorithm of Affine encryption method

Input: : $Image, Width, Height$

Output: : $PixelsShiftedImage\ (P)$

Step-1 : r ← $Width$ mod $Height$

Step-2 : **while** $r \neq 0$ **do**

a ← b

b ← r

r ← a mod b

Step-3 : **for** $i = 0$ to $Width$ **do**

$P_{[j]} \leftarrow (i * r + \frac{Width}{2})$ mod $Width$

Step-4 : **for** j= 0 to $Height$ **do**

$P_{[j]} \leftarrow (j * r + \frac{Height}{2})$ mod $Height$

Step-5 : return P

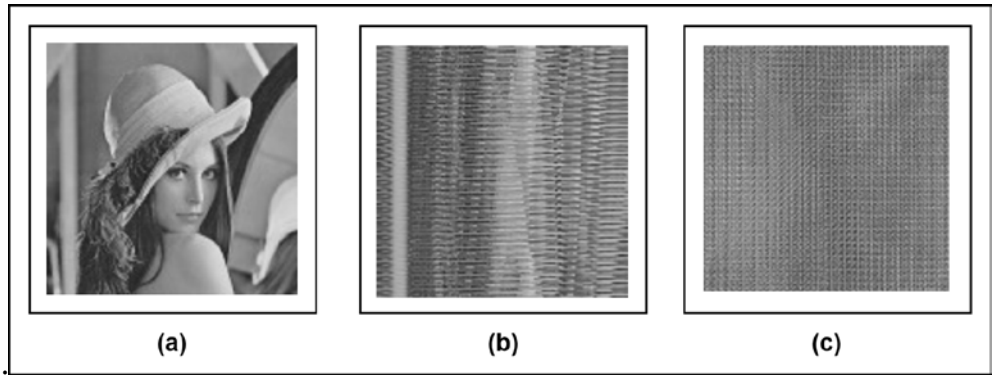

Figure 2. (a) Encryption of gray level Lena image using affine method. Original Lena image, (b) Row shifting with affine algorithm, and (c) Column shifting with affine algorithm.

2.2. Generation of Chaotic Values

This step explains how to generate the input parameter and initial value of the 1D logistic map used in the study using the PGP public key, and then how to prepare the chaotic outputs obtained from the 1D logistic map for use in the next diffusion phase.

First, the PGP public key of the recipient to whom the encrypted image will be sent is obtained. This public key, as described above, is published by the recipient in a publicly accessible format. The sender encrypts the message with the recipient's public key and sends it to the recipient. The recipient decrypts this encrypted message with the private key in their own public and private key pair. In this way, the sender encrypts and sends the message in a way that only the recipient can decipher. This method, used today especially for email and file encryption, eliminates the problem of key sharing. However, as explained in previous sections, this method, called asymmetric encryption, is quite successful in text and file encryption, but it fails in image encryption. However, it is quite successful in key generation and sharing. Thanks to this key sharing advantage, the encryption key, called the secret in image encryption, has become very efficient in generating the input parameters and initial values of the chaotic map.

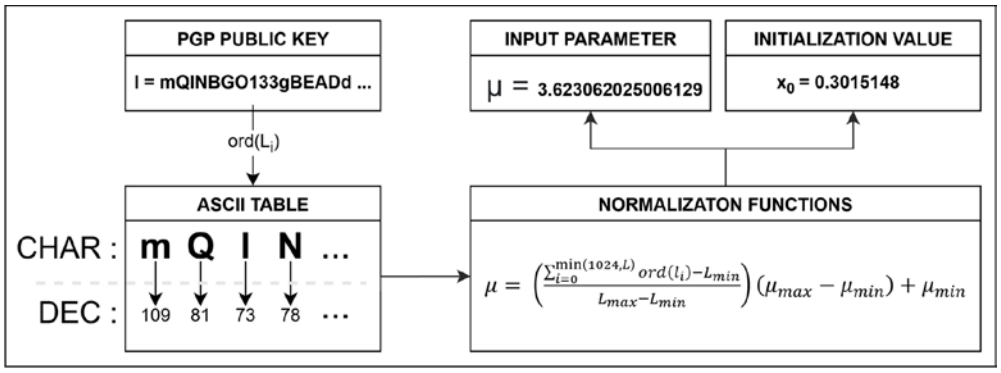

Figure 3. Explaining how to obtain from the PGP public key to 1D logistic map's the μ parameter and the x_0 initialization value.

In this study, the PGP public key of the person to whom the encrypted image will be sent is first obtained. This public key must be at least 1024 characters long. A PGP public key shorter than 1024 characters will reduce the sensitivity of the μ parameter and the initial value of x_0. This will make these values easier to estimate. Therefore, to ensure system security, it was concluded that the PGP public key must be at least 1024 characters long. The remaining characters in the PGP public key exceeding this value are not included in the calculation, as 1024 characters provide sufficient precision. The first 1024 characters of the PGP public key are expressed as decimal numbers using the ASCII code table. This provides numerical values that can be used in the normalization function. The Equation (4) generates the initial value of the μ parameter and x_0. The steps for generating the input parameter and initial value are shown in the diagram in Figure 3. The normalization function is specified in Equation (5) and Algorithm 2.

The value μ specified in the Equation (4) represents the input parameter of the 1D logistic map, the value L represents the character length of the PGP key, and the values L_{min}, L_{max} represent the minimum and maximum character lengths that the PGP key can hold. μ_{min}, μ_{max} represent the value range $[3.47, 4.0]$ in which the 1D logistic map exhibits chaotic behaviour. The $ord(l_i)$ function is used to convert the ASCII characters in the PGP key to decimal values. This function returns the decimal values of the characters specified in the ASCII character table.

After determining the input parameters and initial values of the 1D logistic map, the key array K to be used in the diffusion step was generated by generating the outputs x_i after the first 3000 iterations of the 1D logistic map, as specified in Equation (5). The element length of the key array K is equal to the number of pixels in the image to be encrypted. Thus, the XOR operation was applied such that each element K_i corresponds to the pixel value P_i in the image to be encrypted.

$$\mu = \left(\frac{\sum_{i=0}^{\min(1024,L)} ord(l_i) - L_{min}}{L_{max} - L_{min}}\right)(\mu_{max} - \mu_{min}) + \mu_{min} \quad (4)$$

$$K_i = \left\lfloor \left(\frac{x_i + 1}{2}\right) * 10^{14} \right\rfloor mod(256) \quad (5)$$

Figure 4 illustrates the parameter regions of the 1D logistic map, highlighting the interval in which chaotic dynamics emerge and providing the basis for selecting the control parameter μ in the proposed encryption scheme.

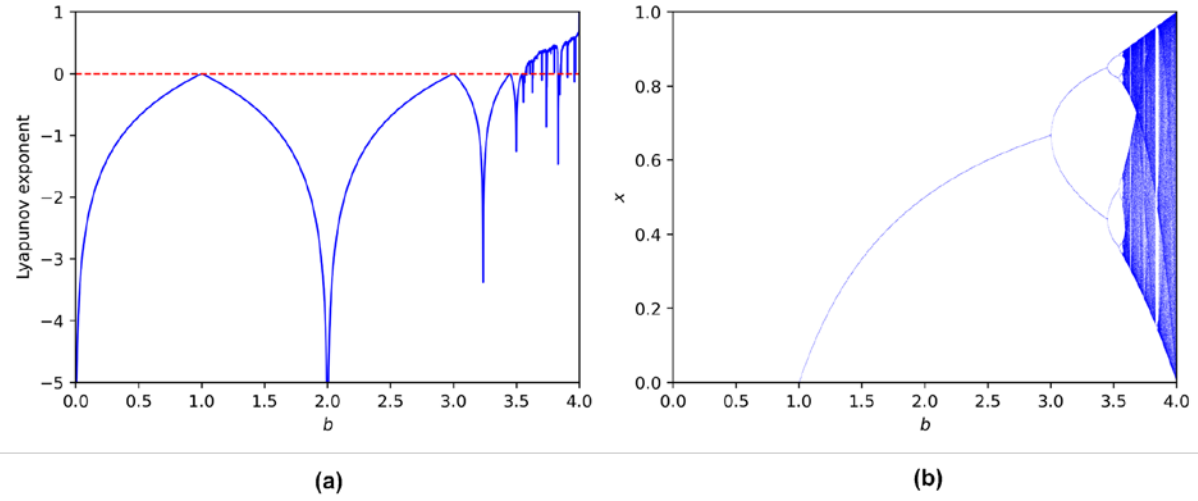

Figure 4. (a) The 1D logistic map Lyapunov exponent and (b) Bifurcation diagram.

2.3. Diffusion And Back Propagation Step

The XOR bit operator compares two bits. It is a logical operator that returns a value of one if the compared bits are different and a value of zero if they are the same. It is used in image encryption and similar tasks requiring bit operators (Gupta et al., 2023). The XOR operator is preferred in image encryption because it can produce unpredictable and random results when used with chaotic map data, making it difficult for unauthorized users to decipher the encrypted data. Furthermore, the XOR operation is computationally efficient and can be implemented on a variety of platforms. This property of the XOR bit operator has made it a popular choice for image encryption (Artuğer, 2025).

The use of the XOR operator in the diffusion and feedback step aims to increase encryption strength and processing speed. The NPCR and UACI values obtained from other studies on image encryption are compared in Tables 4 and 5. Within the scope of this study, it was observed that the proposed method resulted in an average increase of 0.01056 in the NPCR value on the Lena grey level image. Similarly, the results obtained for the UACI values indicated in Table 5 showed an average increase of 0.05385.

In this work, the XOR operator aims to encrypt the pixel values of the entire image one by one. To this end, the chaotic sequence of numbers generated by the 1D logistic map is converted into a K key array using the Equation (9). Each element of the resulting K key array is processed with each pixel on the RGB and grayscale Lena image using the XOR bit operator. This process creates a new encrypted image that can only be decrypted with the original K key array. Using the Equation (6) the two-dimensional matrix structure of dimension MxN in the two-dimensional image is converted into a 1D array for ease of processing and for matching with the 1D K key array. Here, P_i represents the two-dimensional image reduced to one dimension. The i and j values represent the row and column indices of the two-dimensional image, and the W value represents the width of the two-dimensional image.

$$P_i = iW + j, \quad i,j \in \mathbb{Z}, W \in \mathbb{R}, \tag{6}$$

$$C_i = P_i \oplus K_i \oplus C(i), \quad K_i \in \mathbb{Z}_2^n \tag{7}$$

$$C(i) = \begin{cases} i-1 & i > 0, \\ 1 & i \leq 0, \end{cases} \tag{8}$$

$$\Rightarrow C(i) = (iW + j) \oplus K_i \oplus \begin{cases} i-1 & i > 0, \\ 1 & i \leq 0, \end{cases} \tag{9}$$

Algorithm 2: Chaotic values generation and encryption process

Input: : $PGPKey, L_{min}, L_{max}, \mu_{min}, \mu_{max}, PixelsShiftedImage\ (P)$

Output:: $EncryptedImage$

Step-1 : **for** i ← 0 to $PGPKey$ **do**

$l = l + ord(PGPKey_i)$

Step-2 : $\mu \leftarrow \left[\frac{(l - L_{min}) * (\mu_{max} - \mu_{min})}{(L_{max} - L_{min}) + \mu_{min}}\right]$

Step-3 : **for** i ← 0 to 3000 **do**

$x_i \leftarrow \mu * x_0\ (1 - x_i)$

Step-4 : **for** i ← 1 to $Widht * Height + 1$ **do**

$x_{i-1} \leftarrow \mu * x_0\ (1 - x_{i-1})$

$K_{i-1} \leftarrow \left(\frac{x_{i-1} + 1}{2}\right) * 10^{14}\ mod\ 256$

$CipherImage_{i-1} \leftarrow P_i \oplus K_i \oplus P_{i-1}$

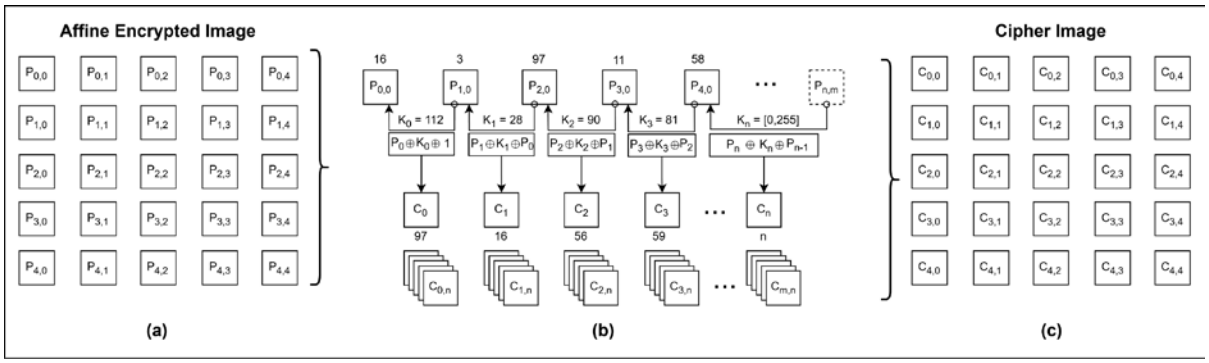

Figure 1. (a) Gray level Affine encrypted image pixels, (b) Application of the proposed method on pixel values, and (c) Cipher image pixels.

3. Results

In this study, the proposed algorithm was developed in the Python 3.8 programming language, and the performance results obtained from the experiments are reported for comparison with other studies. The images used in the experiments were taken from the SIPI database (California, 2024), and the width and height values of the images are shown in Table 1. This section presents the literature performance metrics that test the effectiveness of the proposed algorithm, including numerical results and comparisons. The test results and comparisons are explained in the tables and visual outputs in this section. The results obtained from the experiments conducted with the proposed algorithm demonstrate a high degree of accuracy compared to other studies. The results also demonstrate that the algorithm can accurately encode and decode images without compromising image quality.

Table 1. Images and attributes used in the study.

Filename	Image	Size	Colour
-	Lena	256x256	Gray
5.1.12	Clock	256x256	Gray
5.1.14	Chemical plant	256x256	Gray
boat.512	Boat	512x512	Gray
gray21.512	Gray	512x512	Gray
5.2.10	Stream and Bridge	512x512	Gray
7.1.03	Tank	512x512	Gray
5.3.01	Man	1024x1024	Gray
5.3.02	Airport	1024x1024	Gray
-	Lena	256x256	RGB
5.1.11	Plane	512x512	RGB
4.2.07	Peppers	512x512	RGB
house	Hause	512x512	RGB
4.2.03	Baboon	512x512	RGB

3.1. Histogram Analysis

An image histogram shows the pixel distribution in the image. The values on the horizontal axis in the graph represent the colour values in the image. The values on the vertical axis indicate the number of pixel values corresponding to the horizontal axis. When the pixel density distribution in the image reaches the same average number for each pixel value, the image histogram becomes equal. This makes it more difficult for attackers to decipher the encrypted image using statistical attack methods.

Attackers attempt to decrypt text or images using frequency analysis. In images where the pixel distribution is not homogeneous, attackers can decrypt image by performing frequency analysis and similar attacks. Figure 6 shows the histograms of the plain image and the encrypted image with the algorithm proposed in this paper. The results show that the proposed algorithm distributes the pixels of the image evenly.

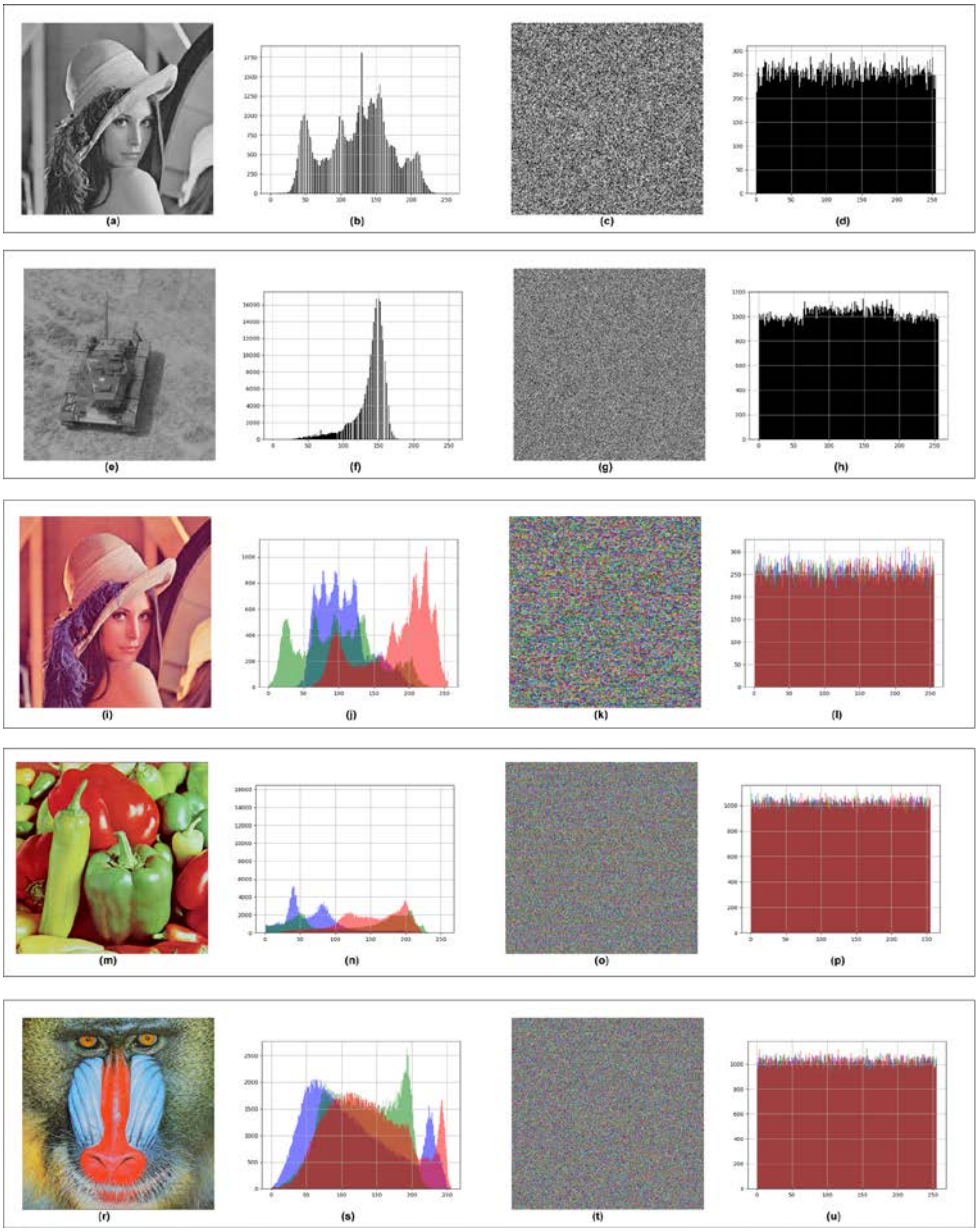

Figure 2. (a) Gray Lena image, (b) Gray Lena image histogram graph, (c) Encrypted Lena image with proposed method, (d) Encrypted gray Lena image with proposed method histogram graph, (e) Gray Tank image, (f) Gray Tank image histogram graph, (g) Encrypted Tank image with proposed method, (h) Encrypted gray Tank image with proposed method histogram graph, (i) RGB Lena image, (j) RGB Lena image histogram graph, (k) Encrypted Lena image with proposed method, (l) Encrypted RGB Lena image with proposed method histogram graph, (m) RGB Peppers image, (n) RGB Peppers image histogram graph, (o) Encrypted Peppers image with proposed method, (p) Encrypted RGB Peppers image with proposed method histogram graph, (r) RGB Baboon image, (s) RGB Baboon image histogram graph, (t) Encrypted Baboon image with proposed method, and (u) Encrypted RGB Baboon image with proposed method histogram graph.

3.2. Entropy Analysis

Entropy is used to measure the randomness of data in a dataset. The random distribution of pixel values in an image result in a high entropy value for the image. In the literature, the optimal entropy value for values consisting of 8 bits, i.e., 2^8 bits, is determined as 8. An entropy value smaller than this indicates that the randomness in the image is low. A low entropy value indicates that the encryption algorithm is not reliable. (Paramesha et al., 2025). In this study, the Shannon entropy calculation method was used, and the obtained data is shown in Table 2. They are compared with other studies in Table 2 According to this table, the obtained entropy values show better performance than other studies.

$$H(X) = -\sum_{i=0}^{L-1} p(x_i) \log_2 p\,(x_i) \tag{9}$$

Table 2. Comparison of the entropy values of the proposed method with other studies.

Studies	Lena	Boat	4.2.07
Proposed	7.9669	7.9675	7.9678
(Hanif et al., 2022)	7.9957	7.9955	-
(Zhou et al., 2022)	7.9977	-	-
(Taqi & Abdul-Haleem, 2023)	7.9920	-	-
(Ullah et al., 2022)	7.9960	-	7.9990
(Wang, 2024)	7.9960	-	-

3.3. Image Sensitivity Analysis

Differential attack analysis is used to determine changes in the encrypted image and the sensitivity of the algorithm. This method requires two images. The first image should be the original, and the second image should be a modified version of the original image's pixel values. Both images are encrypted with the proposed algorithm, and the difference between the resulting encrypted images indicates the algorithm's sensitivity. For these tests, the number of pixels change rate (NPCR) and unified average changing intensity (UACI) are used. The NPCR value should be as close to 100% as possible, and the UACI value should converge to 33.33. These analysis methods are calculated as follows Equation (10), Equation (11), and Equation (12).

$$NPCR(C, C') = \frac{1}{W * H} \sum_{i=0}^{W} \sum_{j=0}^{H} D\left(C_{ij}, C'_{ij}\right) \tag{10}$$

$$D\left(C_{ij}, C'_{ij}\right) = \begin{cases} 1 & if\, C_{ij} = C'_{ij} \\ 0 & if\, C_{ij} \neq C'_{ij} \end{cases} \tag{11}$$

$$UACI(C, C') = \frac{1}{W * H * F} \sum_{i=0}^{W} \sum_{j=0}^{H} | C_{ij} - C'_{ij}| \quad (12)$$

The C, C'values shown in the Equation (10), Equation (11), and Equation (12) represent the image encrypted with the proposed algorithm and the modified version of a pixel in the image. The W, H values represent the width and height of the image, and F represents the maximum possible pixel value.

The NPCR and UACI values obtained in experiments conducted with the proposed algorithm are shown in Table 2. Tables 3 and 4 compare the NPCR and UACI values of the proposed algorithm with those of other studies.

Table 3. NPCR and UACI results of the proposed algorithm.

Filename	Image	NPCR	UACI	Entropy
-	Lena	99.6078	33.4003	7.9669
5.1.12	Clock	99.6048	33.4563	7.9664
5.1.14	Chemical plant	99.6124	33.5052	7.9672
boat.512	Boat	99.6189	33.6229	7.9675
gray21.512	Gray	99.5872	33.4152	7.9661
5.2.10	Stream and Bridge	99.5895	33.4251	7.9663
7.1.03	Tank	99.6094	33.3137	7.9668
5.3.01	Man	99.6177	33.4804	7.9674
5.3.02	Airport	99.6044	33.4567	7.9666
-	Lena (RGB)	99.6158	33.4600	7.9679
5.1.11	Plane (RGB)	99.5952	33.4531	7.9676
4.2.07	Peppers (RGB)	99.6105	33.4582	7.9678
hause	Hause (RGB)	99.6082	33.4574	7.9677
4.2.03	Baboon (RGB)	99.6128	33.4590	7.9681

Table 4. Comparison of NPCR values.

Studies	Lena	Boat	5.1.14	7.1.03	5.3.01	4.2.07
Proposed	99.6078	99.6189	99.6124	99.6094	99.6177	99.6105
(Hanif et al., 2022)	99.5743	99.5987	-	-	-	-
(Li et al., 2023)	-	99.6040	99.6078	99.5972	-	-
(Feng et al., 2022)	-	99.6168	99.6075	99.6281	-	-
(Zhou et al., 2022)	99.6000	-	-	99.6300	99.6000	-
(Taqi & Abdul-Haleem, 2023)	99.6078	-	-	-	-	-
(Ullah et al., 2022)	-	-	-	-	-	99.6000
(Wang, 2024)	99.6050	-	-	-	-	-
(Zhang, 2024)	99.5991	-	-	-	-	-

Table 5. Comparison of UACI values.

Studies	Lena	Boat	5.1.14	7.1.03	5.3.01	4.2.07
Proposed	33.4003	33.6229	33.5052	33.3137	33.4804	33.4582
(Hanif et al., 2022)	33.0509	33.2262	-	-	-	-
(Li et al., 2023)	-	33.4814	33.4086	33.4773	-	-
(Feng et al., 2022)	-	33.4590	33.4667	33.5650	-	-
(Zhou et al., 2022)	33.4500	-	-	33.4200	33.4700	-
(Ullah et al., 2022)	-	-	-	-	-	33.3800
(Wang, 2024)	33.4820	-	-	-	-	-
(Zhang, 2024)	33.4029	-	-	-	-	-

3.4. Cropping and Noise Attack Analysis

Images can experience corruption along the transmission line as they are transmitted. These corruptions can stem from technical problems in the transmission line, and malicious attackers can manipulate the content of the message. This method employed by attackers is called man-in-the-middle (MITM) (Sivasankari & Kamalakkannan, 2022). In this attack, an attacker can intrude between the sender and receiver, view the message, and even manipulate the message content to mislead the recipient. In addition to the attackers themselves, technical problems in the transmission line can also cause corruption in the message. These corruptions typically manifest in the images in the form of salt and pepper noise, Gaussian noise, and changes in image size or pixel values (Yu, 2023).

This paper has conducted experiments using the proposed method. The results of the experiments are shown in Figure 7 and 8. As can be seen from the results, the encrypted image could be decrypted even in cases where more than 50% of the image is distorted.

3.5. Key Space Analysis

In an encryption process, the width of the key space indicates whether it is resistant to brute-force attacks. For the algorithm to be resistant to brute-force attacks, the key space is expected to be greater than 2^{100} (Abba et al., 2024). The algorithm proposed in this article uses the Affine encryption method in the first stage. The key space of this method consists of the values a and b. Each value a can take approximately 128 different values, while the value b can take 256 different values. Therefore, the key space width for the Affine encryption system will be 256*128 = 2^{15}.

In the diffusion step, the encryption process is manipulated with separate K keys for each pixel of the image. Therefore, for a $256x256$ image, the key space value will be 256! Consequently, since the key space width used in this algorithm will be $2^{100} * (2^8)! > 2^{100}$, the proposed algorithm is resistant to brute force attacks.

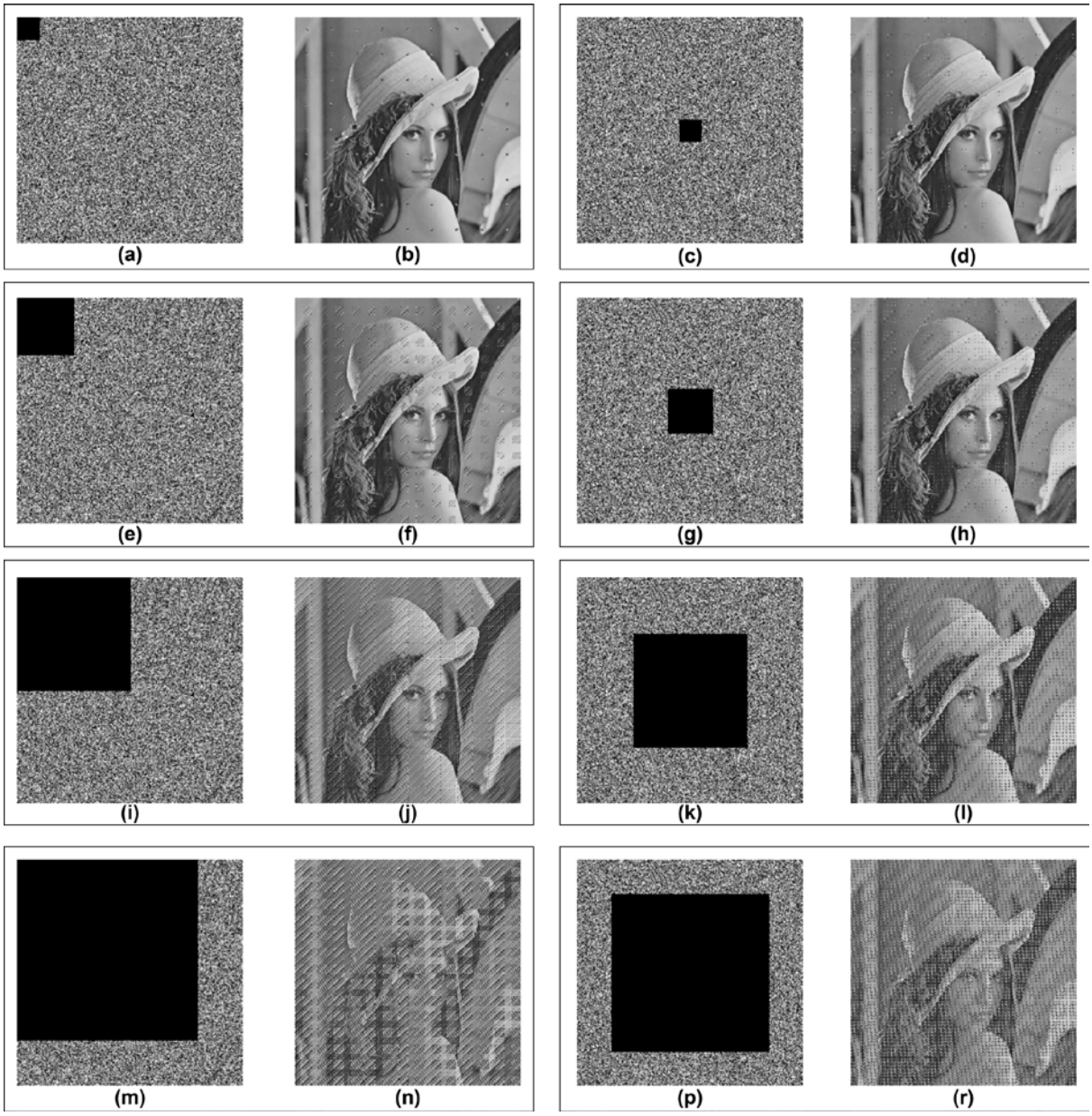

Figure 3. (a) 10% Cropped from left upside encrypted gray Lena image, (b) Decrypted image of the encrypted image with 10% cropped from the top left using the proposed method, (c) 10% Cropped from centre encrypted gray Lena image, (d) Decrypted image of the encrypted image with 10% cropped from the centre using the proposed method, (e) 20% Cropped from left upside encrypted gray Lena image, (f) Decrypted image of the encrypted image with 20% cropped from the top left using the proposed method, (g) 20% Cropped from centre encrypted gray Lena image, (h) Decrypted image of the encrypted image with 20% cropped from the centre using the proposed method, (i) 50% Cropped from left upside encrypted gray Lena image, (j) Decrypted image of the encrypted image with 50% cropped from the top left using the proposed method, (k) 50% Cropped from centre encrypted gray Lena image, (l) Decrypted image of the encrypted image with 50% cropped from the centre using the proposed method, (m) 70% Cropped from left upside encrypted gray Lena image, (n) Decrypted image of the

encrypted image with 70% cropped from the top left using the proposed method, (p) 70% Cropped from centre encrypted gray Lena image, and (r) Decrypted image of the encrypted image with 70% cropped from the centre using the proposed method.

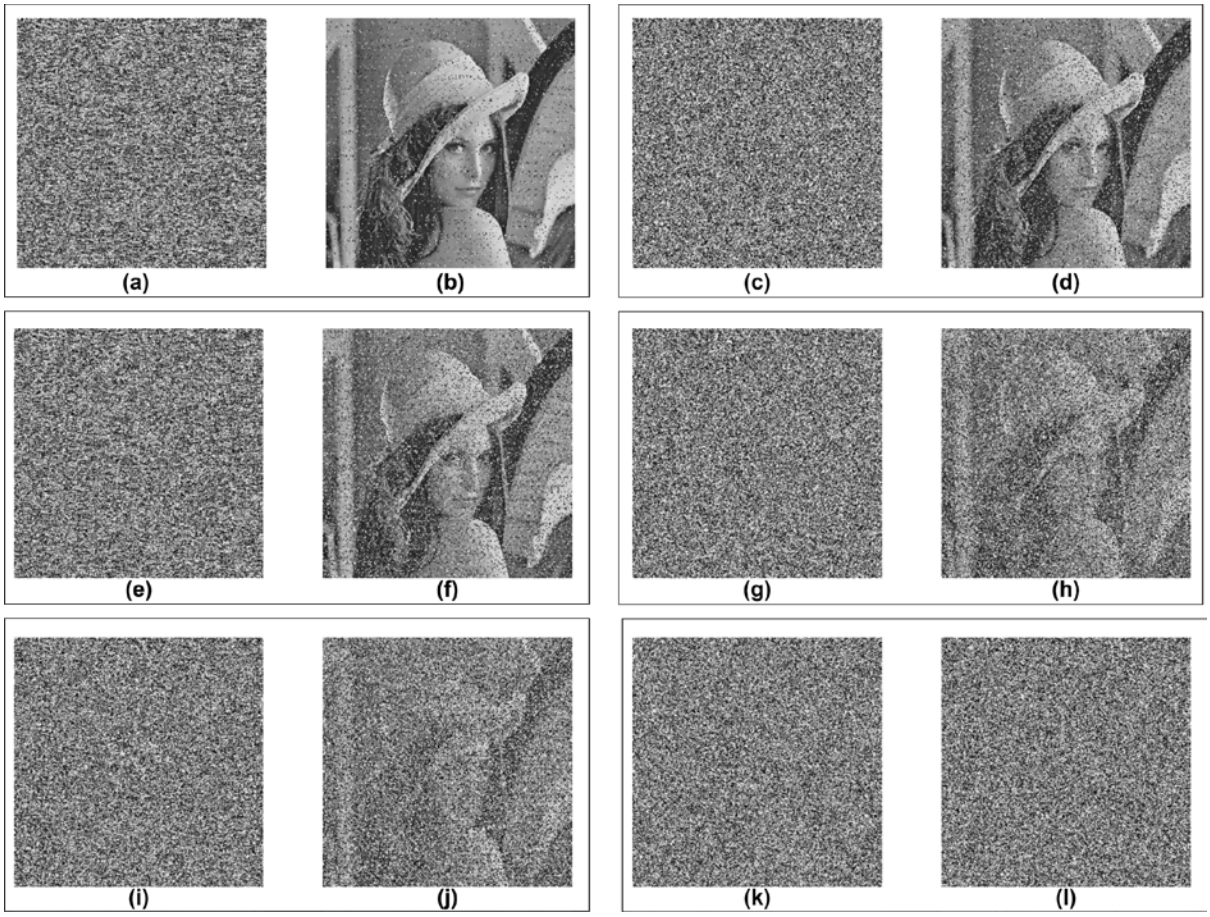

Figure 4. (a) 5% noise added image, (b) 5% noisily decrypted image, (c) 10% noise added image, (d) 10% noisily decrypted image, (e) 15% noise added image, (f) 15% noisily decrypted image, (g) 25% noise added image, (h) 25% noisily decrypted image, (i) 45% noise added image, (j) 45% noisily decrypted image, (k) 90% noise added image, and (l) 90% noisily decrypted image.

3.6. Time Consumption Results

An efficient algorithm should utilize the least number of resources. It is expected to complete the target operation in the shortest possible time, while also maximizing resource usage. Therefore, the developed algorithm is expected to be optimized for both resources and time. The system specifications for the proposed algorithm are as follows: i5-2400 CPU, 8GB 1333MHz RAM, Windows 10 operating system, and Python 3.8 programming language.

In this study, the time taken to encrypt and decrypt each image obtained from the SIPI database using the proposed method is specified in milliseconds in Table 6. The processing times obtained reflect the time consumption depending on the hardware resources of the system on

which the method is executed. Additionally, it can be understood from the results obtained that the time consumption also varies depending on the dimensions and number of channels of the image to be encrypted.

Table 6. Time consumption for encryption of proposed method.

Image	Size	Colour	Encryption Time (ms)	Decryption Time (ms)
Lena	256x256	Gray	107	104
Clock	256x256	Gray	102	99
Chemical plant	256x256	Gray	103	100
Boat	512x512	Gray	406	400
Gray	512x512	Gray	407	403
Stream and Bridge	512x512	Gray	398	392
Tank	512x512	Gray	406	402
Man	1024x1024	Gray	1787	1740
Airport	1024x1024	Gray	1683	1638
Lena	256x256	RGB	1098	1076
Plane	512x512	RGB	1318	1295
Peppers	512x512	RGB	1227	1204
Hause	512x512	RGB	1217	1193
Baboon	512x512	RGB	1194	1170

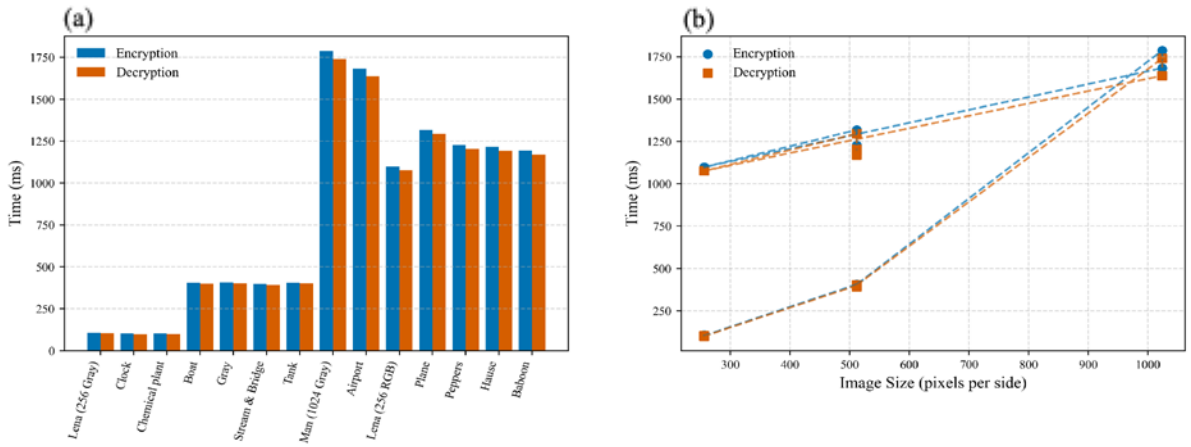

Figure 5. (a) Encryption and decryption time comparison of proposed method, and (b) Relation between image size and encryption - decryption times.

4. Discussion

In this study, a hybrid encryption method is proposed by addressing the problems of fixed initial conditions inherent in chaotic maps and the weakness of forward-only pixel encryption processes used in image encryption. A review of recent studies in the literature reveals that, due to the deterministic nature of chaotic maps, initial parameters are generated using pseudo-

random mechanisms with fixed or limited variation (Zhou et al., 2022), from ECG signals (Eldesouky et al., 2022) or from data obtained through iris scanning (Archana et al., 2022) and assumes secret key sharing as an external assumption. In the proposed study, the key space obtained using the PGP public key is $2^{100} * (2^8)!$ while (Zhou et al., 2022) obtained a value of $2^{1096+8N}$ in their study. Similarly, in their work (Eldesouky et al., 2022), they stated that the key space would be the same as the chaotic map used and referred to the 1D logistic map's value of 2^{104}. Thus, it is observed that the PGP public key method used in generating the chaotic map's parameters is rich in terms of key space. The fundamental difference between the proposed method and similar studies in the literature is the use of an asymmetric encryption mechanism to dynamically generate the input parameters of the chaotic system and the encryption of the obtained values with the XOR operator using the back propagation pixel values. Whereas previous studies (Althamir et al., 2023; Heinrich, 2025; Schwenk, 2022) have generally addressed PGP or RSA-like methods for key transmission but in this study, the recipient's public key has been transformed into a recipient-specific source that determines the μ and $x_0$parameters of the 1D logical map. This approach initiates the encryption process using receiver-specific values by dynamically generating the input parameter and initial value of chaotic maps.

Following the dynamic generation of chaotic map parameters, studies utilizing diffusion methods and bit operators for image encryption were examined. It was concluded that there was an increase in performance in terms of encryption security metrics between the NPCR and UACI results obtained by using the XOR operator continuously to change forward-looking pixel values in methods where both methods were used combined, and the use of the feedback XOR operator proposed in this study. In the study using continuous forward XOR processing (Hanif et al., 2022), a performance increase of 0.0335 for the NPCR value and 0.3494 for the UACI value was observed in the grey-level Lena image. Similarly, in the study by (Feng et al., 2022) a performance increase of 0.0021 for NPCR and 0.1415 for UACI was achieved on the Boat image. Furthermore, a similar performance increase was observed on the multi-channel RGB 4.2.07 image file. However, the increase in pixel density resulting from the increase in image size revealed a disadvantage of the proposed method. In the image named 7.1.03, a performance loss of NPCR 0.2794 and UACI 0.1063 was observed according to the forward XOR operator method proposed by (Zhou et al., 2022).

5. Conclusion

In conclusion, it has been determined that the proposed dynamic chaotic map parameter generation and the use of the back propagated XOR method within the scope of this study are feasible in terms of encryption security but are open to improvement. Furthermore, it is anticipated that eliminating the fixed initial value and input parameter problem in chaotic maps and utilizing the XOR operator through backpropagation across the entire image will provide a new perspective to the literature. Future work is expected to improve the performance metrics of the encryption process by increasing the size of the chaotic map used in image encryption or by developing the backpropagation method of the XOR operator.

Acknowledgments

The authors gratefully acknowledge the support of “HAVELSAN Co. Inc.” in this study.

Data and Code Availability

The source codes of all tests and analyses performed for the developed algorithm are shared at https://github.com/tlhcelik/a_novel_image_encryption_using_1d_logistic_map_and_pgp_key

References

Abba, A., Teh, J. S., & Alawida, M. (2024). Towards accurate keyspace analysis of chaos-based image ciphers. *Multimedia Tools and Applications*, *83*(33), 79047-79066. https://doi.org/10.1007/s11042-024-18628-8

Ali, N. H. M., Hoobi, M. M., & Hussien, S. A. S. (2025). Enhanced TEA Algorithm Performance using Affine Transformation and Chaotic Arnold Map. *Mesopotamian Journal of Computer Science*, *2025*, 341-354. https://doi.org/10.2139/ssrn.5681046

Althamir, M., Alabdulhay, A., & Yasin, M. M. (2023). A systematic literature review on symmetric and asymmetric encryption comparison key size. *2023 3rd International Conference on Smart Data Intelligence (ICSMDI)*, 110-117. https://doi.org/10.1109/icsmdi57622.2023.00027

Archana, K., Kumar, S. S., Gokak, P. P., Pragna, M., & Shruthi, M. (2022). IRIS Image Encryption and Decryption Based Application Using Chaos System and Confusion Technique. In *Modeling, Simulation and Optimization: Proceedings of CoMSO 2021* (pp. 155-175). Springer. https://doi.org/10.1007/978-981-19-0836-1_13

Artuğer, F. (2025). Innovative image encryption approach based on bitwise XOR high nonlinear S-boxes and random permutation. *Soft Computing*, *29*(6), 2891-2903. https://doi.org/10.1155/2017/6969312

Boldo, S., Jeannerod, C.-P., Melquiond, G., & Muller, J.-M. (2023). Floating-point arithmetic. *Acta Numerica*, *32*, 203-290. https://doi.org/10.1017/S0962492922000101

Braverman, L., & Nelson, D. R. (2025). Transition to chaos with conical billiards. *arXiv preprint arXiv:2508.02786*. https://doi.org/doi.org/10.1103/18f7-sqhz

California, U. o. S. (2024). *SIPI database of University of Southern California*. Retrieved 16 March from https://sipi.usc.edu/database/database.php?volume=misc

Diffie, W., & Hellman, M. E. (2022). New directions in cryptography. In Democratizing cryptography: the work of Whitfield Diffie and Martin Hellman (pp. 365-390)

Eldesouky, S., El-Shafai, W., Ahmed, H. E. d. H., & El-Samie, F. E. A. (2022). Cancelable electrocardiogram biometric system based on chaotic encryption using three-dimensional logistic map for biometric-based cloud services. *Security and Privacy, 5*(2), e198. https://doi.org/10.1002/spy2.198

Feng, W., Zhao, X., Zhang, J., Qin, Z., Zhang, J., & He, Y. (2022). Image encryption algorithm based on plane-level image filtering and discrete logarithmic transform. *Mathematics, 10*(15), 2751. https://doi.org/10.3390/math10152751

Geetha, M., Ranjithkannan, T., Yogeshwaran, R., & Nishanth, S. (2025). Next-Gen Multimedia Encryption by Combining Symmetric and Asymmetric Cryptographic Techniques. *2025 5th International Conference on Expert Clouds and Applications (ICOECA)*, 49-53. https://doi.org/10.1109/icoeca66273.2025.00020

Gupta, L., Jaiswal, P., Lather, I., Agarwal, R., & Thakur, A. (2023). Image Encryption using Chaotic maps: State of the art. *2023 3rd International Conference on Intelligent Technologies (CONIT)*, 1-8. https://doi.org/10.1109/CONIT59222.2023.10205829

Hanif, M., Iqbal, N., Ur Rahman, F., Khan, M. A., Ghazal, T. M., Abbas, S., Ahmad, M., Al Hamadi, H., & Yeun, C. Y. (2022). A novel grayscale image encryption scheme based on the block-level swapping of pixels and the chaotic system. *Sensors, 22*(16), 6243. https://doi.org/10.3390/s22166243

Heinrich, C. (2025). Pretty good privacy (PGP). In *Encyclopedia of Cryptography, Security and Privacy* (pp. 1863-1868). Springer. https://doi.org/10.1007/978-3-030-71522-9_215

Ihsan, A., & Doğan, N. (2023). Improved affine encryption algorithm for color images using LFSR and XOR encryption. *Multimedia Tools and Applications, 82*(5), 7621-7637. https://doi.org/10.1007/s11042-022-13727-w

Karthik, G., Suresh, P., Ganesh, V., Kumar, H., & Amirtharajan, R. (2025). Elliptical Curve Cryptography for Images Using Mandelbrot Fractal and Zaslavskii Map Based Multiple Key with DNA Encoding. *2025 11th International Conference on Communication and Signal Processing (ICCSP)*, 960-964. https://doi.org/10.1109/iccsp64183.2025.11088650

Li, H., Yu, S., Feng, W., Chen, Y., Zhang, J., Qin, Z., Zhu, Z., & Wozniak, M. (2023). Exploiting dynamic vector-level operations and a 2D-enhanced logistic modular map for efficient chaotic image encryption. *Entropy, 25*(8), 1147. https://doi.org/10.3390/e25081147

Lyle, M., Sarosh, P., & Parah, S. A. (2022). Adaptive image encryption based on twin chaotic maps. *Multimedia Tools and Applications, 81*(6), 8179-8198. https://doi.org/10.1007/s11042-022-11917-0

Paramesha, K., Karthik, V., Prashanth, M., Sathisha, M., Bhargav, H., HS, R. K., Raju, K., & Puttegowda, K. (2025). A novel image cryptosystem for biomedical images and secured storage by randomized chaotic encryption scheme. *Journal of Integrated Science and Technology*, *13*(5), 1103-1103. https://doi.org/10.62110/sciencein.jist.2025.v13.1103

Saif, A. (2025). *Data Security on cloud using Hybrid Cryptography a PGP based encryption methodology* Dublin, National College of Ireland].

Schwenk, J. (2022). File Encryption: PGP. In *Guide to Internet Cryptography: Security Protocols and Real-World Attack Implications* (pp. 377-399). Springer. https://doi.org/10.1007/978-3-031-19439-9_16

Sivasankari, N., & Kamalakkannan, S. (2022). Detection and prevention of man-in-the-middle attack in iot network using regression modeling. *Advances in Engineering software, 169*, 103126. https://doi.org/10.1016/j.advengsoft.2022.103126

Susanti, B. H., Sumule, A. S. L., & Ardyani, M. W. (2025). Security Analysis of Modified ESRKGS-RSA Using Lenstra's Elliptic Curve Method. *CAUCHY: Jurnal Matematika Murni dan Aplikasi*, *10*(2), 805-820. https://doi.org/0.18860/cauchy.v10i2.32189

Taqi, I. A., & Abdul-Haleem, M. G. (2023). An Efficient Cryptosystem for Image Using 1D and 2D Logistic Chaotic Maps. *International Journal of Intelligent Engineering & Systems, 16*(4). https://doi.org/10.22266/ijies2023.0831.11

Ullah, A., Shah, A. A., Khan, J. S., Sajjad, M., Boulila, W., Akgul, A., Masood, J., Ghaleb, F. A., Shah, S. A., & Ahmad, J. (2022). An efficient lightweight image encryption scheme using multichaos. *Security and Communication Networks, 2022*(1), 5680357. https://doi.org/10.1155/2022/5680357

Wang, G. (2024). *Remote sensing image encryption algorithm based on novel spatiotemporal chaotic system* (Patent No. CN119814936). U. S. T. LIAONING. https://patentscope.wipo.int/search/en/detail.jsf?docId=CN454494386

Yu, J. (2023). Based on Gaussian filter to improve the effect of the images in Gaussian noise and pepper noise. *Journal of Physics: Conference Series, 2580*(1), 012062. https://doi.org/10.1088/1742-6596/2580/1/012062

Zhang, Y. T. Y. (2024). *Image encryption and decryption method and system based on SM2 and DNA* (Patent No. CN118018659A). E. U. O. T. C. P. S. A. P. FORCE. https://patentscope.wipo.int/search/en/detail.jsf?docId=CN429414066

Zhou, S., Wang, X., Zhang, Y., Ge, B., Wang, M., & Gao, S. (2022). A novel image encryption cryptosystem based on true random numbers and chaotic systems. *Multimedia Systems, 28*(1), 95-112. https://doi.org/10.1007/s00530-021-00803-8