INTERNATIONAL JOURNAL OF Telecommunications
AIR DEFENSE COLLEGE

Robust Image Encryption Framework Leveraging Multi-Chaotic Map Synergy for Advanced Data Security

Citation: Abodawood, M. A.; Khalil, A. T; Amer, H. M, Ata, M. M.
Inter. Jour. of Telecommunications, ***IJT'2025****,*
Vol. 05, Issue 01, pp. 1-28, 2025.
Doi: 10.21608/ijt.2025.354464.1079
Editor-in-Chief: Youssef Fayed.
Received: 21/01/2025.
Accepted: date 14/04/2025.
Published: date 17/04/2025.
Publisher's Note: The International Journal of Telecommunications, IJT, stays neutral re-gard-ing jurisdictional claims in published maps and institutional affiliations.

Mostafa Abodawood [1], Abeer Twakol Khalil[1], Hanan M. Amer[1], Mohamed Maher Ata[2]

[1] Electronics and Communications Engineering Department, Faculty of Engineering, Mansoura University, Mansoura, Egypt, mostafaabodawood1@gmail.com, abeer.tawakol@mans.edu.eg, eng_hanan_2007@mans.edu.eg
[2] School of Computational Sciences and Artificial Intelligence (CSAI) Zewail City of Science and Technology 6th of October City, Giza, Egypt, momaher@zewailcity.edu.eg
* Correspondence: Mohamed Maher Ata, momaher@zewailcity.edu.eg

Abstract: In the field of data security, image encryption is a crucial technique created especially for protecting visual data from abuse and unwanted access. Digital images are more vulnerable to illegal interception, duplication, and manipulation as they are exchanged and stored more frequently across public and private networks. These images are rendered unintelligible by encryption, which guarantees that the original images may only be accessed by authorized users or systems that the possess decryption keys. The model for image encryption presented in this research uses chaotic maps, making use of chaotic systems' inherent unpredictability to produce a strong encryption technique. Five chaotic maps are used in this approach to carry out the encryption procedure: Logistic, Tent, Circle, Chebyshev, and Piecewise maps. Because each map adds distinct chaotic features, a complex and safe foundation for image encryption is produced. We used a variety of performance indicators to evaluate this encryption method's efficiency such as histo-gram analysis, cross-correlation, execution time, and entropy It showed how effective the suggested algorithm was.

Keywords: Network, Imaging, Algorithm, Interception, Key

1. Introduction

Since data transmission technologies are developing so quickly, it is more crucial than ever to protect data while it is being transferred from sender to recipient in the context of digital communication. To avoid unwanted access and modification, it is essential to ensure the security of the data during this transfer procedure. Three popular methods are frequently employed to protect digital data and keep it safe from unauthorized users: steganography, cryptography, and watermarking [1]. Every approach has advantages of its own, but cryptography is clearly essential for ensuring security during data transport.

The two main categories of cryptographic techniques are block ciphers and stream ciphers [2]. Stream ciphers use a secret key to generate a pseudo-random bit sequence by working with individual bits or bytes of data one at a time. Though it tends to process encryption more slowly than block ciphers, this bit-by-bit method provides a higher level of security. On the other hand, a block cipher applies encryption to groups of pixels in the case of image encryption and encrypts data in bigger, fixed-size blocks. Although block cyphers are typically quicker than stream cyphers, they may be more vulnerable to specific kinds of attacks. The Triple Data Encryption Algorithm (TDEA), the Advanced Encryption Standard (AES), and the Data Encryption Standard (DES) are the three most widely used block cipher types [3]. But when these techniques are used on images, it's common to see strong correlations between duplicate data items, such neighboring pixels. Furthermore, block ciphers usually take longer to process, which may be a drawback for applications requiring real-time encryption.

Researchers have created a number of creative image encryption techniques to overcome these constraints. Chaotic map algorithms are a particularly successful method because they include features that are ideal for image encryption [4]. Chaotic maps possess features like unpredictability, randomness, extreme sensitivity to

initial condition, ergodic behavior, and reproducibility which allowing for the fast and accurate generation of numerous sequential chaotic patterns. These characteristics allow chaotic map-based algorithms to generate secure, advanced encryption at fast processing speeds.

There are typically two main steps in the chaotic map image encryption process: confusion and diffusion [5]. By rearranging pixel positions in a complex and random way, the confusion stage seeks to hide the relationship among the encrypted (cipher) images and the original (plain) images. Conversely, the diffusion step ensures that small changes in the plaintext result in large changes in the encrypted output by dispersing the plaintext image's statistical characteristics over a large portion of the ciphertext [6]. Because of their twin strategy of confusion and diffusion, chaotic map algorithms offer strong complexity, improved security, and quick processing speeds, making them ideal for image encryption.

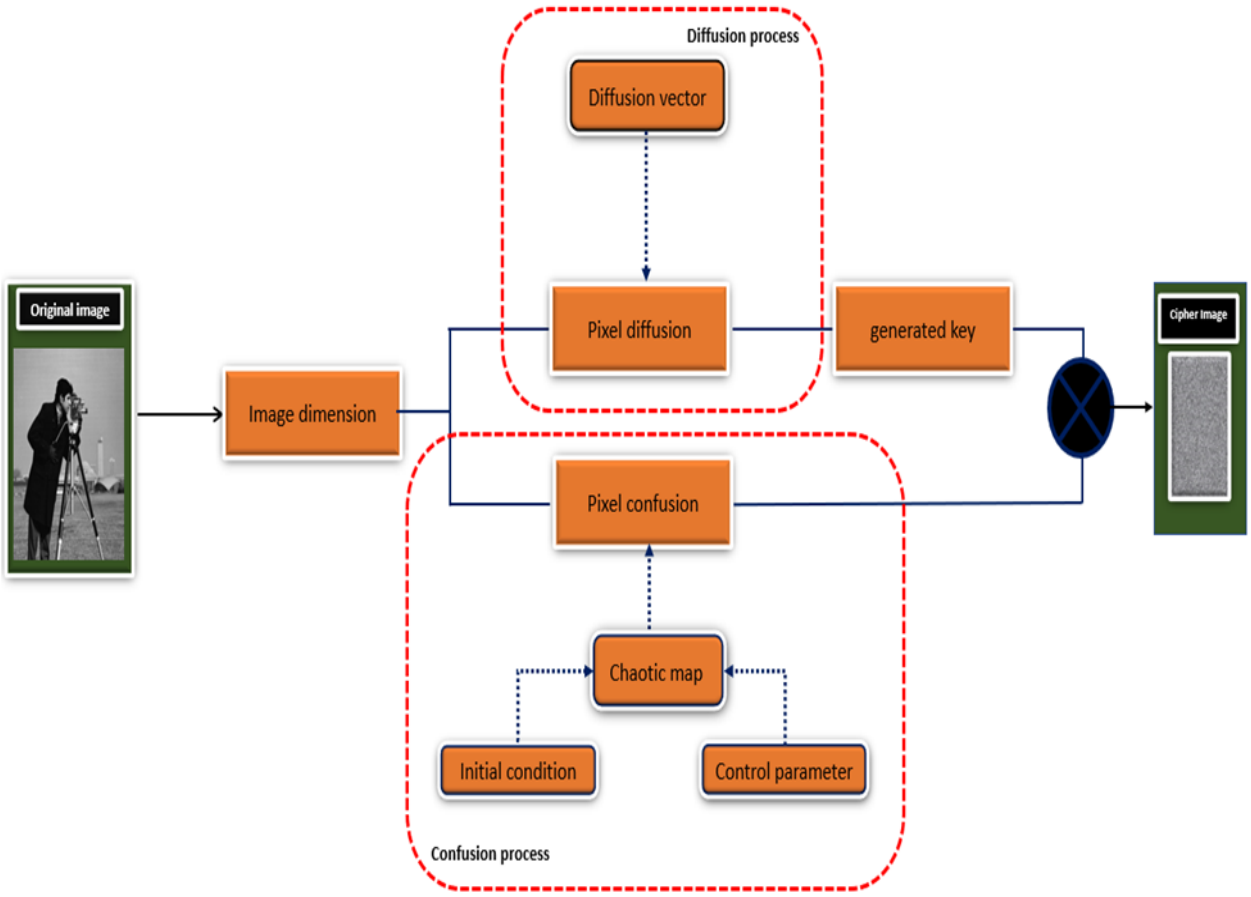


Figure .1 illustrates the encryption process.

A proposed image security technique is shown in figure 1. Several significant advancements in the field of secure image encryption are provided by this technique:

1 - Presenting a novel approach to image encryption that uses chaotic maps to increase security and robustness by using the complexity and randomness of chaotic sequences as crucial components.

2 - Introducing advanced image encryption methods that improve image security and provide robust defense
against a range of attacks

3 - Five chaotic maps are evaluated using standardized images in order to assess the efficacy of the suggested method over a range of image properties.

2. Related Work

Arslan Shafique et al [7]. presents a robust data security technique with five key steps for encrypting grayscale images. Secret key generation, confusion stage, and three levels of diffusion. The suggested encryption framework strengthens its defenses against cyberattacks. Javad Mostafaee et al [8]. It presents a new encryption technique using hyper chaotic map with the usage of complex dynamic behaviors. This method is used to encrypt color images based on DNA coding. The results showed that the presented system increases the security process during encryption. Zilong Liu et al [9]. They presented a new algorithm for face encryption that depends on the characteristics of the tent and Henon map. This method relies on homomorphic encryption to extract visual robustness. It also uses a neural network model to design the encrypted face recognition algorithm. The results presented in this research paper showed that the presented algorithm has the ability to encrypt the face in a safe and effective way.

F. Yang et al [10]. They developed a chaotic map based on an iterative Map with Infinite Collapses in order to control a 2D chaotic map. It was found that the map provided as a result of development is more restrictive and suitable for securing communication operations. Z. Guo et al [11]. Proposed a new algorithm that depends on a reverse zigzag algorithm combined with DNA encoding. Pixel coordinates are arranged using the technique, and a secondary image is then mixed in. Once the scrambling process is finished, the DNA pixel diffusion is finished. the proposed algorithm can resist common types of attacks and provide high security. Hong, Y et al [12]. Proposed an algorithm for encrypting data based on infinite folding binary iterative map. A binary bidirectional zigzag scrambling technique is suggested to dislocate the image during the encryption process, and the mapping is employed to create chaotic sequences for further encryption. The results of the simulation experiment demonstrate the algorithm's high level of safety.

Yubao. S et al [13]. present an improved chaotic system and use it to create a novel image encryption technique that is naturally impenetrable. By using a fixed key instead of the traditional one-time key, this technique improves encryption efficiency and security. Heping. W et al [14]. Produces a bit-level image encryption algorithm (BCIEA) that relies on chaotic maps. Diffusion and confusion processes are integrated by BCIEA, and the dynamic methods used in these stages are largely responsible for its security performance. The proposed dynamic technique has observable statistical features, which makes it especially susceptible to an all-zero ciphertext attack, according to an examination of its confusion. This algorithm provides a high level of safety depending on bit level that used in image encryption.

Elkhalil, N et al [15]. proposed a new two-dimensional chaotic map called the Two-Dimensional Beta Chaotic Map (2D-BCM) based on the one-dimensional Beta Chaotic Map (1D-BCM). The three main steps of the suggested algorithm are substitution, diffusion, and permutation. In order to create the encryption key, the new 2D-BCM produces chaotic sequences. The proposed system provides high sensitivity. Yeh, T. et al [16]. Produces a second-order chaotic maps that produce chaotic sequences for image encryption. Eight different weighting schemes are used in a particle swarm optimization process to find the best chaotic maps. To further decrease the encryption period, a hybrid sequence generation (HSG) technique is also demonstrated. Shao, S., Li et al [17]. A novel Piecewise-Logistic-Sine map (PLSM) is presented. A PLSM-based image encryption method with random exclusive OR diffusion is presented. The key distribution approach uses a 256-bit secret key to determine the PLSM's initial value and parameters. The performance evaluation shows how secure the images encryption approach is, and this scheme can spread a small alteration in the original image across all pixels.

Rahul Bhogal et al [18]. produce a new one-dimensional (1-D) Sine-Tangent Chaotic map (STCM) and a shared key is used to present a cryptographic system for protecting medical photographs. XOR operations are used to produce a private shared key first, and then permutation and substitution procedures are applied. The novel (STCM) is then used to create a chaotic sequence. The proposed model provides a high level of security. Mujeeb Ur Rehman et al [19]. The proposed study presents a reliable data security technique to encrypt grayscale im-

ages. The system is divided into five primary stages. The creation of secret keys and the implementation of the confusion stage take up the first two stages of the suggested encryption structure. Then, in phases 3, 4, and 5, the diffusion processes at level 1, level 2, and level 3 are carried out, accordingly.
Alaklabi, A et al [20]. In order to increase security and improve resilience to attacks, this study presents Z-Crypt, a novel image encryption technique that combines the Chirp Z-Transform (CZT) and a substitution-permutation network (SPN) with a chaotic logistic map (CLM). The experimental results verify that the suggested approach outperforms existing techniques, providing strong security while guaranteeing computational effectiveness. Vijayakumar, M al [21]. This paper introduces a new encryption method that overcomes common problems in chaotic encryption approaches by using substitution boxes (S-box) generated from cellular automata (CA) and chaotic maps. The proposed hybrid approach surpasses traditional algorithms while preserving image quality.

3. Chaotic Map

A system or mathematical function that demonstrates chaotic behavior —characterized by complex dynamic patterns, deterministic but unpredictable results, and great sensitivity to initial conditions [22]. Five chaotic maps are briefly reviewed in this section, along with each map's special capabilities for producing chaotic sequences.

3.1 Logistic Map

A popular mathematical tool in chaos theory for simulating chaotic behavior is the logistic chaotic map. It comes from the logistic equation, which is written as follows:

$$W_{z+1} = GW_z(1 - W_z) \tag{1}$$

Here, **G** serves as the control parameter, varying between 0 and 4, the number of iterations represented by **z**, and **W** indicates the sequence of the chaotic map, which ranges between 0 and 1 [23].

3.2 Tent Map

A piecewise linear chaotic system called the tent chaotic map is frequently used to simulate chaotic activity and produce sequences that resemble randomness. The definition of its equation is:

$$W_{z+1} = \begin{cases} G\,W_z & if \quad 0 \prec W_z \prec 0.5 \\ G\,(1 - W_z) & if \quad 0.5 \preccurlyeq W_z \prec 1 \end{cases} \tag{2}$$

Here, **W** stands for the chaotic sequence, which has a range of 0 to 1, **G** for the parameter control, which has a range of 0 to 2, and number of iterations represents by **z**.

3.3 Circle Map

A mathematical system called the circle chaotic map is used to generate chaotic behavior It is frequently employed in random sequence generation and cryptography. Typically, this equation is written as:

$$W_{z+1} = mod\ (W_z + D - \left(\frac{K}{2\pi}\right) sin\ (2\pi W_z)\ , 1) \tag{3}$$

In this case, the chaotic sequence, denoted by **W**, ranges from 0 to 1, the system parameters, represented by **k** and **D**, have an impact on the chaotic behavior.

3.4 Chebyshev Map

A mathematical model called the Chebyshev chaotic map, which is based on Chebyshev polynomials, is well-known for its capacity to produce chaotic sequences. It is described as:

$$\boldsymbol{W_{z+1} = \cos(G * \cos^{-1}(W_z))} \tag{4}$$

Here, **W** stands for the chaotic sequence, which can vary from -1 to 1, **z** for the number of repetitions, and **G** for the control parameter, which can vary from 0 to 2 [24].

3.5 Piecewise Map

The piecewise chaotic map is a kind of mathematical technique that divides the domain into discrete intervals, each of which is subject to a unique functional rule, in order to produce chaotic behavior. Its standard form is:

$$W_{z+1} = \begin{cases} \frac{W_z}{a} & if \quad 0 \le y_z \le G \\ \frac{W_z - G}{0.5 - G} & if \quad G \le y_z < 0.5 \\ \frac{1 - G - W_z}{0.5 - G} & if \quad 0.5 < y_z < 1 - G \\ \frac{1 - W_z}{G} & if \quad 1 - G \le W_z < 1 \end{cases} \tag{5}$$

In this case, the chaotic sequence, represented b **W**, W is between 0 and 1, the number of iterations, represented by **z**, and the control parameter, represented by **G**, is between 0 and 2.

The bifurcation diagram for the five chaotic maps that used in the suggested approach is shown in Figure 2. The horizontal axis in this graphic indicates the control parameter G, and the vertical axis represents W. The bifurcation diagram demonstrates how the initial conditions influence the resulting chaotic behavior [25].

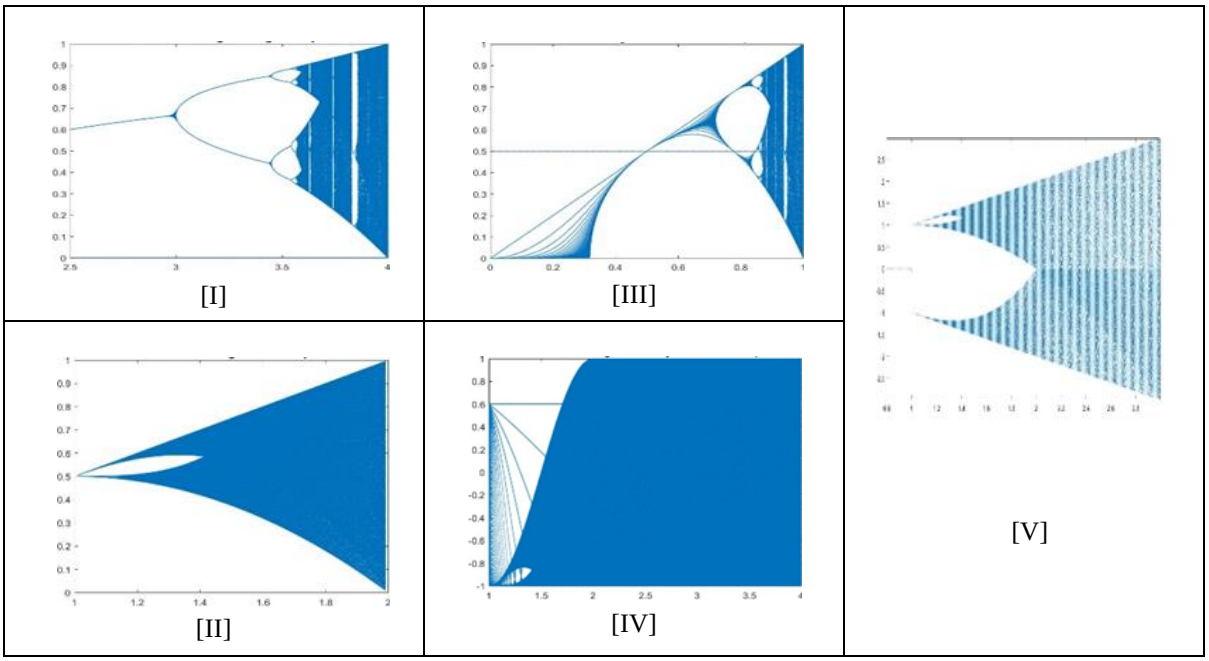


Figure .2 depicts the bifurcation structure of [I] logistic map, [II] Tent map, [III] Circle map, [IV] Chebyshev map, and [V] Piecewise map

4. proposed Algorithm

A strong technique for image encryption, chaotic maps take advantage of their internal complexity and pseudo-randomness. High security and comparatively low computing overhead are possible with their appropriate implementation. The initial condition, which forms the foundation of the encryption process, is crucial to chaotic maps.

First: choosing the initial condition:

The robustness, unpredictability, and security of applications such as image encryption depend on the initial circumstances chosen for chaotic maps. The chaotic aspect of the system is enhanced by carefully chosen initial conditions, which makes the encryption process impenetrable to attacks while keeping reproducibility for authorized users. The following are the main justifications for why initial conditions are crucial in chaotic maps:

1. **Initial Condition Sensitivity:**

The butterfly effect refers to chaotic maps' extreme sensitivity to their initial conditions. Over time, even a slight alteration to these parameters produces wildly different results.

2. **Deterministic Behavior:**

This guarantees repeatability in encryption, enabling the original image to be correctly decrypted provided that the right initial conditions are known.

3. **Security Against Attacks:**

The predictability or redundancy of keys is required for many cryptographic attacks, including differential and brute-force attacks. A high degree of sensitivity and randomness is ensured by carefully selected for initial conditions, which makes it challenging for attackers to guess or use the key.

In order to identify the best initial conditions, put in place a system that dynamically modifies them in response to the success rates shown by statistical tests. In order to achieve the best outcomes, this method seeks to reduce the range of the initial conditions chosen. The following steps are essential for the procedures of choosing the initial condition:

Step 1: Choose several values within the same range (probably between 0 and 1) or begin with a predate mind set of initial condition, such as (0.1 - 0.4) or other value. The foundation for additional processing will be this value.

Step 2: A single-digit number (0 through 9) can be generated by using a random number generator, such as the rand function. This adds an element of chance to the procedure.

Step 3: To the existing initial state, add the number that was produced at random in Step 2. By changing the initial condition, this procedure generates a new value that can be used later.

Step 4: Observe the numbers or outcomes produced. Find the number that comes up most often. Statistical observation is used in this step to identify common patterns.

Step 5: Several times, repeat Step 2 until the resulting initial condition is as precise as 15 decimal places. This guarantees a very fine-tuned and accurate starting condition.

Table1. show the achieved initial condition.

Initial conditions	Value
Initial condition values	0.766655321852092
	0.479041456288128
	0.468325532750961
	0.444614236929226
	0.306152236972732
	0.187152236382040

An essential part of the encryption process, the finalized initial condition is shown in Table 1. During the confusion stage, when permutations are carried out, this requirement and the chaotic map's control parameter are crucial. The control parameter, denoted by G in Figure 3, is a crucial component of chaotic systems. The way the technique reorganizes the original image to produce the encrypted image is greatly influenced by these parameters.

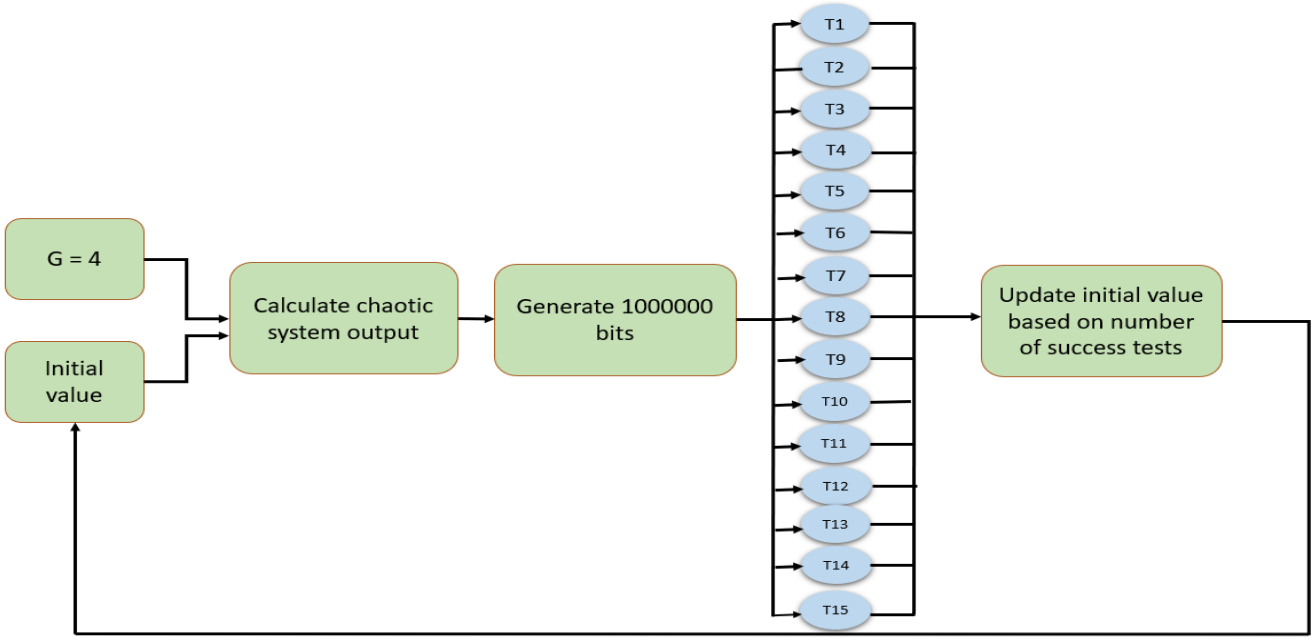


Figure .3 Summary of the approach for determining initial conditions.

Second: Confusion process:

1) Insert the original image (M) into the system to start the procedure. The image's dimensions are A×B, where A is the number of rows and B is the number of columns respectively. The input data is set up in this stage for use in later processes.

2) In order to improve speed and streamline the computation, the original image (M) is transformed into a grayscale version. Different shades of gray are represented by pixel values in grayscale images, which range from 0 to 255. By removing color information while maintaining the necessary structure for encryption, this lowers the image's complexity.

3) To generate a chaotic sequence vector (y), use initial circumstances and control parameter. A chaotic map is used to create the chaotic sequence, which is a collection of pseudo-random numbers. The image's pixel values are rearranged and encrypted using this sequence, which guarantees security and randomization.

4) In order to establish a connection between the chaotic sequence and the indices of the pixel values in the image, sorting is done accord[illegible] tion:

$$[X, Y] = \text{sort}(t) \tag{6}$$

The newly arranged vector is represented by **X**, and its index is represented by **Y**.

Third: Diffusion process:

1) Beginning with the original image, use the formula

$$R(t+1) = \cos(p \cos(R(t))) \tag{7}$$

to build a vector (R). Repeat the previous equation A×B times, where A and B stand for the image's dimensions. As a result, a vector is produced for the diffusion.

2) After multiplying each diffusion vector R element by 255 and rounding the result, determine each element's absolute value. The vector D is the outcome of this.

3) Transform the vector D's values from binary to decimal. After conversion, give the binary data's bits a single circular shift.

4) Return to decimal from the shifted binary vector. Next, combine the shifted vector with the vector D using the XOR method. A key stream will result from this.

5) Lastly, use the key stream created in the previous step to perform the XOR operation on the permuted (changed) image. This creates the image's encrypted version.

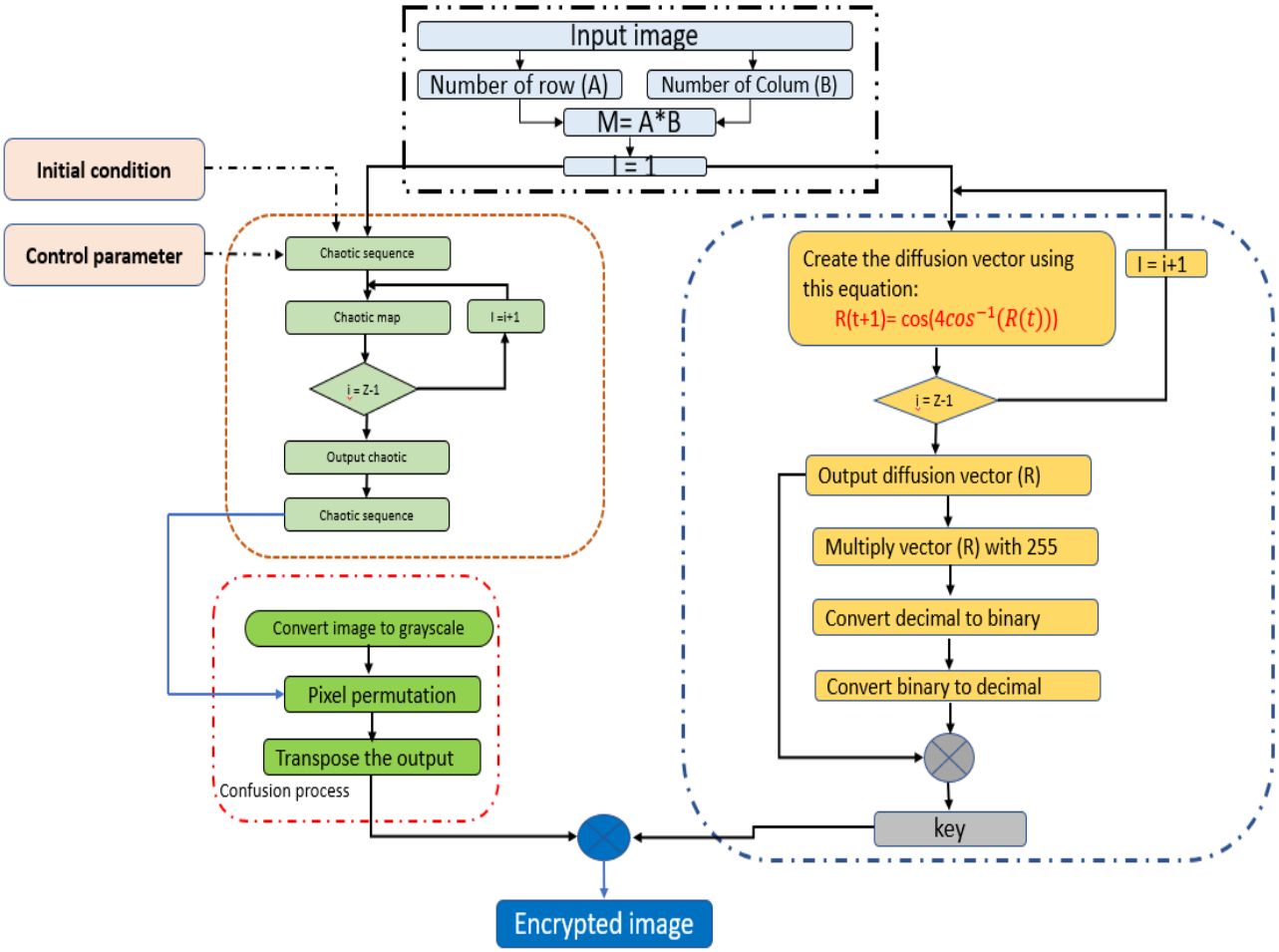


Figure.4 Block diagram illustrating the encryption procedure.

A comprehensive, pseudo-coded overview of the complete encryption process is provided by the second algorithm, whereas the first algorithm establishes the initial state.

Algorithm .1 generating the initial condition

```
1-      format long
2-    S = [0.1 0.4 0.9]        %any number within the range
3-  R = [rand() rand() rand()]
4-   sum= S + R
5-       if (sum(1)> 1)
           sum(1)=(sum(1)-1);
         end
6-      if (sum(2)> 1)
           sum(2)=(sum(2)-1);
         end
7-      if (sum(3)> 1)
           sum(3)=(sum(3)-1);
         end
8-   final = [ sum(1) sum(2) sum(3)]
%repeat these steps until the number of digits after comma reaches 15.
```

Algorithm 2. Encryption algorithm

```
Input    : A plain-image of size A * B          % original image
Output   : encrypted image
A: put the initial condition that outside from the first algorithm
B: Generating chaotic sequence vector
1.   Img ◄--- Read input image
2.   A , B ◄-- size of the original image
3.   M= A * B
4.   W(1)◄--   Use the optimal solution resulting from the first algorithm as the initial parameter.
5.   For (z= 1 : Z-1)
6.        W(z+1)◄---  chaotic map function
7.     end
8.   W(z)◄--- chaotic sequence generated that used to make permutations
9.   [X , Y]◄-- sort chaotic sequence element
10.  return sorted chaotic sequence
C: confusion
20. Img ◄---read input image
21. Img ◄--- reshape img to one dimensional column vector
22. For ( m=1 :size of img)
    a. Img (in(m)) ◄-- img (m), pixel permutation using the sorted chaotic sequence
23. end
24. Img ◄--  return permuted vector
25. timg ◄--  transpose the permuted vector
D: generation of diffusion key
10. R(1) ◄--  set initial condition for Key image
11. For (z= 1 : Z-1)
12 R(z+1)◄-- cos (4* acos (R((z)))
13. End
14. D ◄--  R *255, then roundoff & consider absolute
15. A◄--  convert D from decimal to binary
16. A ◄--  one-bit circular shift of A
17. A◄--  convert A from binary to decimal
18. Key  ◄-- D  x-or  A
19. return key
E: final encryption
26. Eimg◄---  timg  xor key
27. Eimg ◄---  reshape  Eimg vector to A*B matrix
28. return encrypted image
```

Explaining the second algorithm:

1- The image is converted into a stream of data, usually represented by the intensity values of the pixels (e.g., grayscale values for each pixel). This stage transforms the image into a format that can be altered mathematically, readying it for processing.

2- The image is converted to grayscale if it is in color (RGB format). By displaying the intensity of each pixel on a scale (often ranging from 0 to 255), grayscale simplifies the image and focuses the encryption process more on intensity values.

3- Using the output of the first algorithm (the initial condition), a chaotic sequence—a set of values that seem random but are actually determined by a specified process—is produced (e.g., chaotic maps). The encryption of the image's pixels will be influenced by this sequence.

4- Sorting is done on the chaotic sequence vector, usually in either ascending or descending order. Sorting makes it easier to create a consistent pattern for later encryption operations of the image pixels.

5- The surrounding pixels are rearranged or permuted using the sorted chaotic sequence. Basically, the chaotic sequence is used to shuffle the image pixels, changing their positions while maintaining their intensity values.

6- Transposing the permuted vector comes after permuting the pixels. Transposing adds an additional layer of manipulation to further obfuscate the original image by altering the data structure of the image (i.e., switching rows and columns).

7- A key stream is created in this stage. The permuted image will be encrypted using a key stream, which is a series of pseudo-random values. In order to further mix the pixel values and make the encryption more difficult to crack, this method entails producing values that will disperse throughout the image.

8- Lastly, the key stream and the permuted image are XORed. A bitwise operation called XOR joins two values so that the outcome can only be undone with the right key. This produces the encrypted image, which is completely different from the original and requires the correct key to decipher.

5. Results

5.1 performance metrics

Performance metrics are important indicators that are used to assess and gauge the encryption process's efficacy, efficiency, and quality**.**

5.1.1 Statistical Attack Analysis

5.1.1.1 Entropy Analysis: is a term used in image encryption to describe the degree of unpredictability or randomness in the encrypted image's pixel values. It is an essential indicator for assessing an encryption algorithm's efficacy.

$$H(w) = -\sum_{i=1}^{n} P(w_i) log_2 P(w_i) \quad (8)$$

Where $p(w_i)$ represents the probability of the pixel having intensity and **n** represents the number of intensities. As seen in figure 5, the cipher image's entropy is close to its optimal value.

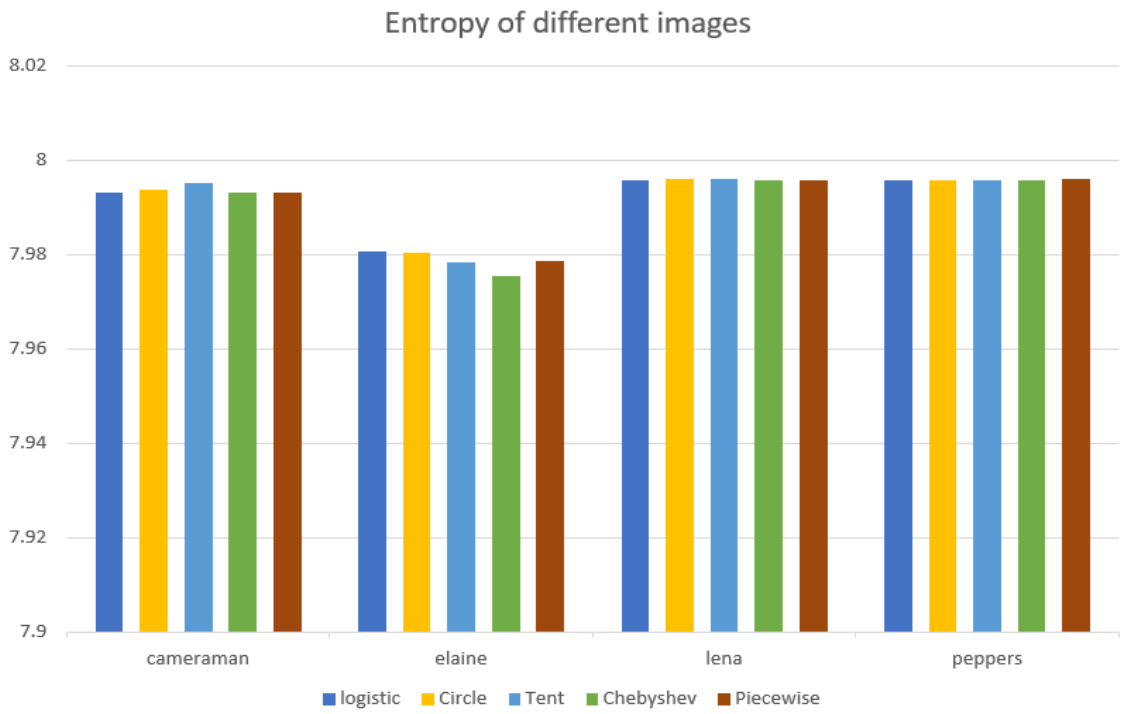
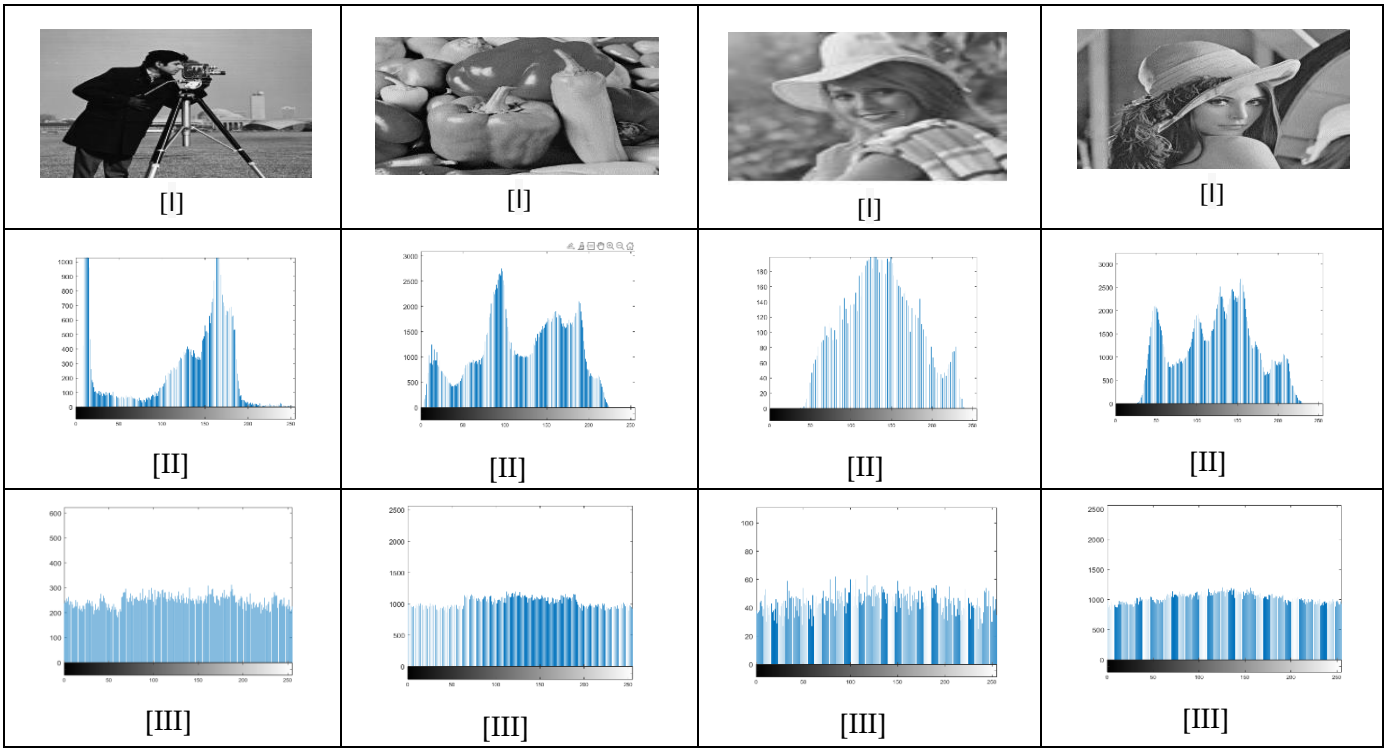


Figure 5. show the encrypted image's entropy result.

Table 2. represent entropy comparison of the proposed algorithm with alternative algorithms.

name	Suggested	[26]	[27]	[28]	[29]	[30]	[31]
Cameraman	7.9925	7.9862	7.991	___	7.9971	___	___
Peppers	7.9985	7.9827	7.9842	___	7.9871	___	___
Elaine	7.9814	7.9895	___	7.981	7.9935	___	___
Lena	7.9945	7.9925	7.9633	7.926	7.9907	7.9945	7.9987

5.1.1.2 Histogram analysis: Analyzing an image's pixel distribution is one way to assess how effective an encryption technique is. This evaluation looks at the frequency of appearance of various pixel intensities. The goal of a strong image's encryption method is to mask the original image's statistical features. A uniform pixel intensity distribution, with equal representation of all intensity levels, is ideal for an encrypted image. Because of this consistency, adversaries find difficult to deduce the plain image's content. The pixel intensity distributions of the plain and cipher images are shown in Figure 6, which provides an encryption with five chaotic maps produce a uniform distribution.



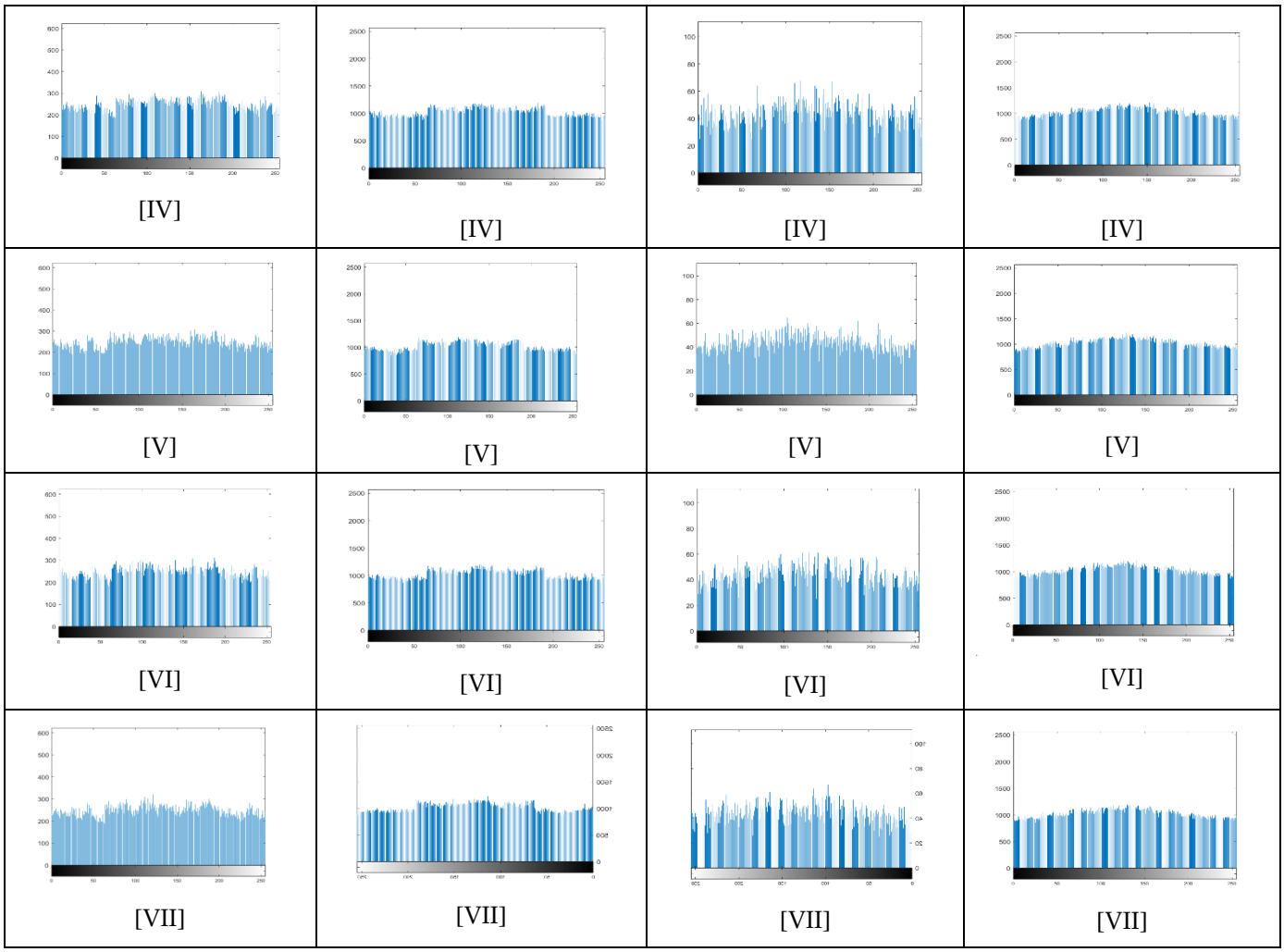


Figure 6. shows the original image (I), the plain image analysis (II), and the cipher image using the following techniques [III] Logistic map, [IV] Tent map, [V] Circle map, [VI] Chebyshev map, and [VII] Piecewise map.

5.1.1.3 Correlation Analysis: In image encryption, correlation analysis analyzes how effectively an encryption technique weakens the connection between neighboring pixels in an image. It calculates how dependent or similar nearby pixels are in both the original and encrypted images. In order to demonstrate randomness and make the encrypted image impervious to statistical attacks, a robust encryption algorithm makes sure that the correlation between neighboring pixels in the encrypted image is as near to zero as feasible.

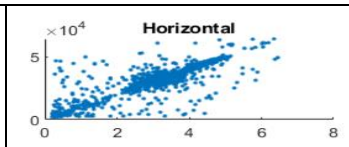
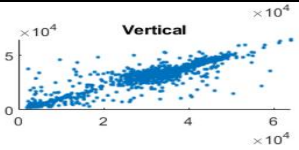
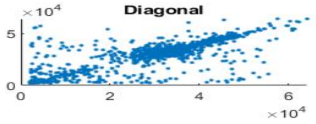
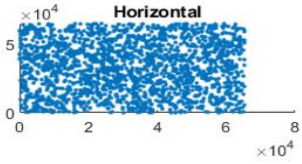
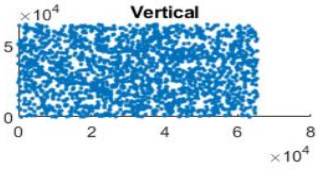
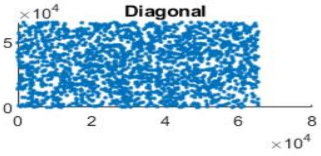
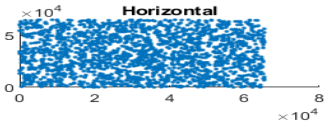
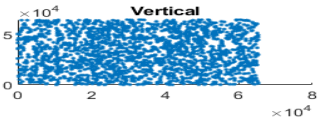
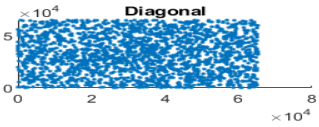
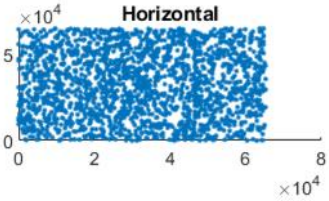
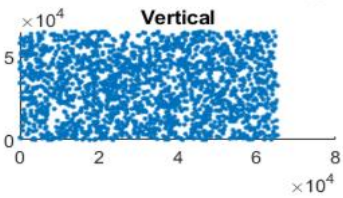
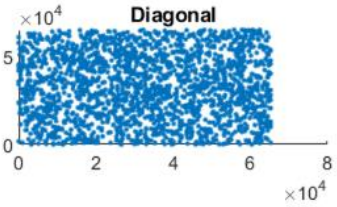
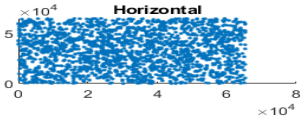
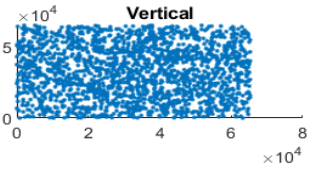
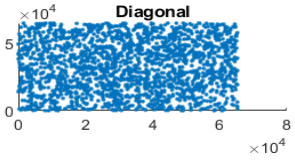
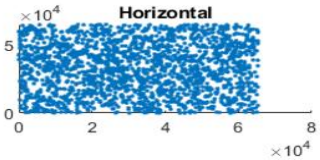
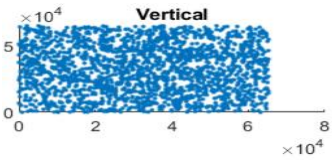
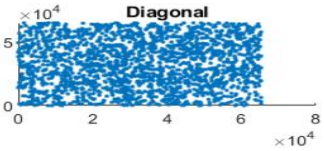
Plain image			
Logistic			
Piecewise			
Tent			
Chebyshev			
Circle			

Figure 7. Correlation analysis of the Cameraman image before and after encryption.

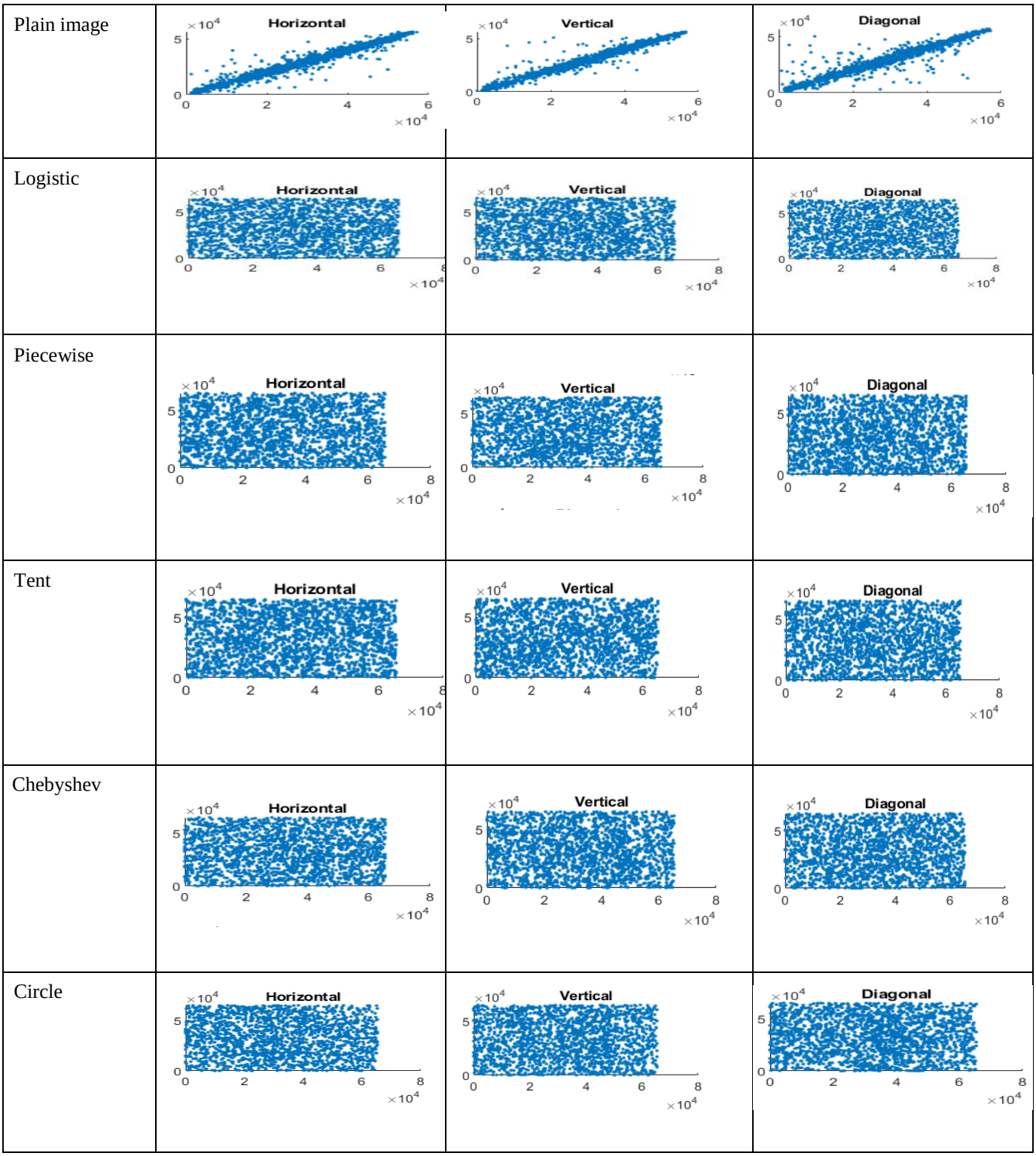


Figure 8: Correlation analysis of the Peppers image before and after encryption.

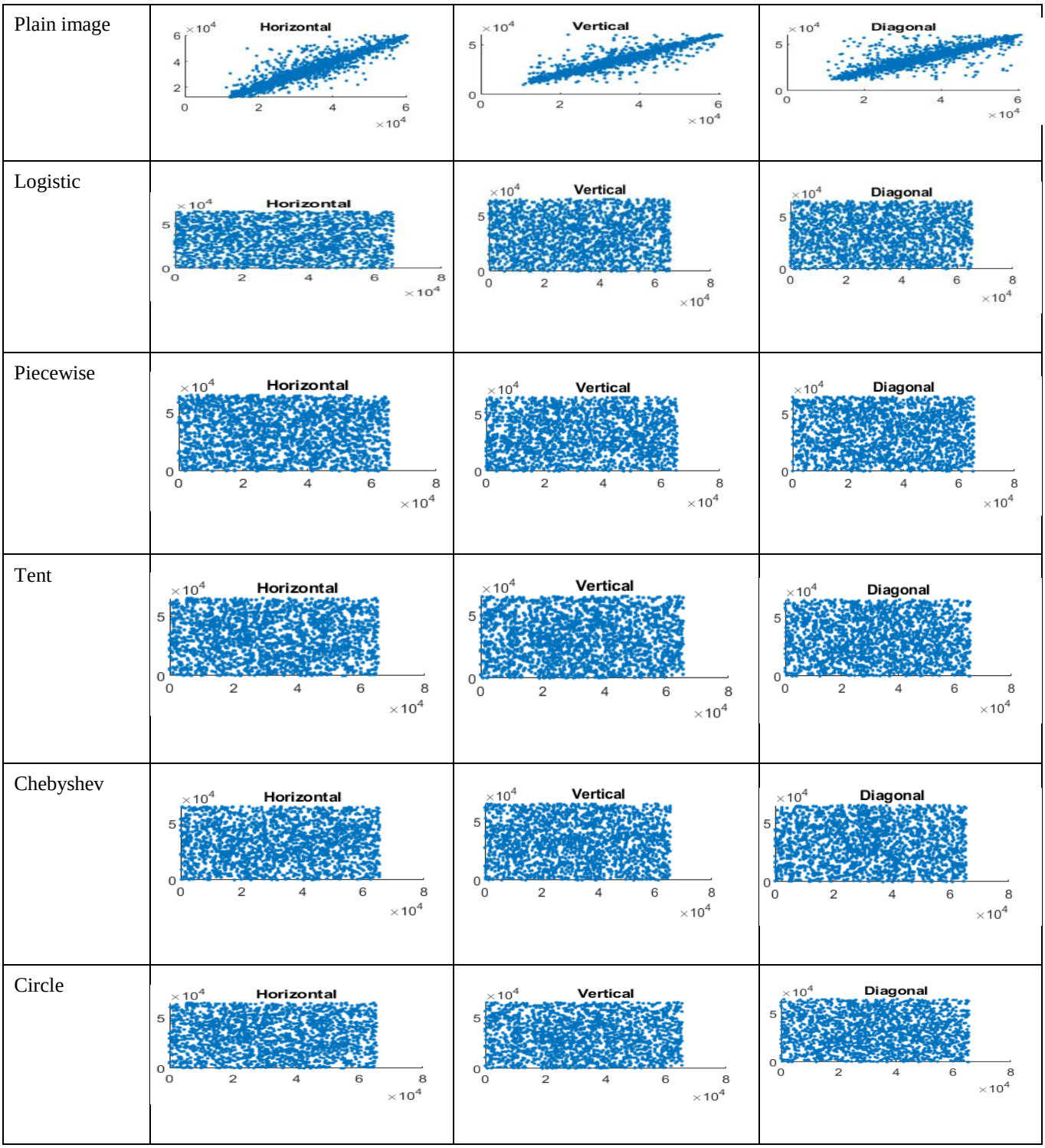


Figure 9. Correlation analysis of the Elaine image before and after encryption.

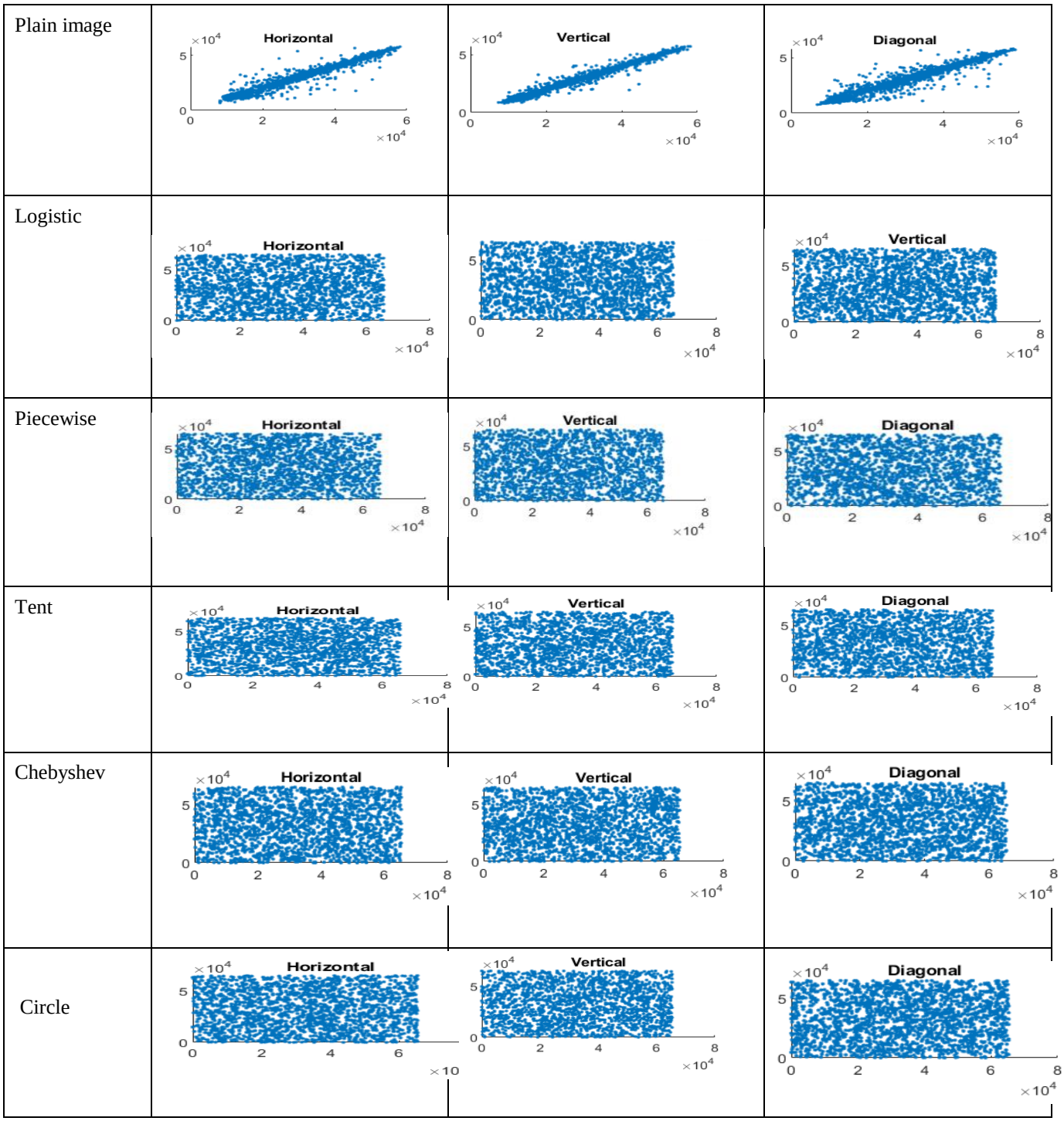


Figure 10. Correlation analysis of the Lena image before and after encryption.

Table 3. Correlation values between the suggested approach and other recent techniques for the encrypted Cameraman image.

algorithm	Horizontal	Vertical	Diagonal
proposed technique	-0.0034	-0.0043	-0.0012
[32]	0.0044	-0.0041	0.0016
[33]	0.0002	-0.0048	-0.0029
[34]	-0.0031	-0.0006	0.0011

Table 4. correlation values for the encrypted Peppers image between the proposed method and other current methods.

algorithm	Horizontal	Vertical	Diagonal
proposed technique	-0.0015	-0.0014	-0.0019
[33]	0.042	0.0053	-0.0032
[35]	0.0046	0.0039	-0.0018
[36]	-0.0020	0.00044	0.004
[37]	0.0295	0.0187	0.0393
[38]	-0.0112	-0.0047	0.0053
[39]	0.0046	0.0089	0.0128
[40]	-0.0036	0.0023	0.0022
[41]	0.0002	-0.0018	0.0014

Table 5. correlation values for the encrypted Elaine image between the proposed method and other current methods.

algorithm	Horizontal	Vertical	Diagonal
proposed technique	-0.005	-0.0024	-0.0036
[41]	0.0019	0.0008	0.0010
[42]	0.0014	0.0007	-0.0013
[43]	0.0048	-0.0041	-0.0036
[33]	-0.0061	-0.00007	0.0007
[37]	0.0005	-0.0133	0.0419

Table 6. correlation values for the encrypted Lena image between the proposed method and other current methods.

algorithm	Horizontal	Vertical	Diagonal
proposed technique	-0.0028	-0.0041	-0.056
[35]	0.0008	0.0004	0.002
[44]	-0.0007	0.00001	-0.0010
[40]	0.0002	0.0022	-0.001
[43]	0.0026	0.0006	0.0005
[45]	0.0068	0.000	-0.0028
[41]	0.0007	-0.0004	0.0011
[33]	0.0020	-0.0005	-0.0009
[34]	0.0040	-0.0012	-0.0021

5.1.1.4 Peak signal-to-noise ratio: is a metric commonly employed to assess the quality of an encrypted image in relation to the original image. It is necessary to compute MSE before calculating PSNR.

$$\mathrm{MSE} = \frac{1}{\mathrm{A}*\mathrm{B}}\sum_{\mathrm{a}=1}^{\mathrm{A}}\sum_{\mathrm{b}=1}^{\mathrm{B}}|\mathrm{A}(\mathrm{a},\mathrm{b}) - \mathrm{A}_{\mathrm{d}}(\mathrm{a},\mathrm{b})|^2 \tag{9}$$

$$PSNR = 10 * log_{10}\left(\frac{255^2}{MSE}\right) \tag{10}$$

When the original image is represented by A (a, b) and the decrypted image is represented by A_d(a, b), with (A, B) representing the image dimensions. PSNR is used as a metric of image similarity. The goal of encryption is to change the data into an unidentifiable form, and a high PSNR value suggests that the encrypted image closely matches the original. Therefore, it is preferable for encryption systems to achieve a lower PSNR. The PSNR value was consistently recorded at 99% for all standard test images employed, provided that the Mean Squared Error (MSE) was less than zero. This demonstrates how resilient the proposed method is to various forms of attacks by demonstrating how closely the decrypted images resemble the originals.

5.1.1.5 Computational speed analysis: assesses the speed at which an encryption or decryption method processes images. This analysis is essential to evaluating the encryption technique's effectiveness and usefulness, especially in real-time or resource-constrained contexts.

Table 7. compares the time execution of the method that is being presented with other algorithms.

Encryption technique	Execution time
suggested (Cameraman)	**0.3651**
[42]	**0.5539**
[46]	**4.711**
suggested (Peppers)	**0.1525**
[47]	**0.5474**
[48]	**0.1563**
[46]	**5.2199**
[36]	**0.27444**
[42]	**0.5474**
suggested (Elaine)	**0.169**
[48]	**0.176**
[43]	**1.3105**
suggested (Lena)	**0.1653**
[47]	**0.25**
[48]	**0.179**
[44]	**1.1044**
[40]	**0.62**
[42]	**0.5454**

Table 7 shows the execution times, which show that the suggested model's computational complexity has decreased. When the results are compared, it can be seen that the circle map takes the longest to process, while the sine map takes the quickest.

5.2 Robust analysis

refers to assessing the encryption algorithm's resilience to different issues, threats, or unfavorable circumstances while preserving its security and functionality.

5.2.1 Noise Attack Analysis

The quality of an image can be decreased by noise, which is defined by erratic variations in pixel intensity. A variety of noise types, including Gaussian noise with a variance of 0.02, salt-and-pepper noise with an intensity of 0.05, speckle noise, and Poisson noise, were added to the encrypted images in order to assess the efficacy of the suggested approach. This procedure is shown in Figure 11, where the matching decrypted images are shown in subfigures [V]through [VIII], and the encrypted images with additional noise are shown in subfigures [I] through [IV]. Figure 11's results demonstrate that the algorithm effectively fends off noise-based attacks, offering a strong defense against these distortions.

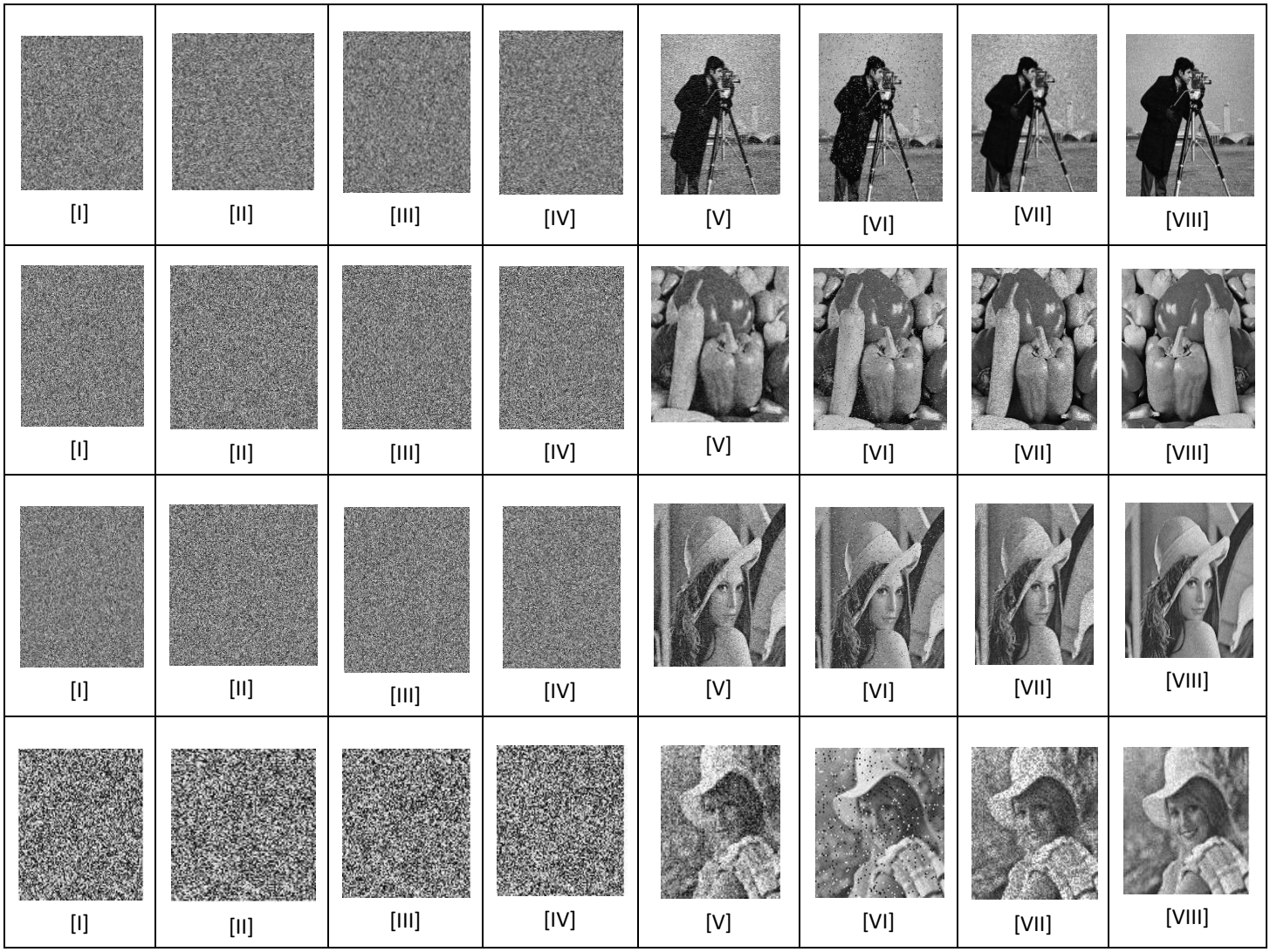


Figure 11. shows an encrypted image that has been exposed to various noise types: [I] Gaussian noise (mean = 0 and variance = 0.002), [II] salt-and-pepper noise (intensity = 0.04), [III] speckle noise, and [IV] Poisson noise.The matching decrypted pictures are shown as follows: Poisson noise is denoted by [VIII], Gaussian noise by [V], salt-and-pepper noise by [VII], and speckle noise by [VIII].

5.2.2 Cropping Analysis

By deleting portions of the encrypted image, a cropping attack in image encryption aims to reveal information about the original image. Before the encrypted images were decrypted for this investigation, various portions of them were cropped at varied percentages. Figure 12 displays the results of the cropped cipher images after decryption.

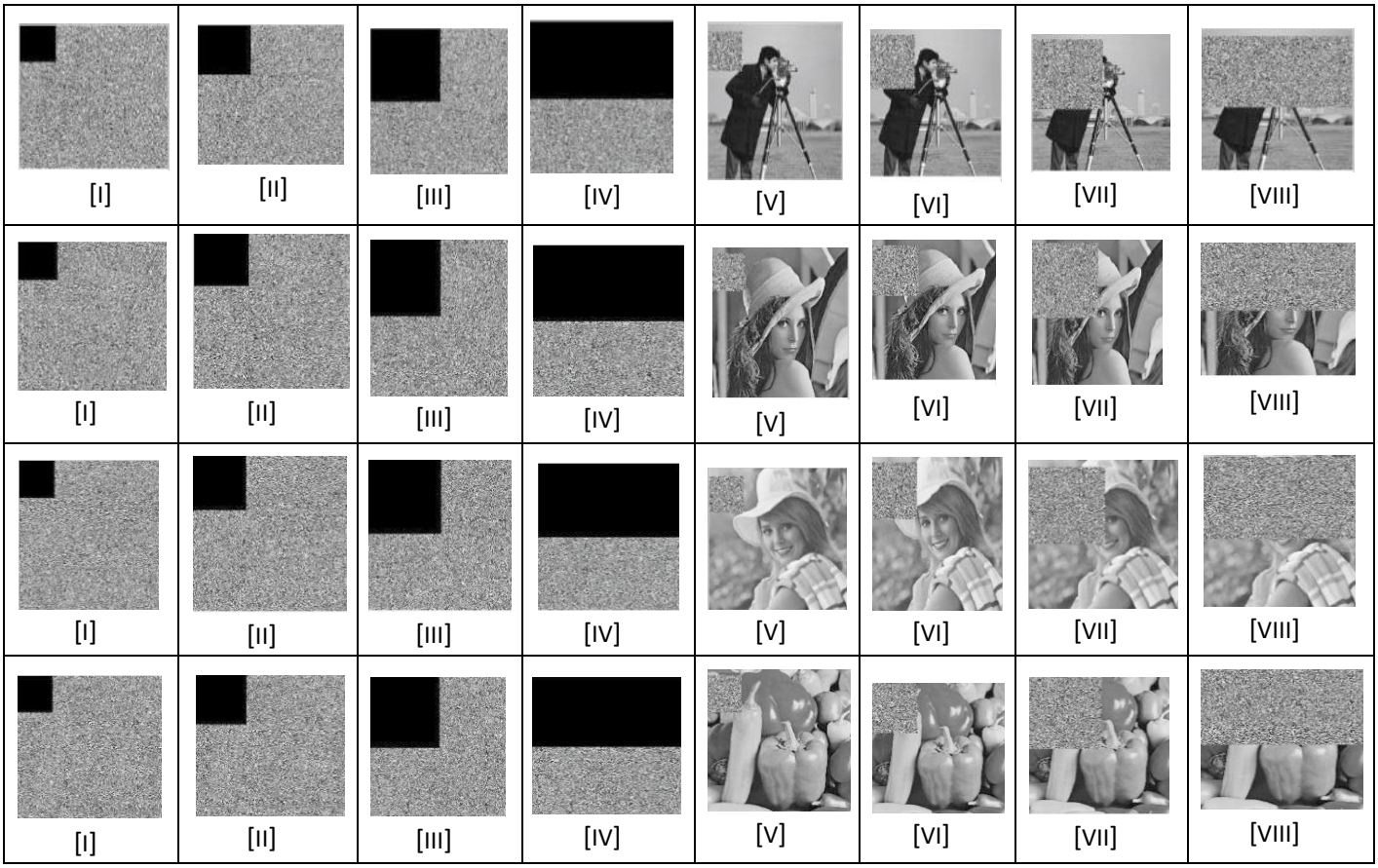


Figure 12. shows the effect of cropping on cipher images at different percentages—[I] 6.5%, [II] 13%, [III] 26%, and [IV] 52%. For these clipped sections, the matching decrypted images are displayed as [V] for 6.5%, [VI] for 13%, [VII] for 26%, and [VIII] for 52%.

5.3 Key Analysis

5.3.1 Key Space Analysis: The key space determines how difficult it is to crack the encryption using key guessing. The strength and security of the encryption are increased by a bigger key space, which makes it more difficult for an attacker to figure out the proper key even if they are aware of the encryption technique. The secret keys in suggested algorithm are initial conditions and control parameters. The circle map yields a key space of $(10^{16})^5$ with three variables for control and two initial conditions for change. This large key space guarantees the algorithm's resilience to several types of attacks. Table 8 contrasts the key spaces of several chaotic maps and table 9 contrasts the key space of several chaotic algorithms, emphasizing how robust the suggested approach is against numerous

Table 8. contrasts the key spaces of several chaotic map

Map	Number of control Parameters and initial conditions	Space key
Logistic Map	2 + 2 = 4	$((10)^{16})^{4}$
Tent Map	2 + 2 = 4	$((10)^{16})^{4}$
Piecewise Map	2 + 2 = 4	$((10)^{16})^{4}$
Circle Map	2 + 3 = 5	$((10)^{16})^{5}$
Chebyshev Map	2 + 2= 4	$((10)^{16})^{4}$

table 9. contrasts the key space of several chaotic algorithms

Algorithm	Key space
proposed algorithm	$(10^{16})^{5} = 10^{80}$
[35]	10^{64}
[36]	10^{36}
[39]	10^{56}
[44]	10^{45}
[49]	10^{42}
[47]	10^{40}

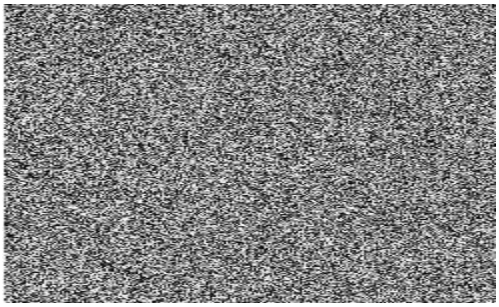
The suggested model's potential key space is shown in Tables 9 and 10, where a total of 10^{80} variable keys are found. Predicting the encryption key is made exceedingly difficult and unlikely by this enormous key space.

5.3.2 Key Sensitivity Analysis

In addition to key space, key sensitivity is a crucial component for secure image encryption. It gauges how much even the slightest alteration to the secret encryption key affects the encrypted output.

For example, have a look at the decryption key X = 0.866. This key ensures that the encrypted image is exactly the same as the plain image. The decoded image, however, differs greatly from the original if the key is slightly changed to X=0.8660000000000000001. This is seen in Figure 13: Whereas (b) illustrates the decryption outcome with a minor key modification, (a) displays the decryption using the appropriate key.

[I]



[II]

5.3.3 Brute-Force Attack

Extreme sensitivity to changes in the secret key is necessary for an encryption algorithm to resist brute-force attacks. An adversary methodically checks every potential key in such attacks until the right one is found. However, because there are so many possible keys, brute-force methods become extremely ineffective and time-consuming when the key space is large.

An illustrative example:

Imagine that there are countless key combinations for a lock. Finding the right key through trial and error is more difficult as the number of choices increases. This idea is the cornerstone of brute-force defense.

5.4 Differential Analysis

Important metrics for evaluating an algorithm's defense against differential attacks are NPCR and UACI. To calculate these indications, two encrypted images, $\boldsymbol{E_1}$ and $\boldsymbol{E_2}$ are needed.

5.4.1 Unified Average Changing Intensity (UACI)

By measuring the difference in intensity values between comparable pixels in two encrypted images, this measure assesses how well image encryption techniques work. A single pixel alteration in the source image causes this variation. The following formula can be used to calculate the UACI:

$$\text{UACI} = \frac{1}{A*B}\left[\sum_{a,b}\frac{\left|E_{1(a,b)}-E_{2(a,b)}\right|}{255}\right] * 100\% \qquad (11)$$

In this case, A*B stands for the image's dimensions, and E(a, b) indicates the value of the pixel at position (a, b) in the encrypted image $E*$. When an encryption algorithm's UACI value becomes close to 33.46%, it is deemed very sensitive. This suggests that even little changes to the source image result in significant variations in the encrypted image's average intensity.

5.4.2 Number of Pixel Change Rate (NPCR)

The NPCR measure is used to assess the rate of pixel changes between the plain image and its encrypted counterpart. The NPCR value is calculated using the formula below.:

$$\text{NPCR}= \frac{1}{A*B}\sum_{c,d} D(c,d) * 100\% \tag{12}$$

In which

$$\text{D (c, d)} = \begin{cases} 0 & if \quad E_1(c,d) = E_2(c,d) \\ 1 & if \quad E_1(c,d) \neq E_2(c,d) \end{cases} \tag{13}$$

The images' dimensions are indicated in this context by A×B, and the pixel at image E's (c, d) coordinates is referred to as E (c, d). The highest level of resistance against differential attacks is ideally indicated by an NPCR rating of 100%. When an encryption technique has an NPCR score near 100%, it is deemed safe against these types of attacks.

Table 10 displays a comparison of the NPCR of the cipher image.

Image	Cameraman	Peppers	Lena	Elaine
proposed technique	0.99631	0.99415	0.9932	0.99259
[50]	0.99630	___	0.99620	___
[41]	___	0.99620	0.99628	0.99618
[51]	0.99630	___	0.99620	___
[33]	___	0.99610	0.9960	0.99650

Table 11. display a comparison of the UACI of the cipher image

Image	Cameraman	Peppers	Lena	Elaine
proposed technique	32.8201	32.7270	32.9242	32.4553
[49]	___	34.4092	34.7772	___
[35]	___	32.6708	___	___
[52]	___	30.7975	28.7491	___
[53]	25.492	___	___	___
[50]	31.2802	___	30.7972	___

6. Conclusions

In this work, a chaotic map-based technique that incorporates five distinct chaotic maps is used to investigate image encryption. Several performance parameters, such as entropy execution, time, correlation coefficient, resistance to noise attacks, key sensitivity, and PSNR, were assessed after the suggested technique was applied to standard images. The execution time of the encryption operation was 0.1525 ms. The encrypted images' histogram analysis showed almost uniform frequency distributions, demonstrating the algorithm's resilience to a variety of attacks. High security was shown by the correlation coefficient being near zero, and well-randomized image data was indicated by the entropy value of 7.9985.

To create more robust and secure encryption methods, future studies can concentrate on incorporating more chaotic maps and investigating novel ones. These developments will open the door for creative encryption techniques and further strengthen the defense of digital data from malevolent threats.

References

1. A. Pourjabbar Kari, A. Habibizad Navin, A. M. Bidgoli, and M. Mirnia. A new image encryption scheme based on hybrid chaotic maps. Multimed Tools Appl 2021, vol. 80, pp. 2753–2772.
2. G. Cheng, C. Wang, and H. Chen. A novel color image encryption algorithm based on hyperchaotic system and permutation-diffusion architecture. International Journal of Bifurcation and Chaos 2019, vol. 29, no. 09, p. 1950115.
3. L. Gong, K. Qiu, C. Deng, and N. Zhou. An image compression and encryption algorithm based on chaotic system and compressive sensing. Opt Laser Technol 2019, vol. 115, pp. 257–267.
4. Z. A. Abduljabbar et al. Provably secure and fast color image encryption algorithm based on s-boxes and hyperchaotic map. IEEE Access 2022, vol. 10, pp. 26257–26270.
5. S. M. Sameh, H. E.-D. Moustafa, E. H. AbdelHay, and M. M. Ata, an effective chaotic maps image encryption based on metaheuristic optimizers. J Supercomput 2024, vol. 80, no. 1, pp. 141–201.
6. J. Arif et al. A novel chaotic permutation-substitution image encryption scheme based on logistic map and random substitution. IEEE Access 2022, vol. 10, pp. 12966–12982.
7. Hazzazi, M. M., Rehman, M. U., Shafique, A., Aljaedi, A., Bassfar, Z., & Usman, A. B. Enhancing image security via chaotic maps, Fibonacci, Tribonacci transformations, and DWT diffusion: a robust data encryption approach. Scientific Reports 2024, 14(1), 12277.
8. Mostafaee, J., Mobayen, S., Vaseghi, B., Vahedi, M., & Fekih, A. Complex dynamical behaviors of a novel exponential hyper–chaotic system and its application in fast synchronization and color image encryption. Science Progress 2021.
9. Liu, Z., Li, J., & Liu, J. Encrypted face recognition algorithm based on Ridgelet-DCT transform and THM. chaos.Math. Biosci. Eng 2022, 19(2), 1373-1387.
10. Yang, F., & An, X. A new discrete chaotic map application in image encryption algorithm. Physica Scripta 2022, 97(3), 035202.
11. Guo, Z., & Sun, P. Improved reverse zigzag transform and DNA diffusion chaotic image encryption method. Multimedia Tools and Applications 2022, 81(8), 11301-11323.
12. Hong, Y., Wang, Y., Su, J., Wen, Y., & Yang, Z. Image encryption algorithm based on chaotic mapping and binary bidirectional zigzag transform. IEEE Access 2023.
13. Zou, C., Shang, Y., Yang, Y., Zhou, C., & Liu, Y. A novel image encryption algorithm with anti-tampering attack capability. Chaos, Solitons & Fractals 2024, 189, 115638.
14. Wen, H., Lin, Y., & Feng, Z. Cryptanalyzing a bit-level image encryption algorithm based on chaotic maps. Engineering Science and Technology, an International Journal 2024, 51, 101634.

15. Elkhalil, N., Weddy, Y. C., & Ejbali, R. Image encryption using the new two-dimensional Beta chaotic map. Multimedia Tools and Applications 2023, 82(20), 31575-31589.
16. Yeh, T. C., & Kiang, J. F. Second-Order Chaotic Maps with Random Coefficients to Generate Complex Chaotic Sequences for High-Security Image Encryption. IEEE Access 2023.
17. Shao, S., Li, J., Shao, P., & Xu, G. Chaotic image encryption using piecewise-logistic-sine map. IEEE Access 2023, 11, 27477-27488.
18. Dua, M., & Bhogal, R. Medical Image Encryption using Novel Sine-Tangent Chaotic Map. e-Prime-Advances in Electrical Engineering, Electronics and Energy 2024, 100642.
19. Hazzazi, M. M., Rehman, M. U., Shafique, A., Aljaedi, A., Bassfar, Z., & Usman, A. B. Enhancing image security via chaotic
maps, Fibonacci, Tribonacci transformations, and DWT diffusion: a robust data encryption approach. Scientific Reports 2024, 14(1), 12277.
20. Alaklabi, A., Munir, A., Hafeez, M. A., & Khattak, M. A. K. Z-Crypt: Chirp Z-Transform-Based Image Encryption Leveraging
Chaotic Logistic Maps and Substitution Permutation Network. IEEE Access 2024.
21. Vijayakumar, M., & Ahilan, A. An optimized chaotic S-box for real-time image encryption scheme based on 4-dimensional memristive hyperchaotic map. Ain Shams Engineering Journal 2024, 15(4), 102620.
22. N. Munir, M. Khan, A. Al Karim Haj Ismail, and I. Hussain, Cryptanalysis and improvement of novel image encryption technique using hybrid method of discrete dynamical chaotic maps and brownian motion. Multimed Tools Appl 2022, vol. 81, no. 5, pp. 6571–6584.
23. X. Wang, S. Chen, and Y. Zhang. A chaotic image encryption algorithm based on random dynamic mixing. Opt Laser Technol 2021, vol. 138, p. 106837.
24. S. Arora and P. Anand,.Chaotic grasshopper optimization algorithm for global optimization. Neural Comput Appl 2019, vol. 31, pp. 4385–440.
25. Q. Xu, K. Sun, C. Cao, and C. Zhu, .A fast image encryption algorithm based on compressive sensing and hyperchaotic map. Opt Lasers Eng 2019, vol. 121, pp. 203–214.
26. S. Xu, C.-C. Chang, and Y. Liu,. A high-capacity reversible data hiding scheme for encrypted images employing vector quantization prediction. Multimed Tools Appl 2021, vol. 80, no. 13, pp. 20307–20325.
27. Ghaffari, Aboozar. Image encryption-compression method via encryption based sparse decomposition. Multimedia Tools and Applications, 2024.
28. R. Premkumar and S. Anand,.Secured and compound 3-D chaos image encryption using hybrid mutation and crossover operator. Multimed Tools Appl 2019, vol. 78, pp. 9577–9593.
29. S. Xu, C.-C. Chang, and Y. Liu, .A high-capacity reversible data hiding scheme for encrypted images employing vector quantization prediction, Multimed Tools Appl 2021, vol. 80, no. 13, pp. 20307–20325.
30. Güvenoğlu, Erdal. An image encryption algorithm based on multi-layered chaotic maps and its security analysis. Connection Science 2024.
31. Tang, Jianeng, et al. Novel asymmetrical color image encryption using 2D sine-power coupling map. Nonlinear Dynamics (2024), p1-23.
32. X. Liu, X. Tong, Z. Wang, and M. Zhang, A new n-dimensional conservative chaos based on Generalized Hamiltonian System and its' applications in image encryption, Chaos Solitons Fractals 2022, vol. 154, p. 111693.
33. M. Ahmad et al., .An image encryption algorithm based on new generalized fusion fractal structure. Inf Sci (N Y) 2022, vol. 592, pp. 1–20.

34. S. Zhang and L. Liu, A novel image encryption algorithm based on SPWLCM and DNA coding. Math Comput Simul 2021, vol. 190, pp. 723–744.
35. B. Mondal, P. K. Behera, and S. Gangopadhyay, .A secure image encryption scheme based on a novel 2D sine–cosine cross-chaotic (SC3) map. J Real Time Image Process 2021, vol. 18, no. 1, pp. 1–18.
36. X. Wang and S. Gao, .Image encryption algorithm for synchronously updating Boolean networks based on matrix semi-tensor product theory. Inf Sci (N Y) 2020, vol. 507, pp. 16–36.
37. X. Liu, D. Xiao, and C. Liu. Three-level quantum image encryption based on Arnold transform and logistic map. Quantum Inf Process 2021, vol. 20, pp. 1–22.
38. M. Munoz-Guillermo, .Image encryption using q-deformed logistic map. Inf Sci (N Y) 2021, vol. 552, pp. 352–364.
39. X. Wang and H. Sun, A chaotic image encryption algorithm based on improved Joseph traversal and cyclic shift function, Opt Laser Technol 2020, vol. 122, p. 105854.
40. N. Khalil, A. Sarhan, and M. A. M. Alshewimy,.An efficient color/grayscale image encryption scheme based on hybrid chaotic maps. Opt Laser Technol 2021, vol. 143, p. 107326.
41. K. A. K. Patro and B. Acharya, .An efficient dual-layer cross-coupled chaotic map security-based multi-image encryption system. Nonlinear Dyn 2021, vol. 104, no. 3, pp. 2759–2805.
42. Z. Guo and P. Sun, Improved reverse zigzag transform and DNA diffusion chaotic image encryption method. Multimed Tools Appl 2022, vol. 81, no. 8, pp. 11301–11323.
43. X. Wang, N. Guan, and P. Liu, .A selective image encryption algorithm based on a chaotic model using modular sine arithmetic Optik (Stuttg) 2022, vol. 258, p. 168955.
44. X. Wang and S. Gao, .A chaotic image encryption algorithm based on a counting system and the semi-tensor product. Multimed Tools Appl 2021, vol. 80, no. 7, pp. 10301–10322.
45. M. A. Midoun, X. Wang, and M. Z. Talhaoui, .A sensitive dynamic mutual encryption system based on a new 1D chaotic map, Opt Lasers Eng 2021, vol. 139, p. 106485.
46. X. Yan, X. Wang, and Y. Xian, .Chaotic image encryption algorithm based on arithmetic sequence scrambling model and DNA encoding operation. Multimed Tools Appl 2021, vol. 80, pp. 10949–10983.
47. M. Lyle, P. Sarosh, and S. A. Parah, .Adaptive image encryption based on twin chaotic maps. Multimed Tools Appl 2022, vol. 81, no. 6, pp. 8179–8198.
48. D. Zareai, M. Balafar, and M. R. Feizi Derakhshi, .A new Grayscale image encryption algorithm composed of logistic mapping, Arnold cat, and image blocking. Multimed Tools Appl 2021, vol. 80, pp. 18317–183440.
49. A. Jan, S. A. Parah, and B. A. Malik, .EFHAC: Image encryption framework based on hessenberg transform and chaotic theory for smart health,. Multimed Tools Appl 2022, vol. 81, no. 13, pp. 18829–18853.
50. Kamrani, Abdelhalim, Khalid Zenkouar, and Said Najah. .A new set of image encryption algorithms based on discrete-orthogonal moments and Chaos theory. Multimedia Tools and Applications 2020.
51. Zhao, Hongxiang, Shucui Xie, and Jianzhong Zhang. Fast image encryption based on new cascade chaotic system and Rubik's cube strategy. Multimedia Tools and Applications 2023.
52. H. Wu, F. Li, C. Qin, and W. Wei, Separable reversible data hiding in encrypted images based on scalable blocks, Multimed Tools Appl 2019, vol. 78, pp. 25349–25372.
53. Ghaffari, Aboozar. Image encryption-compression method via encryption based sparse decomposition. Multimedia Tools and Applications 2024.