Journal of King Saud University – Computer and Information Sciences 35 (2023) 101613

HOSTED BY

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

# Efficient and secure image encryption using key substitution process with discrete wavelet transform

Mujeeb Ur Rehman [a], Arslan Shafique [b,*], Kashif Hesham Khan [c], Mohammad Mazyad Hazzazi [d]

[a] School of Science, Technology and Health, York St John University, York, UK
[b] Department of Computer Science (Cyber Security Research Group), York St John University, York, UK
[c] School of Computing Technologies, STEM College, RMIT University, Melbourne, VIC 3000, Australia
[d] Department of Mathematics, College of Science, King Khalid University, Abha 61413, Saudi Arabia

ARTICLE INFO

*Article history:*
Received 12 March 2023
Revised 4 May 2023
Accepted 4 June 2023
Available online 15 June 2023

*Keywords:*
Data privacy
Key scheming
Substitution box
Chaos
Cyber security

ABSTRACT

Over the past few years, there has been a rise in the utilization of chaotic encryption algorithms for securing images. The majority of chaos-based encryption algorithms adhere to the conventional model of confusion and diffusion, which typically involves either implementing multiple encryption rounds or employing a single round of intricate encryption to guarantee robust security. However, such kind of approaches reduces the computational efficiency of the encryption process but compromises security. There is a trade-off between security and computational efficiency. Prioritizing security may require high computational processes. To overcome this issue, a key substitution encryption process with discrete wavelet transform (KSP-DWT) is developed in the proposed image encryption technique (IET). Based on KSP-DWT and IET, the abbreviation of the proposed work is used in this paper as KSP-DWT-IET. The proposed KSP-DWT algorithm employs a key scheming technique to update the initial keys and uses a novel substitution method to encrypt digital images of different sizes. Additionally, the integration of DWT can result in the compression of frequency sub-bands of the source image, leading to lower computational overheads without compromising the security of the encryption. The KSP-DWT-IET performs a single encryption round and is highly secure and efficient. The simulation results and security analysis conducted on KSP-DWT-IET confirm its effectiveness in ensuring high-security image encryption while minimizing computational overhead. The proposed encryption technique undergoes various security analyses, including entropy, contrast, correlation, energy, NPCR (Number of Pixel Changes Rate), UACI (Unified Average Change Intensity) and computational complexity. The statistical values obtained for such parameters are 7.9991, 10.9889, 0.0001, 0.0152, 33.6767, and 33.6899, respectively, which indicate that the encryption technique performs very well in terms of security and computational efficiency. The proposed encryption scheme is also analyzed for its computational time in addition to its security. The analysis shows that the scheme can efficiently encrypt images of varying sizes with a high level of security in a short amount of time (i.e., 2 ms). Therefore, it is feasible to use this encryption scheme in real-time applications without causing any significant delays. Moreover, the key space of the proposed encryption scheme is large enough (i.e. Keyspace $> 2^{100}$) to resist the brute force attack.


## 1. Introduction

Digital images have become a crucial part of modern society due to their ability to provide clear and easy-to-understand descriptions. However, with the increasing amount of secret or private information contained within digital images, it has become important to consider their security. To ensure the security of data, conventional encryption algorithms, including AES, IDEA, and DES, have been devised, but due to the substantial amount of content present in digital images, these conventional ciphers are not efficient for encrypting them. Therefore, researchers have proposed

* Corresponding author.
*E-mail address:* Arslanshafique762@gmail.com (A. Shafique).
Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2023.101613
1319-1578/

various image encryption algorithms that use alternative encryption techniques to achieve reasonable encryption performance for images.

There are many different types of algorithms used to encrypt images. Some of these algorithms are based on Sudoku matrices, Latin squares (Parameshachari et al., 2019; Diaconu, 2014), DNA (Chen et al., 2020) and chaos theory (Wang et al., 2022; Rehman et al., 2022). Most image encryption methods follow a structure called confusion-diffusion, which was first introduced by Fridrich (Fridrich, 1998). Confusion refers to changing the positions of the pixels, while diffusion alters the pixel values. This classical encryption process is illustrated in Fig. 1.

The distinct attributes of chaotic systems, such as sensitivity to initial conditions, unpredictability and ergodicity, have drawn the attention of researchers toward using chaos-based approaches for image encryption. These traits make chaotic systems well-suited for developing secure image encryption algorithms.

Encryption algorithms can be broadly categorized based on the techniques they use to achieve a high level of security. Employing several rounds of confusion and diffusion is a method for achieving a high level of security in encrypted data (Alawida et al., 2022). In each round of these algorithms, the encryption operation is repeatedly applied in the same manner. One example from this category is the encryption algorithm proposed in Hua et al. (2015), which uses a combination of a new two-dimensional Sine Logistic modulation map (2D-SLMM) and a chaotic magic transform (CMT) to encrypt digital images. This algorithm involves two rounds of encryption. In Hua and Zhou (2016), Hua et al. introduced a method for encrypting images that uses the 2D-LASM (a 2-dimensional logistic-adjusted sine map) (Sharma, 2020) and performs two rounds of confusion diffusion. The security analysis suggests that this approach is highly resistant to various security attacks. Similarly, in Zhu et al. (2019), Zhu et al. proposed another method for image encryption that involves two rounds of permutation and diffusion operations, using a new chaotic map called LSMCL (Logistic modulated Sine-Coupling-Logistic) (Wang and Guan, 2022). Both theoretical analysis and simulations showed the effectiveness algorithm in terms of image security. In Ping et al. (2018), Ping et al. described an encryption algorithm that combines three rounds of permutations and the remaining three rounds of diffusion to secure digital images. The algorithm is reported to have high security and efficiency, as supported by simulation results. While in Diab (2018), Diab and Hossam proposed an image cryptosystem that uses the simultaneous process of diffusion and permutation processes which contain two encryption rounds.

The second category of encryption algorithms implements only one round of diffusion and one round of permutation, and their security is enhanced through complex operations (Hoang, 2022).

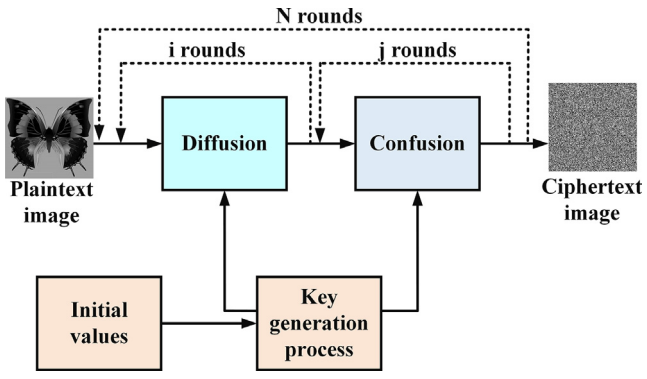



**Fig. 1.** A classical image encryption process.

In Pak and Huang (2017), Pak et al. proposed a novel encryption scheme that utilizes a new 1-D chaotic system (Rehman et al., 2021), which has been shown to perform well against various attacks through experimental demonstration. In Ye and Huang (2017), Ye et al. presented another algorithm for securing digital images that employ complex diffusion and permutation operations to enhance the security of the digital images where the keystream for encryption is dependent on the plain image (Huang and Ye, 2014). The security of the cipher has been demonstrated through numerical experiments and security analysis. Another image encryption technique is introduced in Amina and Mohamed (2018), which incorporates an enhanced one-dimensional chaotic system and SHA-256 to create initial conditions (Liao et al., 2016). To improve encryption performance, this algorithm incorporated the pixel confusion-diffusion phenomenon. In Chai et al. (2022a), Chai et al. presented a novel image encryption approach that employs multi-objective optimization and incorporates block-compressed sensing to enhance the security of digital images. In order to introduce diffusion in color images, they utilized a random sequence generated through the Henon chaotic map. Additionally, they used Hessenberg Decomposition (HD) to embed secret images into the carrier image. The proposed encryption scheme is found to be robust based on their analysis. Another work by Chai et al. (2022b) presents a solution to the vulnerabilities present in Thumbnail Preserving Encryption (TPE) which is based on sum-preserving encryption through the introduction of a new approach called TPE that is based on GAN with key. In Faragallah et al. (2020a), Faragallah et al. utilized the confusion methods of Cat, Baker, and logistic maps in both the spatial and frequency domains to evaluate the effectiveness of image cryptosystems based on chaos for cybersecurity and encryption purposes. In Alarifi et al. (2020), Alarifi et al. introduced a unique hybrid cryptographic system that integrates DNA sequences, Mandelbrot sets and Arnold chaotic map. This system is designed for the secure transmission of compressed HEVC streams. In Faragallah et al. (2021), Faragallah et al. introduced an efficient and secure encryption scheme for protecting color images. This technique employs the use of 2D fractional Fourier transform and 2D logistic mapping. Additionally, the security of the plaintext color images is further improved by implementing double confusion in color images. In Faragallah et al. (2018), Faragallah et al. investigated and presented a block-based opto-color cipher using double random phase encoding (DRPE) with various block sizes. In El-Shafai et al. (2021), Shafai et al. presented an efficient cryptosystem for medical image security by leveraging the benefits of deoxyribonucleic acid (DNA) rules and chaos maps. In Faragallah et al. (2020b), Faragallah et al. proposed an efficient Fractional Fourier Transform (FrFT)-based logistic map (LM) color image encryption scheme, which applies a 2D LM on FrFT. In El-Shafai et al. (2022a), Shafai et al. presented a medical image cryptosystem that utilizes a Stacked Auto-Encoder (SAE) network to produce two sets of chaotic random matrices. In Faragallah et al. (2020c), Faragallah et al. also presented an efficient color image cryptosystem based on RC6 with different modes of operation in their paper. In El-Shafai et al. (2021), Shafai et al. introduced an optical HEVC cipher algorithm based on bit-plane 3D-JST and multistage 2D-FrFT encryption, where 3D-JST has an inverse transform used to reorganize the HEVC frame-blocks in an indiscriminate way. In El-Shafai et al. (2021), Elashry et al. proposed a design of a 2-D chaotic Baker map for image encryption that utilizes three modes of operations: cipher block chaining (CBC) mode, cipher feedback (CFB) mode, and output feedback (OFB) mode. In El-Shafai et al. (2022a), Shafai et al. proposed a robust cryptosystem based on a 3D chaotic map for medical image encryption in secure IoMT and cloud services. In Alqahtani et al. (2022), Alqahtani et al. proposed an optical medical image security approach that is based on the optical bit-plane

Jigsaw Transform (JT) and Fractional Fourier Transform (FFT). In El-Shafai et al. (2022b), Shafai et al. proposed an asymmetric PTFrFT-based color medical image cryptosystem, where two different phases in the fractional Fourier and output planes are provided as deciphering keys.

In Cao et al. (2018), Cao et al. proposed a new approach for image encryption is introduced that leverages a newly developed 2-D chaotic map to implement bit-level confusion and diffusion techniques simultaneously (Chai, 2017). This algorithm updated the initial values of the chaotic system based on the obtained ciphertext. Performance analysis showed that this algorithm achieved satisfactory encryption performance and high efficiency. In Tong et al. (2015), Tong et al. present a methodology for encrypting color images using a chaotic system with four dimensions. This algorithm is designed to provide enhanced security and speed compared to previous methods. The algorithm is created with the aim of providing security, robustness, and efficiency.

### 1.1. Motivation

In today's digital age, the transmission and storage of sensitive information have become an essential part of everyday life. As a result, the need for secure data encryption techniques has become increasingly critical. Cyber-attacks and data breaches are becoming more sophisticated and frequent, making it more challenging to keep sensitive information safe from unauthorized access. According to a report by Risk Based Security, in 2021, over 1.5 billion records were compromised in data breaches globally. Furthermore, cyber-attacks are becoming more diverse, with phishing scams, ransomware, and distributed denial-of-service (DDoS) attacks becoming more prevalent.

Image encryption is a crucial part of secure data transmission, and traditional encryption techniques have their limitations. Therefore, there is a need for more efficient and secure image encryption techniques to protect against cyber-attacks. The Key Substitution Process with Discrete Wavelet Transform (DWT) is a promising technique that can offer robust and efficient image encryption. It uses a combination of substitution and transformation techniques to encrypt images, making it difficult for attackers to decipher the encrypted data. Thus, the development of an efficient and secure image encryption technique using the Key Substitution Process with DWT can address the increasing need for secure data transmission and storage, and mitigate the risks of cyber-attacks and data breaches.

### 1.2. Problem statement and contributions of the paper

Image encryption techniques that rely on chaos theory usually employ either multiple encryption rounds or a single encryption round to protect digital images. The chaos-based encryption techniques are based on the traditional confusion-diffusion framework, where the image pixels are randomly shuffled or rearranged (confusion) and then spread across the image to make it appear more chaotic (diffusion). However, our analysis suggests that these methods are not secure enough or efficient. In the proposed work, a new approach called KSP-DWT-IET is proposed which utilizes a single encryption round. This novel approach overcomes the security and efficiency issues observed in traditional methods, offering a more reliable and effective encryption solution for images. Our contributions can be summarized as follows:

- The proposed encryption scheme presents a new method that effectively overcomes the limitations of the traditional confusion-diffusion structure. This is achieved by implementing a key-substitution approach, which significantly improves both the security and efficiency of the encryption process.
- A new key scheming approach is also developed that utilizes weighted addition to improve the algorithm's sensitivity to small variations in the plain image. By utilizing this approach, the encryption scheme is strengthened against various types of cyberattacks, including entropy attacks, brute force attacks, and differential attacks. This results in a more robust and secure encryption solution.
- To enhance the encryption process even further, a novel substitution method is developed that combines several techniques, including random grouping and S-box construction, with random substitution. Through this approach, we achieve highly secure encryption of the plain image while maintaining optimal performance.
- The implementation of the proposed KSP-DWT-IET scheme involves a new methodology for substituting image pixels that utilize DWT. This method involves decomposing the manipulated image to the $5^{th}$ level using DWT, which significantly reduces the encryption computation time.
- To assess the efficacy of the proposed KSP-DWT-IET algorithm, a series of experimental tests such as differential attacks, cropping attack analysis, and computational time analysis is conducted.

The rest of the paper is structured as follows: Section 2 outlines the proposed key substitution process. Section 3 provides a detailed explanation of the method proposed to encrypt digital images. In Section 4, the experimental results and analysis for the proposed encryption scheme are presented, including various statistical analyses. Finally, Section 5 concludes the research work.

## 2. Key-substitution process

In this research, a new encryption architecture is proposed that uses a key-substitution approach to improve the security of the cryptosystem while reducing computational complexity. This KSP involves two main stages, namely key scheming and substitution. Unlike the traditional framework that relies on confusion-diffusion, the proposed approach employs chaos-based key scheming to enhance sensitivity to changes in the plaintext image. This helps to protect the encryption algorithm from known/chosen plaintext attacks. The substitution phase in image encryption algorithms involves replacing the values of all pixels using an S-box, which is based on chaos theory. This S-box makes use of chaos-based techniques to enhance the computational efficiency of the substitution phase. By using a chaos-based S-box, the substitution phase can be performed more quickly and with higher security, making it an essential part of many image encryption algorithms. Moreover, a DWT is employed to substitute the $5^{th}$ level decomposition band which also helps to reduce the computational time significantly. The proposed KSP enables satisfactory encryption performance with just one round of encryption. Fig. 2 illustrates the proposed generalized encryption process.

### 2.1. Key scheming

Key scheming is a process that creates a strong relationship between the plaintext information and the initial key in an encryption algorithm. The initial key values can be altered by including information from the plaintext information, which allows it to be updated according to the original image. In chaos-based image encryption algorithms, the initial key plays a vital role as it strongly affects the behavior of the chaotic system (Mondal and Mandal, 2017). If the initial key is changed with variations in the original image, it will result in a different key stream and therefore produce distinct ciphertext images (Liu et al., 2022). As a result,

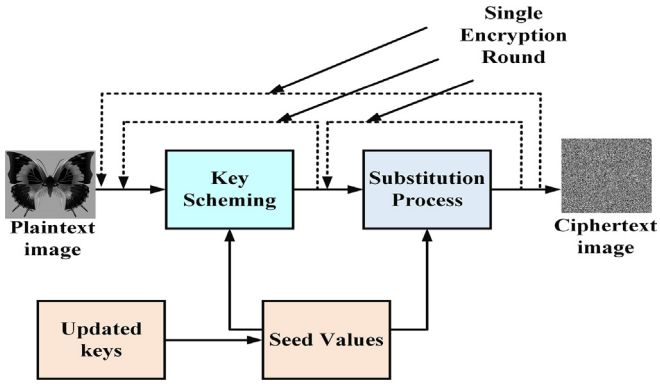



**Fig. 2.** Generalized proposed image encryption process.

key scheming is an effective means of ensuring sensitivity to changes in the original image and improving the security of a chaotic image encryption algorithm.

The level of sensitivity to plaintext changes in key scheming significantly affects the encryption performance of a cryptosystem (Nan et al., 2022). With strong key scheming, the system becomes more responsive to changes in the original image, thereby improving its ability to resist known or chosen plaintext attacks. This can be achieved by employing a simple key scheming operation, which can yield satisfactory encryption performance with just one round of encryption. Efficiency in encryption can be improved by employing a key-scheming technique that requires fewer encryption rounds (Gondal and Anees, 2013). Using this approach eliminates the necessity for numerous iterations of encryption operations or a single round of intricate encryption operations. This, in turn, leads to quicker and more effective encryption. Thus, an excellent key scheming method can optimize encryption efficiency while maintaining the security of the cryptosystem. A bit is the fundamental unit of an image, and modifying one or more bits can alter an original image (Lai et al., 2023). However, in an original image, the bits within the same bit-plane carry the same significance, while the weight of bits in different bit-planes is proportionate. This can result in lower plaintext sensitivity when using key scheming techniques. To address this, some methods, such as summation over all pixels, have been proposed to improve plaintext sensitivity. These techniques can help to enhance the security of the cryptosystem by increasing its responsiveness to changes in the plain image.

We have introduced a new key scheming technique called weighted addition (KS-WA) to address the issues mentioned earlier. This approach aims to improve plaintext sensitivity in the encryption algorithm by assigning weights to different bits in the original image. The corresponding equation for KS-WA is as follows:

$$\mathbf{R_1} \times I_m \times \mathbf{R_2} = [r_1 \quad r_2 \quad \cdots \quad r_M] \times \begin{bmatrix} I_{m1} & I_{mM+1} & \cdots & I_{m(N-1)M+1} \\ I_{m2} & I_{mM+2} & \cdots & I_{m(N-1)M+2} \\ \vdots & \vdots & \ddots & \vdots \\ I_{mM} & I_{m2}M & \cdots & I_{mMN} \end{bmatrix} = S_{\mathbf{R}I_m} \tag{1}$$

where:

$S_{\mathbf{R}I_m}$: Weighted Addition.

$\mathbf{R_1}$, Random vector of size $1 \times M$,

$\mathbf{R_2}$ Random vector of size $N \times 1$.

These vectors are utilized in the weighted comprehensiveness process, with I representing the original image of size $M \times N$. Eq. 1 shows that each bit in the original image has a unique and pseudo-random weight, which means that the weights of the bits are not proportional to each other. Using this approach eliminates the necessity for numerous iterations of encryption operations or a single round of intricate encryption operations. This, in turn, leads to quicker and more effective encryption. The encryption process heavily relies on the weighted addition $S_{\mathbf{R}I_m}$, which plays a critical role in ensuring the security of the encrypted data. Any minor alteration in the input image can lead to a significantly different output, making it difficult to decrypt the data without the correct key. In this paper, four different chaotic maps such as Arnolds' chaotic map, Sine map, Tent map and the Logistic map are used due to following reasons:

- **Increased randomness:** Incorporating multiple chaotic maps into an encryption algorithm can enhance the level of randomness and unpredictability. By doing this, the sequence of numbers used to encrypt an image becomes more difficult for potential attackers to predict, improving the security of the encryption.
- **Higher resistance to attacks:** The use of multiple chaotic maps in an encryption algorithm can enhance its security against various attacks such as differential cryptanalysis, linear cryptanalysis, and brute-force attacks. These attacks are based on identifying statistical patterns in encrypted data, and the incorporation of multiple chaotic maps can increase the difficulty of finding such patterns.
- **Improved key space:** Incorporating multiple chaotic maps in an encryption technique can expand the size of the key space, which represents the set of all feasible keys that can be employed to secure an image. A larger key space makes it challenging for attackers to determine the appropriate key and decrypt the image.

The following subsections provide a brief explanation of the chaotic map utilized in the proposed study.

### 2.2. Arnods' chaotic map

Arnold's transformation is a mathematical formula as given in Eq. 2 that can be applied to an image to change the arrangement of its pixels in a seemingly random way (Masood et al., 2022). However, if this transformation is applied repeatedly, the original image can be restored. The number of iterations required for the image to return to its original state after undergoing the transformation is referred to as Arnold's period. This period is dependent on various factors, including the image size and specific parameters (i.e.) $c$ and $d$.

$$\begin{aligned} \begin{bmatrix} y_{m+1} \\ z_{m+1} \end{bmatrix} \times \begin{bmatrix} y_{m+1} \\ z_{m+1} \end{bmatrix} &= B \times \begin{bmatrix} y_m \\ z_m \end{bmatrix} \times (modM) \\ &= \begin{bmatrix} 1 & b \\ c & bc+1 \end{bmatrix} \times \begin{bmatrix} y_m \\ z_m \end{bmatrix} \times (modM) \end{aligned} \tag{2}$$

where, M represents the size of the image having an equal number of rows and columns ($M \times M$). It involves using a matrix A, where the determinant of A is equal to 1, and two positive integers $b$ and $c$. The image pixels are located at specific positions, which are denoted by their $(y_m, z_m)$ coordinates, with $y$ and $z$ ranging from 0 to M-1.

The simplest form of Arnold's chaotic map is given in Eq. 4.

$$\begin{aligned} &\begin{bmatrix} (y_{m+1})^2 + (y_{m+1})(z_{m+1}) \\ (z_{m+1})(y_{m+1}) + (z_{m+1})^2 \end{bmatrix} \times \begin{bmatrix} y_m \\ z_m \end{bmatrix}' \\ &= B \times (modM) = \begin{bmatrix} y_m + b \cdot z_m \\ c \cdot y_m + bc \cdot z_m + z_m \end{bmatrix} \times (modM) \end{aligned} \tag{3}$$

$$\begin{bmatrix} y_m(y_m+3+z_m)+z_m+2 \\ z_m(z_m+3+y_m)+y_m+2 \end{bmatrix} \times \begin{bmatrix} y_m \\ z_m \end{bmatrix}' = B \times (modM) = \begin{bmatrix} y_m + b \cdot z_m \\ c \cdot y_m + bc \cdot z_m + z_m \end{bmatrix} \times (modM) \quad (4)$$

The transformation of the sample positions is achieved using a mathematical process called a "cat map". The new positions of the samples after the transformation are denoted as $(y_{m+1}, z_{m+1})$, which represents the $(y_m, z_m)$ coordinates of the transformed samples.

Arnold's cat map has two fundamental operations: tension and fold. Tension is achieved by multiplying a matrix (A) with the coordinates $(y_m, z_m)$, which causes the points to be stretched and enlarged. Fold, on the other hand, involves taking the coordinates $(y_m, z_m)$ modulo a unit matrix, which folds the points back into a limited range of values. These two operations work together to produce the complex and unpredictable behavior that characterizes chaotic systems.

### 2.3. Sine map

The chaotic sine map is a type of one-dimensional chaotic map that possesses a basic structure but is capable of generating intricate and random sequences within a predetermined range of values that usually lie between 0 and 1 (Hosny et al., 2023). Its definition involves applying a sine function to a variable, which is then multiplied by a constant factor to obtain the next value in the sequence. Although the sine map is relatively simple in structure, it can demonstrate a diverse array of chaotic behavior, making it a valuable tool for analyzing chaotic systems and their properties. Mathematically, the sine map may be represented as follows:

$$g_{m+1} = q \times sine(\pi g_m) \quad (5)$$

In Eq. 5, $q_0$ is the initial condition for $g_m$. According to the bifurcation diagram shown in Fig. 3(a), when $q$ lies between 2.1 to 4 (i.e. $q \in [3.1, 4]$), the system shows more random behavior.

### 2.4. Tent map

The Tent map is a type of chaotic map that operates on one dimension and is utilized in various applications (Akraam et al., 2022). The Tent map is defined as follows:

$$h_{m+1} = \begin{cases} 2uh_m h_m < 0.5 \\ 2u(1-h_m)h_m \geqslant 0.5 \end{cases} \quad (6)$$

where $u$ is a control parameter and $u_0$ is the seed value for $h_m$. Its bifurcation diagram is typically depicted as a curve resembling that of a Tent function as shown in Fig. 3(b) where it can also be seen that the system enters into the chaotic states when the value of $u$ lies between 0.5 to 1 (i.e $u \in [0.5\ 1]$).

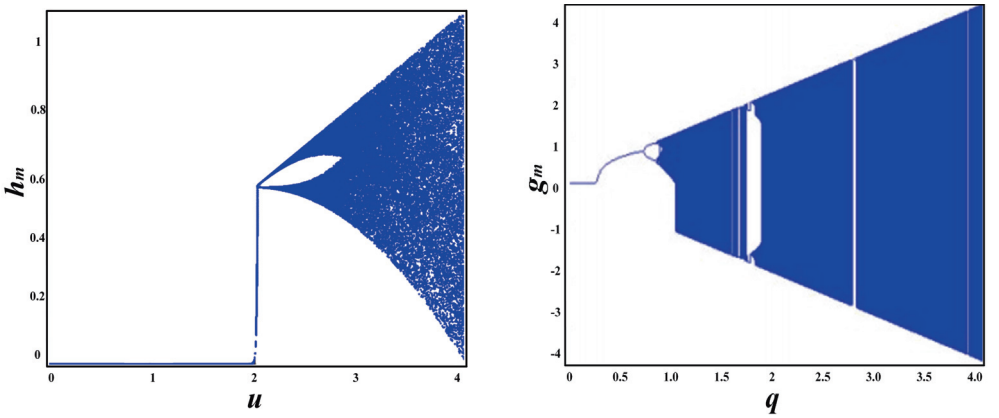
### 2.5. Substitution

The substitution process can alter the values of pixels in an image, and it is not very computationally intensive. However, the distribution of these altered values is not even across the image (Khan et al., 2023). This uneven distribution makes it easier for attackers to analyze the image statistically, so traditional substitution methods are not effective at securing the image effectively. To address this problem, a new substitution technique called "ST" is proposed. The proposed ST involves three main steps to encrypt an image.

1. In the first step, a chaotic system is used to randomly group the pixels of the original image. This grouping process ensures that the correlation between adjacent pixels is eliminated and the order of pixel encryption becomes unknown to any potential attackers. After the encryption process is finished, the pixels are restored to their initial positions.
2. To optimize storage and transmission space, a chaotic S-box is created in the second step. This type of S-box is chosen due to the inherent properties that make it a suitable choice for this purpose, such as its chaotic nature.
3. In the final step, the constructed S-boxes are randomly allocated to each group, and the substitution process is executed. This allocation helps in reducing the number of S-boxes needed for the encryption process. In order to attain uniform distribution of the ciphertext image, Eq. 7 presents a novel method of random substitution.

$$E_i = \begin{cases} S_1(O_i \oplus S_2(\theta))i = 1 \\ S_1(O_i) \oplus S_2(E(i-1))i \neq 1 \end{cases} \quad (7)$$

where two sets of data, are represented as $O_i$ and $E_i$. The encryption process involves using two allocated S-boxes, $S_1$ and $S_2$. Additionally, a pseudo-random integer $\theta$ is used to make $S_2(\theta)$ pseudo-random. In simple terms, S-boxes are used to scramble the original image and the $\theta$ integer ensures that the scrambling is unpredictable and robust against cyberattacks. In this random *ST*, each original image is first converted into a random value using a process




(a) Bifurcation diagram of chaotic sine map (b) Bifurcation diagram of chaotic tent map

**Fig. 3.** Bifurcation diagrams of different chaotic maps used in the proposed work.

called XOR operation. This operation involves using a pseudo-random value, known as $S_2(\theta)$ or $S_2(E(i-1))$, to create a new value. After this conversion, the resulting value is used in the substitution process, where the output $E_i$ of the S-box $S_1$ is also a random value. By using this method, the ciphertext image is generated with a uniform distribution, which ensures that the encryption performance is satisfactory.

## 3. Proposed KSP-DWT-IET

A new image encryption technique (IET) called KSP-IET is proposed in this research, which is based on the key scheduling process. The KEP-IET uses three main methods, namely KS-WA, DWT and ST, and only requires one encryption round. The ST method consists of three steps: (a) The method of random grouping (MRG), (b) the development of S-box (DSB) and (c) the method of random substitution (MRS). These steps are designed to encrypt different types of images effectively.

The major steps involved in the proposed encryption process of KSP-IET are illustrated in Fig. 4. The algorithm presented intends to offer a safe and effective image encryption resolution utilizing the KSP method and various encryption methods.

### 3.1. KS-WA stage

Based on Section 2.1, a key scheme called KS-WA is developed to update initial keys using original images. This scheme is designed to provide strong sensitivity to even slight changes in the original image.

The process for updating keys using KS-WA on an $R(row) \times C(column)$ grayscale image $I_m$ involves the following steps:

- **Step 1:** The chaotic systems are iterated with the weighting key ($y_m, z_m, g_m, q, h_m$ and $u$) using Eqs. (4)–(6). The first 500 values are discarded, and two pseudo-random weighting vectors are obtained: $\mathbf{R_1}$ of size $1 \times M$ and $\mathbf{R_2}$ of size $N \times 1$.
- **Step 2:** The weighted summation $S_{RI_m}$ is calculated using $\mathbf{R_1}, \mathbf{R_2}$, and the $(I_m)$, using Eq. 1.
- **Step 3:** The fractional part of $S_{\mathbf{R}I_m}$ is used to update the initial key $y_m, z_m, g_m, q, h_m$ and $u$ using Eq. 8 to obtain the updated key z.

$$\text{Update initial keys} = \left\{ \begin{array}{l} y = mod(y_0 + (S_{WI} - floor(S_{WI})), 1) \\ z = mod(z_0 + (S_{WI} - floor(S_{WI})), 1) \\ g = mod(g_0 + (S_{WI} - floor(S_{WI})), 1) \\ q = mod(q_0 + (S_{WI} - floor(S_{WI})), 1) \\ h = mod(h_0 + (S_{WI} - floor(S_{WI})), 1) \end{array} \right\} \tag{8}$$

### 3.2. Discrete wavelet transform

The wavelet transform (WT) is a mathematical tool developed in the 1980s that can efficiently analyze non-stationary and fast transient signals with wide frequency bands (Holschneider, 1988). Wavelets are signals that are localized in both time and scale and have irregular shapes. By decomposing a signal into many shifted and scaled representations of a mother wavelet, the WT can separate a signal into component wavelets. These wavelets can be further decimated to remove some of the details, allowing the fine details that contain high frequencies, such as the LH-sub-band, HL-sub-band and HH-sub-band) in a signal to be isolated using small wavelets and the coarse details to be identified using large wavelets, such as (LL-sub-band). The low-frequency sub-band of an image holds the major part of the plaintext information, while the high-frequency sub-bands demonstrate the more intricate details such as the edges of the original image. These distinctions between the sub-bands can be seen in Fig. 5.

The proposed encryption technique employs the Haar wavelet. This wavelet was first proposed by Alfrd Haar in 1910 (Haar, 1909). The Haar wavelet transform can be represented by the matrix equation $G' = WGW^T$, where is an image of size $A \times A, W$ is the Haar transform matrix of size $A \times A$ and $G'$ is the resulting transformed matrix of size $A \times A$ that contains the Haar basis function $g_m(z)$. This function is defined on the interval $z \in [0, 1]$ where $m$ ranges from 0 to M-1. The decomposition of this function is as follows:

$$m = 2^q + k \tag{9}$$

where $q$ represents the greatest power of 2 present in the integer $m$, and $k$ represents the remainder i.e. $k = 2^q - m$. Eq. 10 defines the Haar basis function.

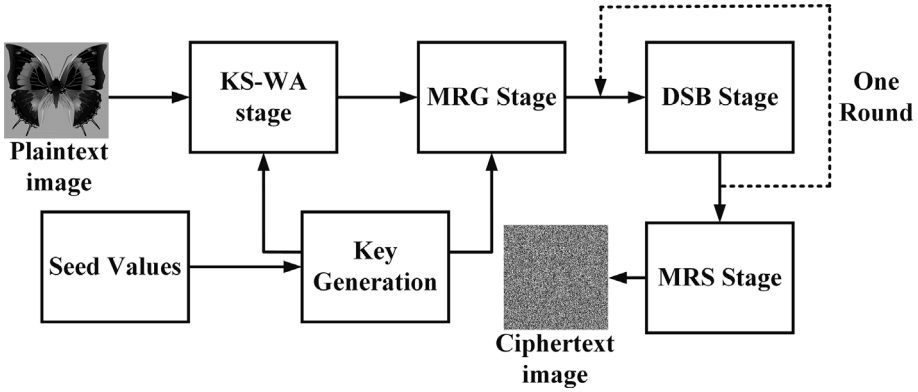



**Fig. 4.** Major stages involved in the proposed encryption process.

(a) Cameraman image

(b) LL (c) LH (d) HL (e) HH

**Fig. 5.** Plaintext image and its corresponding frequency sub-bands.

$$g_m(z) = \frac{1}{\sqrt{N}} \begin{cases} 1 & if\ m = 0 \& 0 \leqslant z \leqslant 1 \\ 2^{q/2} & if\ a > 0 \& k/2^q \leqslant z < \frac{k+0.5}{2^k} \\ -2^{q/2} & if\ m > 0 \& (k+0.5)/2^q \leqslant z < \frac{k+1}{2^q} \\ 0 & \text{Elsewhere} \end{cases} \tag{10}$$

The transformation matrix for the two-dimensional discrete Haar wavelet transform (DHWT) can be found by replacing the inverse transformation kernel, which is defined by Eq. 11.

$$g'(z,m) = \frac{1}{\sqrt{N}} g_m(z/N)\ for\ z = 0, 1, 2, \ldots, N-1 \tag{11}$$

where, $g(m,z)$ will be:

$$g(m,z) = G' = \begin{bmatrix} g_0(\frac{0}{N}) & h_0(\frac{1}{N}) & \cdots & g_0(\frac{N-1}{N}) \\ g_1(\frac{0}{N}) & h_1(\frac{1}{N}) & \cdots & g_1(\frac{N-1}{N}) \\ g_2(\frac{0}{N}) & h_2(\frac{1}{N}) & \cdots & g_2(\frac{N-1}{N}) \\ \vdots & \vdots & \ddots & \vdots \\ g_{N-1}(\frac{0}{N}) & h_{N-1}(\frac{1}{N}) & \cdots & g_{N-1}(\frac{N-1}{N}) \end{bmatrix} \tag{12}$$

Therefore, the resulting transform matrix ($G'$) will be:

$$G = \frac{1}{\sqrt{N}} G' \tag{13}$$

When processing 2-D digital images, two filters are applied to each row of the image: a low-pass filter and a high-pass filter. The resulting outputs from these filters are then downsampled by a factor of two, resulting in the creation of two different kinds of information sub-bands: $L_f$ (the approximate information sub-band) and $H_f$ (fine detail information sub-band) in the horizontal direction. This process is then repeated for each column of these two new images to obtain four sub-bands: $LL1$ (the approximation image), $LH$ (the vertical detail image), $HL$ (the horizontal detail image) and $HH$ (the diagonal detail image). These four sub-bands contain all the information in the original image.

If we apply the 2D DWT again to the $LL_1$ sub-band, we get four new sub-bands: $LL_2, LH_2, HL_2$, and $HH_2$. This process can be repeated $T$ times to yield a sequence of sub-images: $LL_T, LH_K, HL_T$, and $HH_T$. Increasing the value of $K$ results in a decrease in the size of each sub-band by a factor of $2^n$, where $n$ is an integer ranging from 1 to $N-1$ (i.e. $n \in [1N-1]$). For the propose work, the value of $K$ is set to 5. This means that when $K = 5$, the size of each sub-band will be reduced to $16 \times 16$ for an image of size $256 \times 256$. This reduction in size is specifically implemented for substitution purposes because processing large-sized images, such as $256 \times 256$, can take a lot of time. Hence, in order to minimize the computational complexity, the substitution process is only applied to sub-bands that are of size $16 \times 16$.

### 3.3. Substitution technique

This section provides a detailed explanation of the three main sub-stages of the proposed ST, namely MRG, DSB, and MRS.

#### 3.3.1. Method of random grouping

In block ciphers, cryptographic algorithms are used to encrypt data in fixed-size blocks, typically used for image encryption (Amdouni et al., 2022). However, when encrypting image blocks, the high correlation between adjacent pixels and the known grouping of pixels can lead to reduced security. To address this issue, a new MRG is proposed.

The MRG method employs chaotic sequences to randomly group pixels, making it difficult for third parties to determine the pixel grouping. The proposed Modified Random Grouping (MRG) algorithm involves several steps to enhance the encryption process. Chaotic sequences are utilized to group pixels randomly. Afterward, the present set of pixels is subjected to encryption by utilizing the pre-ciphertext. Subsequently, the encrypted pixels are repositioned to their original locations, thus producing the ultimate ciphertext image. The use of MRG ensures that both the pixel grouping and the encryption order remain concealed from third parties, leading to improved security measures. Below are the details of the steps carried out to propose MRG.

- **Step 1:** Use KS-WA to update the initial keys ($y_m, z_m, g_m, q, h_m$ and $u$) and obtain the updated key ($y_m, z'_m, g'_m, q', h'_m$ and $u'$).

- **Step 2:** Use the updated key to generate a pseudo-random sequence $A, B, C, D$ and $E$. Each random sequence of length $R$ is generated by iterating the chaotic systems and discarding the first 500 values. Next, thirty values are discarded in sequence to create a pseudorandom series of length N, represented by $A', B', C', D'$, and $E'$. In addition, three other pseudorandom values are generated: $I_r, I_c$, and $\theta$.
- **Step 3:** To derive the row index sequence $R_0$ and the column index sequence $C_0$, arrange the $A, B, C, D, E$ and $A', B', C', D', E'$ sequences in ascending order.
- **Step 4:** Calculate the initial point position $(I_{0r}, I_{0c})$ and the direction $\theta$ using an Eq. 14.

$$\left.\begin{array}{l} I'_r = mod(floor(I_r \times 10^{14}), M+1) \\ I'_c = mod(floor(I_c \times 10^{14}), N+1) \\ \theta' = mod(floor(\theta \times 10^{14}), 8) \end{array}\right\} \tag{14}$$

- **Step 5:** Read the pixel positions from the image using the row and column index sequences $R_0$ and $C_0$, the start point position $(I_{0r}, I_{0c})$, and the direction $\theta_0$.
- **Step 6:** Divide all the pixels into groups based on the pixel index sequence obtained in Step 5. The groups consist of $T = min(R, C)$ groups, with each group containing $Q = max(R, C)$ pixels. The grouped plaintext of size $T \times Q$ is then created.

The process of MRG is shown in Fig. 6. The figure displays eight different directions such as right-bottom (right), right-bottom (left), (left-bottom (right), left-bottom(left), right-upper (right), right-upper (left), (left-upper (right) and left-upper (left) which result in different random groups. According to the procedure shown in Fig. 6, different groups and the order of pixel encryption are created. Once the grouping is created, the values are substituted according to the positions shown in Fig. 6. The updated values are shown in Fig. 7.

### 3.3.2. Development of the proposed S-box

An S-box is a mathematical function that receives a specific number of input bits and returns the same number of output bits (Abuelyman and Alsehibani, 2008). The function is non-linear and is denoted by S: 0, 1m → {0, 1}m, where 0, 1n is the space of n elements from the Galois field of two elements (GF(2)) (Khan et al., 2023). The S-box is a crucial component in block ciphers as it provides nonlinear confusion to create a high level of randomness in the ciphertext image.

An S-box is considered chaotic if it exhibits sensitivity to the initial key due to its inherent properties. To achieve this, a simple and efficient method for constructing an S-box based on chaos has been designed and is referred to as DSB.

The proposed DSB Method involves creating 16 different substitution boxes (S-boxes) that are used in combination to encrypt data. To improve the security of the encryption algorithm, a different set of S-boxes is selected randomly for every group. This randomization also reduces the number of S-boxes that need to be created. The number of S-boxes utilized can be modified depending on the situation. To elaborate on the SCM process, the following section provides a detailed explanation.

**Step 1:** The first step involves updating the initial key $(y_m, z_m, g_m, q, h_m$ and $u)$ using KS-WA to get the updated key $(y'_m, z'_m, g'_m, q', h'_m$ and $u')$.

**Step 2:** sixteen pseudo-random values are generated using $y'_m, z'_m, g'_m, q', h'_m$ and $u'$ as the initial key and iterating the chaotic systems, but the first fifteen values are discarded. These values are used as initial keys for constructing S-boxes.

**Step 3:** To generate a sequence of 256 chaotic values, the researchers use one of the keys they obtained in step 2 to iterate a chaotic system and discard the first 500 values produced by the system.

**Step 4:** After obtaining the chaotic sequence of 256 values, sort them in ascending order to obtain the sorted index sequence. This sorted sequence is then used as the initial 1D S-box.

**Step 5:** To generate two pseudo-random values, $V$ and $W$, the first step is to discard 30 values. The resulting values are then converted into two integers, $V'$ and $W'$, which are restricted to the range of 0 to 15. In other words, $V'$ and $W'$ belong to the interval [0, 15]. These integers are used as parameters for Arnold's cat map to create further S-boxes as shown in Eq. 15.

$$\left.\begin{array}{l} V' = mod(floor(V \times 10^{14}), 16) \\ W' = mod(floor(W \times 10^{14}), 16) \end{array}\right\} \tag{15}$$

**Step 6:** This step of the proposed DSB algorithm involves dividing each position of the initial 1D S-box $(S'_1)$ into two 4-bit index values, referred to as $(a_i, b_i)$. These values are then used to calculate new index values $(a'_i, b'_i)$ using Arnold's cat map formula (Eq. 16). Whereas, Eq. 17 is derived from Eq. 16. The elements of the initial S-box are then rearranged into their new positions based on the new index values obtained. The resulting rearranged S-box is referred to as the 2D S-box $(S^{2D})$.

$$\begin{bmatrix} a'_i \\ b'_i \end{bmatrix} = \begin{bmatrix} 1 & V' \\ W' & V'W'+1 \end{bmatrix} \begin{bmatrix} a'i \\ b'i \end{bmatrix} \times (mod 16) \tag{16}$$

$$\begin{bmatrix} a'_i \\ b'_i \end{bmatrix} = \begin{bmatrix} a'_i + V'b'_i \\ W'a'_i + V'W'b'_i + b'_i \end{bmatrix} \times (mod 16) \tag{17}$$

**Step 7:**To obtain 16 S-boxes, perform Steps 3 through Step 6 repeatedly.

### 3.3.3. Method of random substitution

Our proposed MRS method applies a randomized input value to the S-box before substitution, resulting in a random output value. To enhance the security of the ciphertext image, randomization is used to ensure that the distribution of pixels is uniform. This helps to make the image more resistant to statistical analysis attacks. To further increase the diversity of the encryption process, two S-boxes are randomly assigned to each group. Moreover, the proposed RSM technique reduces the number of S-boxes required for encryption. The encryption procedure in RSM includes encrypting the current pixel group using the preceding ciphertext group. The steps of our MRS are summarized below.

**Step 1:** The seed values $(y_m, z_m, g_m, q, h_m$ and $u)$ is updated using KS-WA to get new seed values $(y'_m, z'_m, g'_m, q', h'_m$ and $u')$.

**Step 2:** The chaotic systems are then iterated with the updated keys to generate pseudo-random values. To prevent transient effects, discard the initial 500 values, followed by ten successive discards.

**Step 3:** From the generated pseudo-random values, three values are taken at a time $(q_i, 1, q_i, 2, q_i, 3)$ to calculate two indices of the S-box, namely $\{\psi_i, 1, \psi_i, 2\}$, and a random integer $\phi_i$ using Eqs. (18)–(20). Here, $EI_m(i-1, C)$ is the last ciphertext of the previous group. This process is repeated for each of the $T$ groups.

$$\psi_{i,1} = \left\{\begin{array}{l} mod(floor(q_{i,1} \times 10^{14}), 16) + 1, i = 1 \\ mod(mod(floor(q_{i,1} \times 10^{14}), 256) \\ \oplus EI_m(i-1, C), 16) + 1 i \neq 1 \end{array}\right\} \tag{18}$$

$$\psi_{i,2} = \left\{\begin{array}{l} mod(floor(q_{i,2} \times 10^{14}), 16) + 1, i = 1 \\ mod(mod(floor(q_{i,2} \times 10^{14}), 256) \\ \oplus EI_m(i-1, C), 16) + 1 i \neq 1 \end{array}\right\} \tag{19}$$

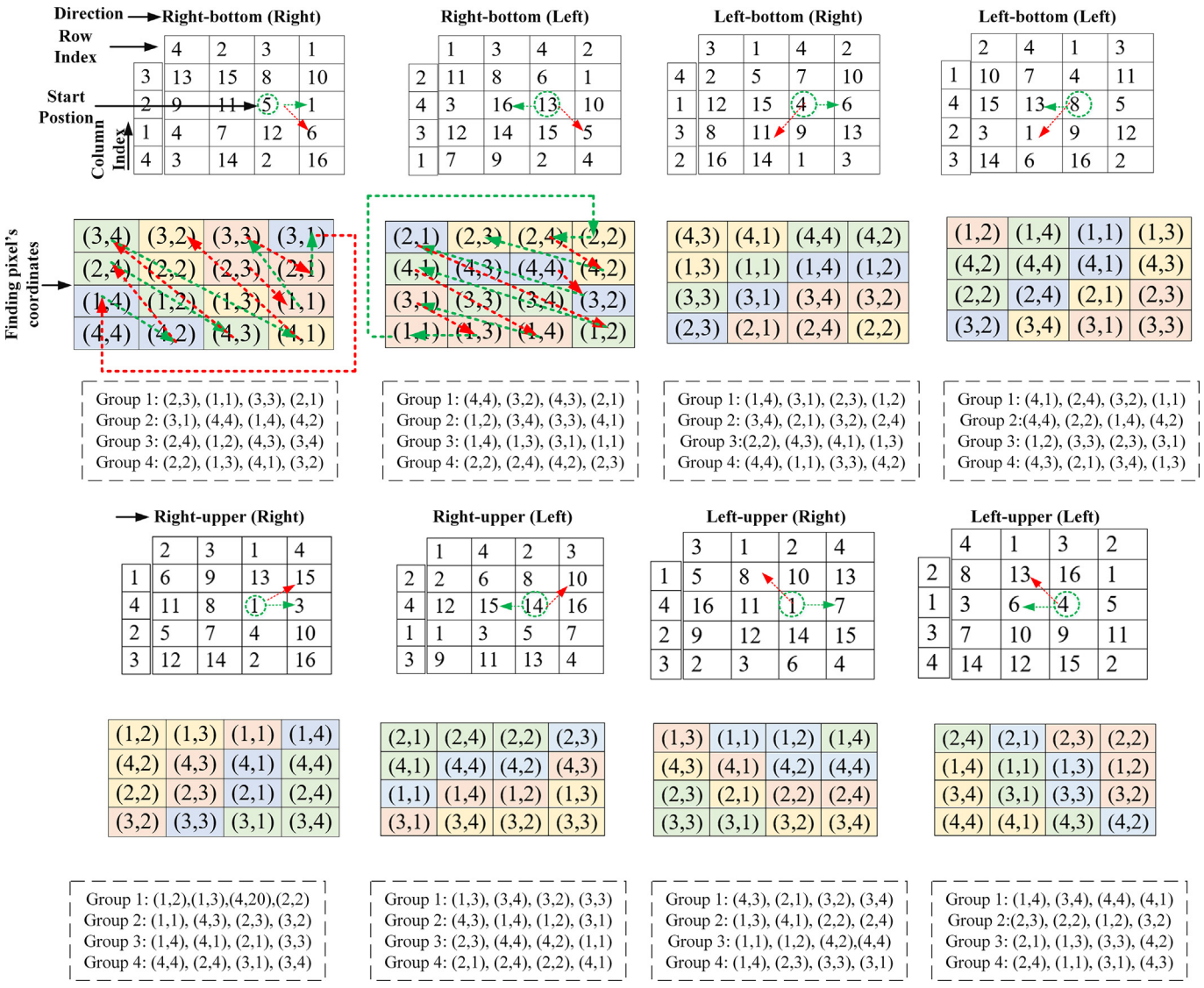



**Fig. 6.** Proposed MRG methodology for the encryption process.

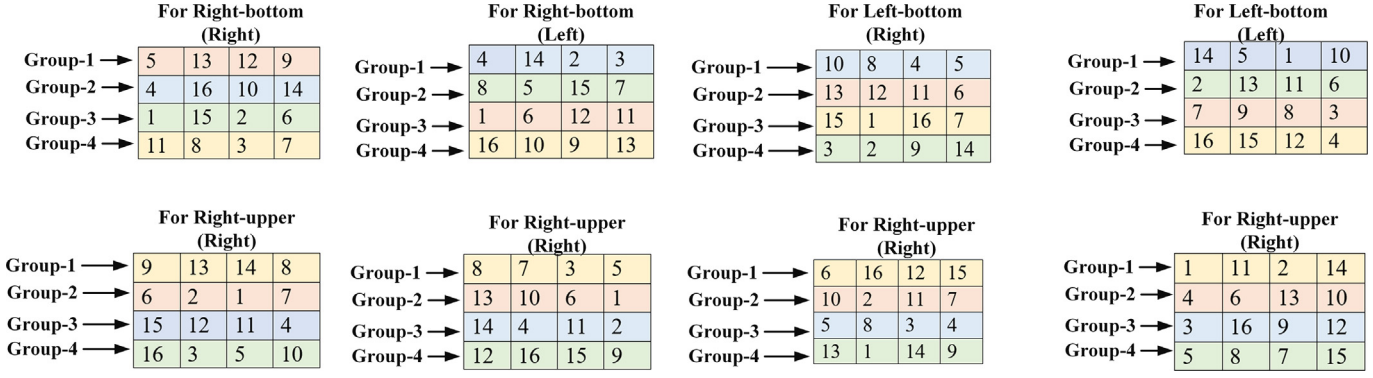



**Fig. 7.** Values updated according to respective positions.

$$\phi_i = \begin{cases} mod(floor(q_{i,3} \times 10^{14}), 16) + 1, i = 1 \\ mod(mod(floor(q_{i,3} \times 10^{14}), 256) \\ \oplus EI_m(i-1, C), 16) + 1 i \neq 1 \end{cases} \quad (20)$$

**Step 4:** Finally, using Eq. 21, substitute the plaintext of the $i^{th}$ group to obtain the ciphertext image. This substitution involves using a function called $g(x)$ which takes a value $x$ and converts it into two 4-bit index values. These index values are then used as

inputs for the S-box, which is a lookup table that maps each possible input to a unique output. The output from the S-box is the substituted value that replaces the original plaintext value, resulting in the ciphertext image.

$$EI_m(i,j) = \left\{ \begin{array}{l} I_1^{\psi_{i,1}}(g(O(i,j) \oplus I_2^{\psi_{i,2}}(g(\theta_i)))) i = 1 \\ I_1^{\psi_{i,1}}(g(O(i,j) \oplus I_2^{\psi_{i,2}}(g(EI_m(i,j-1))))) i \neq 1 \end{array} \right\} \quad (21)$$

where, $O(i,j)$ is an original image.

**Step 5:**To create the ultimate encrypted image, the encryption process is repeated through the execution of both Step 3 and Step 4. Then, the encrypted pixels are re-positioned to their original locations. Fig. 8 depicts a detailed block diagram of the proposed approach, while Algorithm 1 presents a pseudo-code description.

**Algorithm 1.** Pseudo code for the proposed work

**Start**
**Plaintext Image**
→Define chaotic map:
→**Arnold's cat map**

$$\begin{bmatrix} y_m(y_m+3+z_m)+z_m+2 \\ z_m(z_m+3+y_m)+y_m+2 \end{bmatrix} \times \begin{bmatrix} y_m \\ z_m \end{bmatrix}' = B \times (modM) = \begin{bmatrix} y_m + b \cdot z_m \\ c \cdot y_m + bc \cdot z_m + z_m \end{bmatrix} \times (modM)$$

→ **Chaotic sine map** $g_{m+1} = q \times sine(\pi g_m)$
→ **Chaotic tent map** $h_{m+1} = 2uh_m h_m < 0.5$
$h_{m+1} = 2u(1-h_m)h_m \geqslant 0.5$
**for** N = 1:1000
$y_{m+1}$ = Arnold's cat map
$z_{m+1}$ = Arnold's cat map
$g_{m+1}$ = Chaotic Sine map
$h_{m+1}$ = Chaotic Tent map
**end**
→ Update initial keys: $y = mod(y_0 + (S_{WI} - floor(S_{WI})), 1)$
$z = mod(z_0 + (S_{WI} - floor(S_{WI})), 1)$
$g = mod(g_0 + (S_{WI} - floor(S_{WI})), 1)$
$q = mod(q_0 + (S_{WI} - floor(S_{WI})), 1)$
$h = mod(h_0 + (S_{WI} - floor(S_{WI})), 1)$
**for** N = 1:1000
→ Generate sequences A,B,C,D,E,F
**end**
→ Find initial positions and direction:
$I'_r = mod(floor(I_r \times 10^{14}), M+1)$
$I'_c = mod(floor(I_c \times 10^{14}), N+1)$
$\theta' = mod(floor(\theta \times 10^{14}), 8)$
→ Generate 256 values for the proposed S-box
→Generate pseudo random values:
**for** N = 1:1000
**for** N = 1:1000
$V' = mod(floor(V \times 10^{14}), 16)$
$W' = mod(floor(W \times 10^{14}), 16)$
**end**
**end**
→ Calculate two indices and an integer: $\psi_{i,1}, \psi_{i,2}$ and $\phi_i$
→ Apply DWT: **When K = 5**
→ Perform substitution:
$I_1^{\psi_{i,1}}(g(O(i,j) \oplus I_2^{\psi_{i,2}}(g(\theta_i)))) i = 1$
$I_1^{\psi_{i,1}}(g(O(i,j) \oplus I_2^{\psi_{i,2}}(g(EI_m(i,j-1))))) i \neq 1$
**Output:** Ciphertext image
**End**

Despite incorporating various approaches in our research work, we have employed unique and innovative methods by utilizing the combined techniques in different ways. For instance, the key substitution process is a completely new process proposed in this research work. Additionally, both the Sbox and random sequence generation processes have also been implemented in novel ways.

## 4. Statistical results and analysis

To assess the strength of the encryption method presented, various security attacks such as statistical, brute-force, and differential attacks are conducted. Bedoui et al. (2022). Traditional images such as baboon, cameraman and Lenna, etc., are used as test images, each with a size of 256 × 256. The experiments are conducted on a Windows 11 operating system using MATLAB R2022a with 8 GB of RAM and a Core (TM) i5-3210 M-CPU at 2.50 GHz.

To evaluate the effectiveness of the proposed encryption scheme, a comparative analysis between our proposed method and the existing ones is also conducted, as detailed in the literature review section. The reason for comparing a proposed encryption scheme with specific existing work is to evaluate its effectiveness, strengths, and weaknesses in terms of security as well as processing time in comparison to other encryption schemes. The specific papers chosen for comparison are representative of the state-of-the-art in the field of encryption.

The existing research works are selected based on the techniques used in them. Because the proposed work heavily relies on chaos, it is preferable to select encryption schemes based on chaotic maps in order to make a fair comparison between the proposed work and existing works. Additionally, the selected existing research works provide a comprehensive understanding of the existing methods and their limitations.

The encrypted image generated by this method has a noisy appearance, and no plain text information is visible to the naked eye, as demonstrated in Fig. 9. Furthermore, Fig. 10 presents the histogram of both the plaintext and ciphertext images.

### 4.1. Statistical attacks

In order to determine the resilience of the proposed encryption scheme against statistical attacks, several statistical analyses are conducted including histogram, correlation, entropy, lossless, contrast, homogeneity, and energy analysis. Moreover, a color Lena image is also taken into account to gauge the performance of the proposed encryption algorithm. Each individual component, including Red ($R$), Green $G$, and Blue $B$, is independently encrypted using the proposed algorithm.

#### 4.1.1. Histogram analysis

A histogram is a visual depiction of the distribution of pixel intensity values in an image (Agrawal et al., 2022). Histogram analysis, on the other hand, is a method used to evaluate the efficacy of a cryptographic algorithm by analyzing the pixel distribution in an image. This assessment involves comparing the histograms of both the plain image and cipher image components.

Fig. 10 displays the histograms of the plain images and ciphertext images. The analysis shows that the pixel intensity distribution of the ciphertext images is uniformly distributed, which is a desirable attribute of a robust encryption algorithm. This indicates that the proposed encryption algorithm has the capability to con-

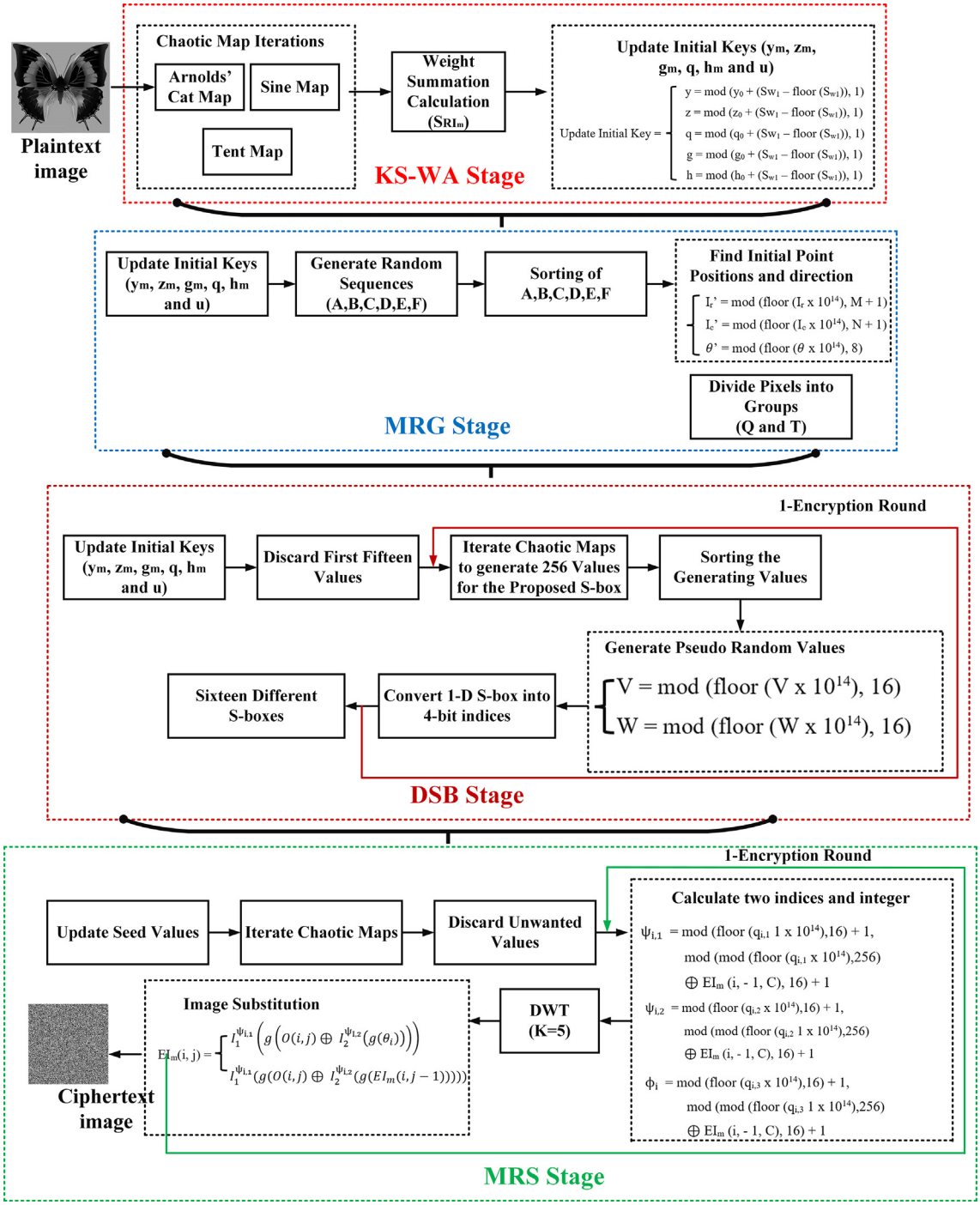



**Fig. 8.** Detailed block of the proposed image encryption process.

ceal the statistical information of the plaintext image by uniformly distributing pixels in the cipher text image.

*4.1.2. Correlation analysis*

To evaluate the efficiency of the novel encryption technique, 3000 pair of adjacent pixels are chosen from both the initial and encrypted images, taking into account the horizontal, vertical, and diagonal orientations (Zhu et al., 2022). Typically, in a plaintext image, the pixel values are highly correlated with their adjacent pixels, horizontally, vertically, and diagonally. Correlation between the image pixels can be calculated using Eq. 22.

$$\begin{cases} \Psi_{a,b} = \dfrac{\frac{1}{M}\sum_{i=1}^{M}(a_i - E(a))(b_i - E(b))}{\sqrt{\frac{1}{M}\sqrt{\sum_{i=1}^{M}(a_i - E(a))^2}\sqrt{\sum_{i=1}^{M}(b_i - E(b))^2}}} \\ E(a) = \frac{1}{M}\sum_{i=1}^{M}(a_i) \\ E(b) = \frac{1}{M}\sum_{i=1}^{M}(b_i) \end{cases} \quad (22)$$

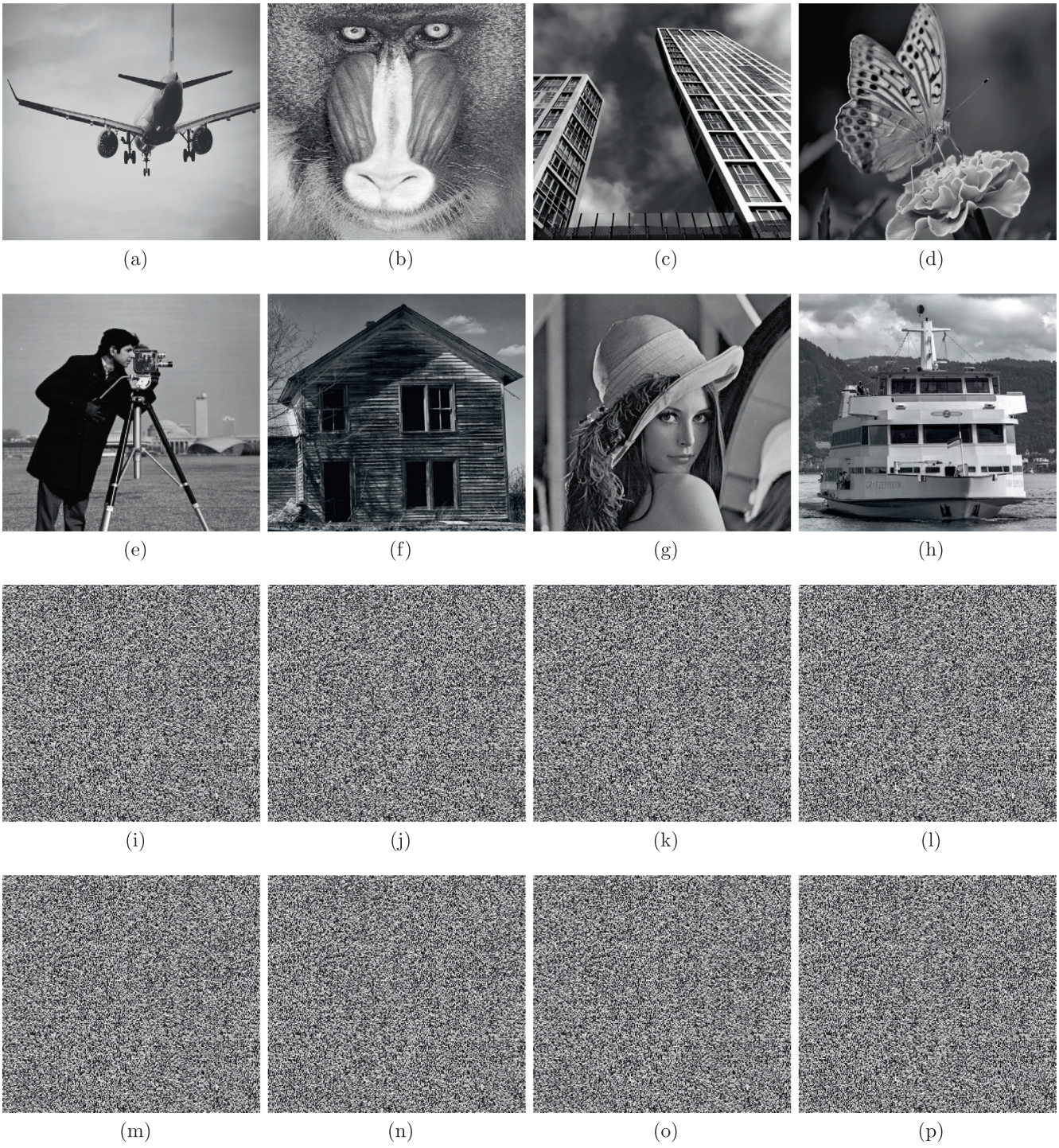



**Fig. 9.** Plaintext and their corresponding enciphered images.

where $M$ represents the number of pixels in an image. The symbols $a$ and $b$ are used to represent two adjacent pixels, while $E(a)$ and $E(b)$ denote their corresponding expectations.

Fig. 11 depicts the correlation measured for the cameraman image, Lenna, Baboon and Aeroplane images and their corresponding ciphertext images. The green and red dots in Fig. 11 show the image pixels. The increased dispersion of dots indicates a lower correlation between the pixels in the image. From the scattered diagrams in Fig. 11, it can be seen that the red dots are completely scattered, which shows the pixels correlation in ciphertext images is very low. whereas the green dots are closer to each other, which shows the pixels correlation in plaintext images is very high. This indicates that the encryption scheme proposed is effective in preventing correlation between adjacent pixels in the encrypted image, thereby enhancing its security.

Table 1 displays the correlation between the values of plaintext and ciphertext images in all three directions. The plaintext image correlation values are nearly 1 in all three directions, which suggests a significant correlation between the values of adjacent pixels.

The distribution of pixels in the ciphertext images appears to be uniform, which results in low correlation between neighboring

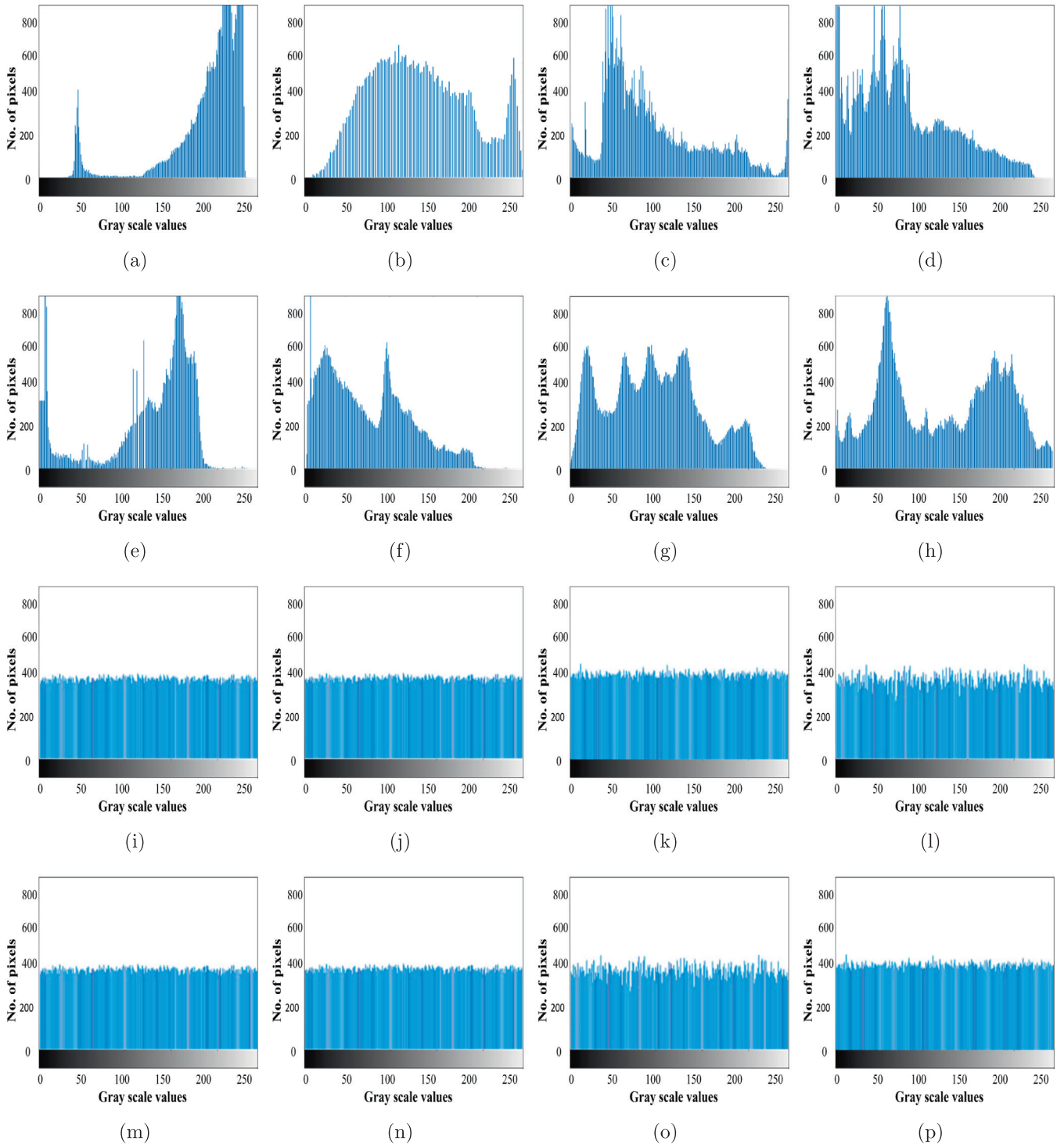
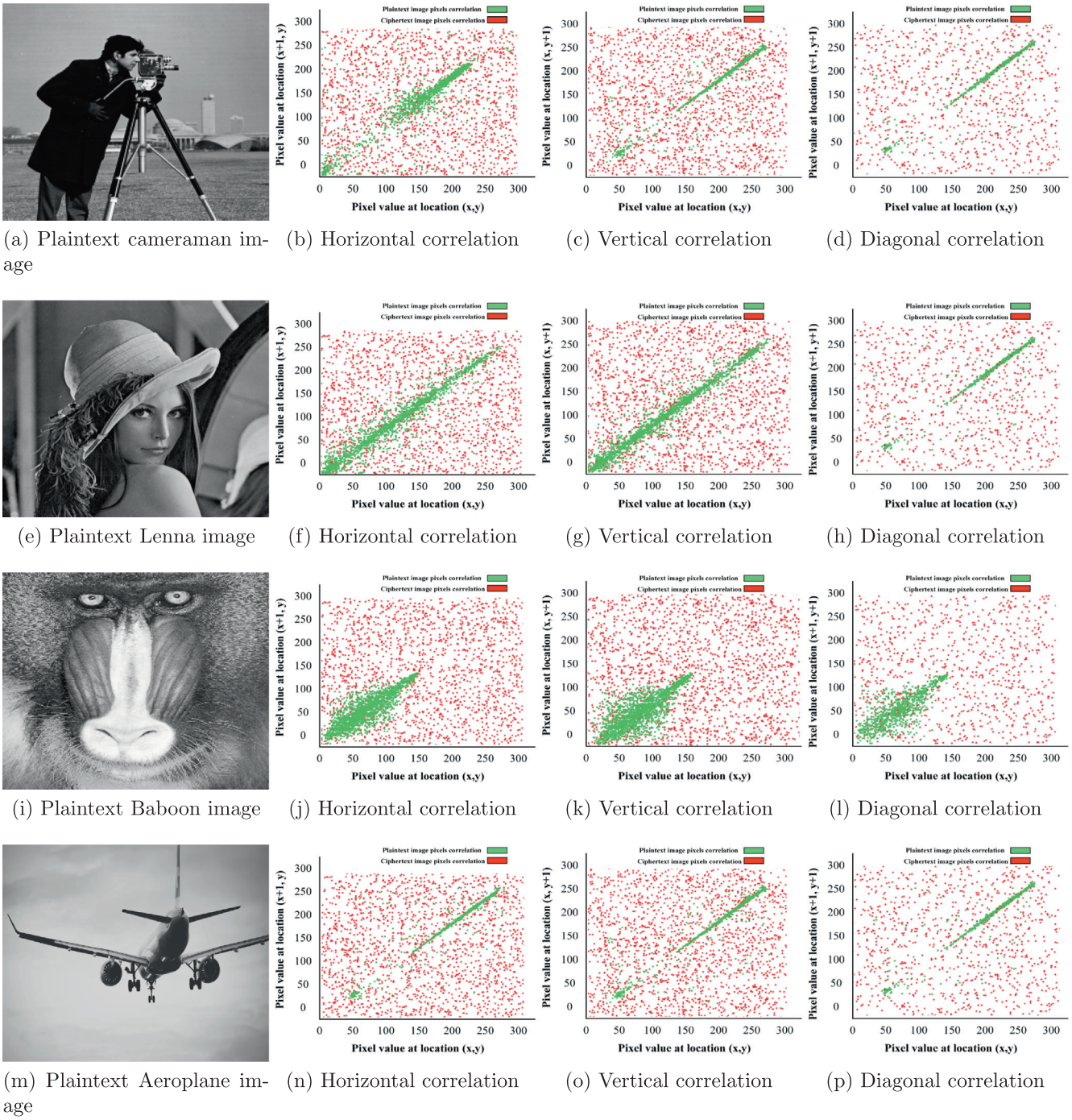



**Fig. 10.** Histograms of the plaintext and their corresponding ciphertext images.

pixels within the encrypted image, as depicted in Table 1. The correction values of the ciphertext image in all directions are all near 0, indicating that the suggested encryption technique disrupts entirely the correlation between adjacent pixels. Consequently, the suggested scheme is highly resilient to statistical attacks. In addition, Table 1 presents a comparison between the proposed encryption scheme and the existing encryption algorithms. The results of this comparison demonstrate that the proposed encryption method is more successful in disrupting the correlation between pixels in an image.

### *4.1.3. Entropy analysis*

Information entropy analysis is a technique used to measure the degree of randomness in an encrypted image and evaluate the distribution of gray values in the image. The level of randomness is expressed by an entropy value, ranging from 0 (lowest random-




(a) Plaintext cameraman image (b) Horizontal correlation (c) Vertical correlation (d) Diagonal correlation

(e) Plaintext Lenna image (f) Horizontal correlation (g) Vertical correlation (h) Diagonal correlation

(i) Plaintext Baboon image (j) Horizontal correlation (k) Vertical correlation (l) Diagonal correlation

(m) Plaintext Aeroplane image (n) Horizontal correlation (o) Vertical correlation (p) Diagonal correlation

**Fig. 11.** Correlation of different plaintext and their corresponding ciphertext images.

ness) to 8 (highest randomness). The analysis results are presented in Table 2, where the proposed scheme is compared to the ideal entropy value of 8. Eq. 23 is used to calculate the entropy value of the scheme, which indicates a significant degree of randomness in the resulting encrypted image.

$$Entropy(I) = \sum_{s=0}^{2^8-1} P(I_s) log_2 \frac{1}{P(I_i)} \tag{23}$$

where, $I_i$ is the information source its probability $P(I_i)$ with 28 states is examined. The information entropy of the cameraman, Lenna, baboon and airplane images are analyzed and compared with existing encryption algorithms. The results presented in Table 2 indicate that the ciphertext image information entropy is approximately 8, indicating that the level of randomness in the proposed scheme is high and approaching the ideal level.

#### 4.1.4. Contrast analysis

Contrast analysis is used to detect objects or patterns within an image. When an image is encrypted, the data is scrambled to make it difficult to interpret. This process typically results in a higher level of randomness within the image, which in turn increases the contrast value. The higher the contrast value, the stronger the encryption is assumed to be. The contrast value can be calculated mathematically using Eq. 24.

**Table 1**
Correlation analysis.

| Plaintext images | Directions | Ref (Hua and Zhou, 2016) | Ref (Zhu et al., 2019) | Ref (Ping et al., 2018) | Ref (Diab, 2018) | Ref (Pak and Huang, 2017) | Ref (Ye and Huang, 2017) | Proposed |
|---|---|---|---|---|---|---|---|---|
| Cameraman | Horizontal | 0.006 | 0.0015 | 0.0018 | 0.0021 | 0.0015 | 0.0032 | 0.0001 |
| | Vertical | 0.004 | 0.0035 | 0.0038 | 0.0021 | 0.0035 | 0.0032 | 0.0001 |
| | Diagonal | 0.006 | 0.0025 | 0.0038 | 0.0021 | 0.0035 | 0.0032 | 0.0001 |
| Lenna | Horizontal | 0.0039 | 0.0031 | −0.0221 | −0.0046 | 0.00321 | 0.0022 | −0.0017 |
| | Vertical | 0.0015 | −0.0021 | −0.0035 | 0.0021 | −0.0035 | −0.0020 | −0.0001 |
| | Diagonal | 0.0016 | −0.0035 | −0.0023 | −0.0033 | 0.0042 | 0.0031 | −0.0028 |
| Baboon | Horizontal | 0.0029 | 0.0031 | −0.0326 | −0.0076 | 0.00167 | 0.0036 | −0.0005 |
| | Vertical | 0.0015 | −0.0021 | −0.0035 | 0.0021 | −0.0035 | −0.0020 | 0.0001 |
| | Diagonal | 0.0016 | −0.0035 | −0.0023 | −0.0033 | 0.0042 | 0.0031 | −0.0001 |
| Aeroplane | Horizontal | 0.0039 | 0.0031 | −0.0221 | −0.0046 | 0.00321 | 0.0022 | −0.0002 |
| | Vertical | 0.0036 | −0.0065 | −0.0040 | 0.0031 | −0.0041 | −0.0031 | 0.0001 |
| | Diagonal | 0.0019 | −0.0039 | −0.0030 | −0.0040 | 0.0040 | 0.0039 | −0.0008 |

**Table 2**
Entropy analysis.

| Plaintext images | Ref (Hua and Zhou, 2016) | Ref (Zhu et al., 2019) | Ref (Ping et al., 2018) | Ref (Diab, 2018) | Ref (Pak and Huang, 2017) | Ref (Ye and Huang, 2017) | Proposed |
|---|---|---|---|---|---|---|---|
| Cameraman | 7.9671 | 7.9753 | 7.9720 | 7.9735 | 7.9881 | 7.9860 | 7.9990 |
| Lenna | 7.9965 | 7.9963 | 7.9865 | 7.9981 | 7.9986 | 7.9971 | 7.9991 |
| Baboon | 7.9796 | 7.9880 | 7.9983 | 7.9983 | 7.9975 | 7.9986 | 7.9990 |
| Aeroplane | 7.9899 | 7.9732 | 7.9734 | 7.9878 | 7.9860 | 7.9856 | 7.9989 |
| Color | | | | | | | R:7.9997 |
| Lena image | - | - | - | - | - | - | G:7.9991 B:7.9989 |

$$Contrast = \sum_{a,b=0} |a-b|^2 \gamma(a-b) \tag{24}$$

where, $a$ and $b$, are two gray level 8-bit images. The occurrence of gray levels in these images is represented by $\gamma(a-b)$. Table 3 compares the contrast values of the proposed encryption algorithm and the existing encryption algorithm. Upon examining the data, it becomes evident that the proposed algorithm outperforms the existing one in terms of contrast values.

#### 4.1.5. Homogeneity analysis

The gray level co-occurrence matrix (GLCM) is a representation of the occurrence of various combinations of pixel brightness. By analyzing how the brightness values are distributed in the GLCM, one can determine how similar they are to the diagonal. A lower homogeneity measure indicates more effective encryption. Based on Table 4, the proposed encryption scheme is more effective than existing schemes. Eq. 25 can be used to calculate the homogeneity values.

$$Homogebeity = \sum_{A}\sum_{B} \frac{O(A,B)}{1+|A-B|} \tag{25}$$

where $A$ and $B$ correspond to the rows and columns of the plaintext image $O(A,B)$, respectively, in terms of pixels.

#### 4.1.6. Energy analysis

The energy of an image refers to the level of information it contains. Images that contain a greater amount of information are considered to have higher energy levels. When it comes to encrypting images, it's important for the encryption algorithm to generate ciphertext images that contain minimal visible information in order to ensure strong encryption. The mathematical expression for calculating the energy of an image is as follows:

$$Energy = \sum \gamma(A,B)^2 \tag{26}$$

where, $\gamma(A,B)$ is the GLCM of an enciphered image.

Table 5 compares the energy values of plaintext images, an existing encryption scheme, and a proposed encryption scheme. Based on the values given in Table 5, it can be analyzed that the proposed encryption scheme performs better in terms of energy analysis.

#### 4.1.7. Lossless analysis

For an encryption algorithm to retrieve the original pixel values of an image accurately, it must be lossless. The two most commonly used terms to determine the losslessness of an encryption algorithm are Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE). PSNR measures the difference between the signal strength and noise of the original and encrypted images, while MSE calculates the average squared difference between the pixel values of the two images.

By comparing the PSNR and MSE values of the original and encrypted images, we can assess the level of distortion introduced by the encryption process. A high PSNR and low MSE indicate that the encryption algorithm is relatively lossless, while a low PSNR and high MSE indicate a greater level of distortion and potential loss of information during encryption. These statistical values for PSNR and MSE can be determined using Eqs. 27 and 28, respectively.

$$MSE = \frac{1}{AB}\sum_{i=0}^{A-1}\sum_{j=0}^{B-1}(P(i,j)-E(i,j))^2 \tag{27}$$

$$PSNR = 10 \times log_2 \frac{M_{max}^2}{MSE} \tag{28}$$

The dimensions of an image are denoted by $A$ and $B$, which represent the number of rows and columns, respectively. The plain image and the encrypted image are denoted as $P(i,j)$ and $E(i,j)$, respectively. Additionally, $M_{max}$ denotes the highest value present in the plain image.

When it comes to evaluating the quality of images, PSNR and MSE are two metrics that are often used. PSNR measures the level

of similarity between the original and encrypted images, with higher values indicating greater similarity. However, in image encryption, a high PSNR is not desirable as it indicates a weak encryption algorithm. Strong encryption algorithms result in minimum PSNR values, indicating that the plaintext and ciphertext images are dissimilar.

On the other hand, MSE measures the difference between two images, where higher values indicate greater differences. In strong encryption, high MSE values are desirable, as they signify that the plaintext and ciphertext images are vastly different from each other.

In order to assess the efficacy of the proposed lossless encryption algorithm, both the PSNR and MSE values were calculated and subsequently presented in Tables 6 and 7. The proposed scheme shows zero PSNR and infinite MSE values, indicating that the algorithm effectively encrypts images without compromising their quality. In contrast, other comparable schemes show non-zero PSNR and finite MSE values, making them unsuitable for applications requiring exact pixel values.

### 4.2. Differential analysis

Differential analysis is an analytical technique that examines the impact of modifications in the input data on the output. This approach is often used in image encryption, and two widely used measures for differential analysis are NPCR and UACI. To calculate NPCR and UACI, Eqs. 29 and 30 are typically used. These measures are used for evaluating the security of image encryption schemes and determining their resistance to differential attacks.

$$NPCR = \frac{\sum_{i,j} D(i,j)}{A \times B} \times 100\% \tag{29}$$

where, $D(i,j) = 0, \text{if} E_1(i,j) = E_2(i,j) \text{and} D(i,j) = 1, \text{if} E_1(i,j) \neq E_2(i,j)$

$$UACI = \frac{1}{A \times B}\left[\sum_{i,j} \frac{|E_1(i,j) - E_2(i,j)|}{255}\right] \tag{30}$$

The variables $E_1$ and $E_2$ refer to the cipher images prior to and following the alteration of a single pixel value, while $A$ and $B$ denote the number of pixels rows and columns in an image.

It is evident from Table 8 that the proposed encryption scheme map shows the highest NPCR values than that of the existing ones for all four images A higher NPCR value implies that the encryption algorithm has a better ability to resist the differential attack of intruders.

The data in Table 8 illustrates that the proposed encryption scheme produces higher NPCR values compared to existing schemes for all four images. A higher NPCR value indicates that the encryption algorithm is more effective in protecting against differential attacks by unauthorized individuals.

Additionally, upon comparing the UACI values of both the proposed and existing algorithms, it becomes evident that the pro-

**Table 3**
Contrast analysis.

| Plaintext images | Ref (Hua and Zhou, 2016) | Ref (Zhu et al., 2019) | Ref (Ping et al., 2018) | Ref (Diab, 2018) | Ref (Pak and Huang, 2017) | Ref (Ye and Huang, 2017) | Proposed |
|---|---|---|---|---|---|---|---|
| Cameraman | 9.6886 | 9.9689 | 9.6789 | 9.8980 | 9.7387 | 9.8732 | 10.9889 |
| Lenna | 9.9026 | 9.8010 | 9.9060 | 9.8841 | 9.8982 | 9.8066 | 10.3011 |
| Baboon | 9.9884 | 9.8012 | 9.9250 | 9.9113 | 9.9075 | 9.9073 | 10.7190 |
| Aeroplane | 9.9841 | 9.7080 | 9.8010 | 9.9020 | 9.9111 | 9.9662 | 10.4160 |
| Color | | | | | | | R:10.6478 |
| Lena image | - | - | - | - | - | - | G:10.6710 B:10.3479 |

**Table 4**
Homogeneity analysis.

| Plaintext images | Ref (Hua and Zhou, 2016) | Ref (Zhu et al., 2019) | Ref (Ping et al., 2018) | Ref (Diab, 2018) | Ref (Pak and Huang, 2017) | Ref (Ye and Huang, 2017) | Proposed |
|---|---|---|---|---|---|---|---|
| Cameraman | 0.5036 | 0.5078 | 0.4706 | 0.6098 | 0.4779 | 0.4878 | 0.4432 |
| Lenna | 0.5033 | 0.5077 | 0.4216 | 0.5099 | 0.4799 | 0.4998 | 0.4633 |
| Baboon | 0.5099 | 0.5096 | 0.4763 | 0.5096 | 0.4797 | 0.4888 | 0.4400 |
| Aeroplane | 0.4966 | 0.4996 | 0.4768 | 0.5086 | 0.4896 | 0.4961 | 0.4350 |
| Color | | | | | | | R:0.4124 |
| Lena image | - | - | - | - | - | - | G:0.4317 B:0.4112 |

**Table 5**
Energy analysis.

| Plaintext images | Ref (Hua and Zhou, 2016) | Ref (Zhu et al., 2019) | Ref (Ping et al., 2018) | Ref (Diab, 2018) | Ref (Pak and Huang, 2017) | Ref (Ye and Huang, 2017) | Proposed |
|---|---|---|---|---|---|---|---|
| Cameraman | 0.0160 | 0.0159 | 0.0169 | 0.0168 | 0.0165 | 0.0165 | 0.0154 |
| Lenna | 0.0163 | 0.0162 | 0.0160 | 0.0165 | 0.0160 | 0.0161 | 0.0152 |
| Baboon | 0.0160 | 0.0161 | 0.0162 | 0.0162 | 0.0163 | 0.0162 | 0.0155 |
| Aeroplane | 0.0160 | 0.0159 | 0.0160 | 0.0162 | 0.0164 | 0.0163 | 0.0155 |
| Color | | | | | | | R:0.0154 |
| Lena image | - | - | - | - | - | - | G:0.0156 B:0.0155 |

**Table 6**
MSE analysis.

| Plaintext images | Ref (Hua and Zhou, 2016) | Ref (Zhu et al., 2019) | Ref (Ping et al., 2018) | Ref (Diab, 2018) | Ref (Pak and Huang, 2017) | Ref (Ye and Huang, 2017) | Proposed |
|---|---|---|---|---|---|---|---|
| Cameraman | 5.88 | 8.31 | 6.16 | 5.13 | 3.68 | 6.74 | 0 |
| Lenna | 6.43 | 9.54 | 5.32 | 3.80 | 4.15 | 3.39 | 0 |
| Baboon | 3.32 | 8.18 | 4.06 | 3.39 | 3.71 | 4.91 | 0 |
| Aeroplane | 7.39 | 6.05 | 9.95 | 8.19 | 3.38 | 4.96 | 0 |
| Color Lena image | - | - | - | - | - | - | R: 0 G: 0 B: 0 |

**Table 7**
PSNR analysis.

| Plaintext images | Ref (Hua and Zhou, 2016) | Ref (Zhu et al., 2019) | Ref (Ping et al., 2018) | Ref (Diab, 2018) | Ref (Pak and Huang, 2017) | Ref (Ye and Huang, 2017) | Proposed |
|---|---|---|---|---|---|---|---|
| Cameraman | 202.36 | 209.38 | 189.30 | 198.65 | 192.30 | 216.10 | $\infty$ |
| Lenna | 216.35 | 218.81 | 196.31 | 207.30 | 209.95 | 207.37 | $\infty$ |
| Baboon | 207.98 | 239.16 | 207.90 | 219.16 | 207.37 | 213.16 | $\infty$ |
| Aeroplane | 228.73 | 216.38 | 215.97 | 205.19 | 215.76 | 207.20 | $\infty$ |
| Color Lena image | - | - | - | - | - | - | R: $\infty$ G: $\infty$ B:$\infty$ |

**Table 8**
NPCR analysis.

| Plaintext images | Ref (Hua and Zhou, 2016) | Ref (Zhu et al., 2019) | Ref (Ping et al., 2018) | Ref (Diab, 2018) | Ref (Pak and Huang, 2017) | Ref (Ye and Huang, 2017) | Proposed |
|---|---|---|---|---|---|---|---|
| Cameraman | 33.3850 | 33.4871 | 33.3925 | 33.3068 | 33.2067 | 33.6037 | 33.6596 |
| Lenna | 33.4950 | 33.5072 | 33.3925 | 33.6021 | 33.6031 | 33.6031 | 33.6654 |
| Baboon | 33.3936 | 33.4945 | 33.4036 | 33.4078 | 33.4015 | 33.6076 | 33.6767 |
| Aeroplane | 33.3965 | 33.5037 | 33.4912 | 33.6034 | 33.4082 | 33.4037 | 33.6532 |
| Color Lena image | - | - | - | - | - | - | R:33.6474 G: 33.6512 B: 33.6651 |

posed encryption scheme yields higher UACI values than those found in the literature, as presented in Table 9.

### 4.3. Key sensitivity analysis

slightly different seed values are used to analyze the behavior of the chaotic maps that can provide insights into the overall performance of such chaotic maps used in the proposed work, such as Arnold's cat map, the chaotic tent map and the chaotic sine map. The original and the slightly changed seed values for different chaotic maps are as follows:

**Original seed values:**
For Arnolds' cat map: $y_m$ = 0.3000000000000000, $z_m$ = 0.3500000000000000,
For chaotic sine map: $g_m$ = 0.5100000000000000, $q_m$ = 3.3500000000000000,
For chaotic tent map: $h_m$0.3200000000000000 =, $u$ = 0.5200000000000000.
**Original seed values:**
For Arnolds' cat map: $y'_m$ = 0.3000000000000001, $z'_m$ = 0.3500000000000001,
For chaotic sine map: $g'_m$ = 0.5100000000000001, $q'_m$ = 3.3500000000000001,
For chaotic tent map: $h'_m$0.3200000000000001 =, $u'$ = 0.5200000000000001.

The above-mentioned pairs of seed values are utilized to decrypt the original image from the encrypted one. Fig. 12 depicts the sensitivity analysis of the secret keys used in the proposed encryption process. The amount of information recovered from the image can be seen in Fig. 12(c) when the original seed values are used. Fig. 12(d) shows that even a minor modification in the seed values can lead to a completely different decrypted image from the original one. This demonstrates the high sensitivity of the decryption process to any changes in the secret keys.

### 4.4. Cropping attack analysis

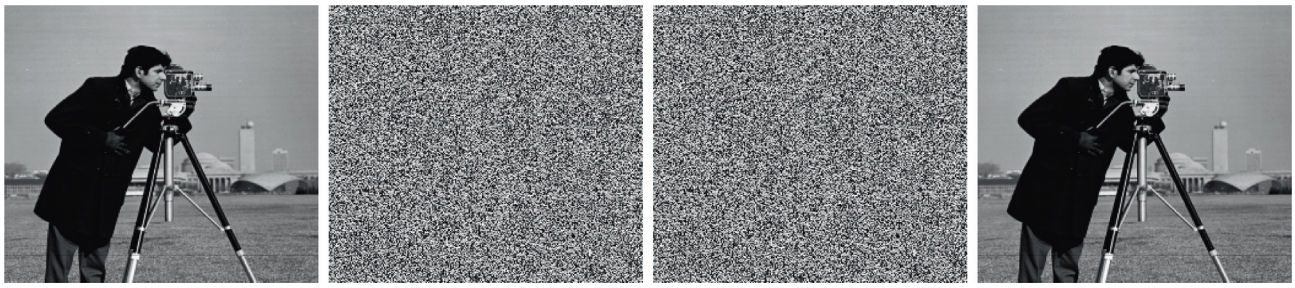
An attacker can try to extract information from an encrypted image through a cropping attack, which involves selecting a portion of the image. This type of attack is particularly effective if the encryption algorithm used to encrypt the image does not incorporate integrity checks or does not use a secure key. To evaluate the integrity of the proposed work, a portion of an encrypted image measuring 50 $\times$ 50 is cropped as shown in Fig. 13c) and and used to decrypt the plaintext data. The decrypted image is compared with the original image, and while there is some distortion, there is a negligible difference between the two images. This indicates that the decrypted image retains the original information with a high degree of accuracy (13(d)).

### 4.5. Noise attack analysis

A noise attack is a type of attack that aims to disrupt or degrade the quality of an encrypted image by introducing noise. The noise may be introduced intentionally by an attacker or may be a result of errors in the encryption process. To evaluate whether the pro-

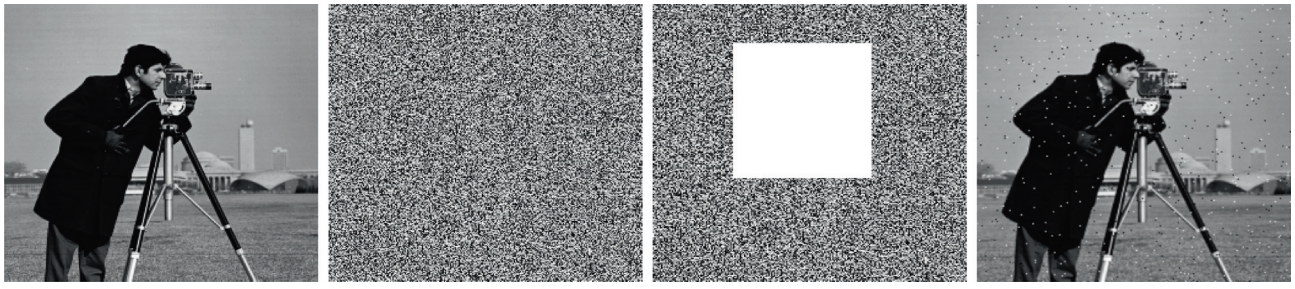
**Table 9**
UACI analysis.

| Plaintext images | Ref (Hua and Zhou, 2016) | Ref (Zhu et al., 2019) | Ref (Ping et al., 2018) | Ref (Diab, 2018) | Ref (Pak and Huang, 2017) | Ref (Ye and Huang, 2017) | Proposed |
|---|---|---|---|---|---|---|---|
| Cameraman | 33.5123 | 33.3965 | 33.1965 | 33.4036 | 33.4027 | 33.4013 | 33.6532 |
| Lenna | 33.4136 | 33.3021 | 33.3687 | 33.6031 | 33.6001 | 33.6012 | 33.6615 |
| Baboon | 33.4961 | 33.3024 | 33.3964 | 33.4031 | 33.4012 | 33.6033 | 33.6723 |
| Aeroplane | 33.4856 | 33.4021 | 33.4954 | 33.4012 | 33.4035 | 33.4011 | 33.6899 |
| Color Lena image | - | - | - | - | - | - | R: 33.6487 G: 33.6145 B: 33.6554 |



(a) Cameraman image (b) Enciphered image (c) Decryption using original seed values (d) Decryption using slightly modified seed values

**Fig. 12.** Key-sensitivity analysis.



(a) Cameraman image (b) Enciphered image (c) Cropped image (d) Decrypted image

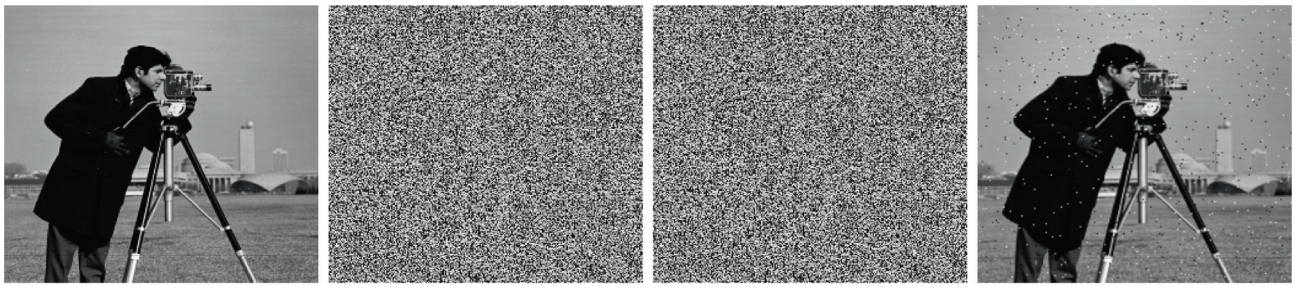
**Fig. 13.** Cropping attack analysis.

posed encryption scheme is resistant to noise, ciphertext pixels are contaminated with noise, which is added using the XOR operation as given in Eq. 31.

$$C_{Noise} = C_{encrypt} \oplus (00001110) \tag{31}$$

where, $C_{encrypt}$ and $C_{Noise}$ are the encrypted image and the contaminated image with noise (00001110), and $\oplus$ is the XOR operator.

The noise attack analysis depicted in Fig. 14 demonstrates that the information present in the decrypted image obtained from



(a) Cameraman image (b) $C_{encrypt}$ (c) $C_{Noise}$ (d) Decrypted image

**Fig. 14.** Noise attack analysis.

**Table 10**
Computational time analysis.

| Plaintext images | Ref (Hua and Zhou, 2016) | Ref (Zhu et al., 2019) | Ref (Ping et al., 2018) | Ref (Diab, 2018) | Ref (Pak and Huang, 2017) | Ref (Ye and Huang, 2017) | Proposed |
|---|---|---|---|---|---|---|---|
| Camerman | 2.2313 | 0.149 | 0.5681 | 0.4449 | 0.2444 | 0.6573 | 0.0048 |
| Lenna | 2.2410 | 0.6512 | 0.4271 | 0.7571 | 0.1812 | 0.7971 | 0.0017 |
| Baboon | 3.6310 | 0.1187 | 0.3471 | 0.7641 | 0.5578 | 0.6572 | 0.0022 |
| Aeroplane | 2.4492 | 0.7612 | 0.4387 | 0.4978 | 0.7349 | 0.4963 | 0.0033 |
| Color Lena image | - | - | - | - | - | - | R: 0.0034 G: 0.0043 B: 0.0033 |

$C_{Noise}$ can be observed. The recovered image contains a substantial amount of information with minimal distortion. This shows that the proposed encryption technique is also capable of withstanding noise attack analysis.

### 4.6. Computational time analysis

Apart from security concerns, the efficiency of an image cryptosystem is also crucial, especially for real-time internet applications. To assess the effectiveness of the proposed approach, a computational time analysis was carried out to compare the encryption speed of images of varying sizes using both the proposed scheme and existing encryption schemes. The tests are conducted on a computer equipped with a 2.4 GHz Intel Core i5 processor and 8 GB of RAM. As shown in Table 10, the proposed KSA-DWT-IET outperforms other existing schemes in terms of encryption speed. Given its fast encryption speed, the KSA-DWT-IET is well-suited for secure, real-time image transmission over broadband networks, where encryption time should be kept to a minimum in comparison to transmission time.

## 5. Conclusion

The proposed method for the encryption of digital images has two primary objectives: The first is to ensure that the security level of the encrypted digital images is high, while the second is to make the encryption process computationally efficient enough to be used in real-time applications. To achieve these goals, the proposed method, KSA-DWT-IET, includes two major components. The first component is key scheming using weighted addition (KSWA), and the second component is a substitution technique that contains MGR, DSB, and MRS stages. Both of these components are utilized in unison to introduce confusion and diffusion in the plaintext image, in order to offer a high level of security while simultaneously reducing the amount of computational time required for encryption. The simulation results indicate that the KSA-DWT-IET can resist both statistical and differential attacks. Furthermore, to ensure that the proposed encryption scheme is useful in real-time applications, a computational time analysis is performed, and it is found that the encryption of a plaintext image can be completed in less than 3 ms.

The limitation of the proposed encryption scheme is that it may be susceptible to chosen-plaintext attacks. This means that an attacker could potentially gain access to the encryption key by selecting specific plaintexts and observing the corresponding ciphertexts, allowing them to deduce information about the key.

To overcome this vulnerability, the proposed method can be augmented with a message authentication code (MAC) or a digital signature in the future. This cryptographic technique can ensure that the ciphertexts generated by the encryption process are not modified by an attacker and that the key used to generate the ciphertexts is authentic.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

The authors extend their gratitude to the Deanship of Scientific Research at King Khalid University for funding this work through the research group's program under Grant No. R. G. P. 2/5/44.

## References

Abuelyman, E.S., Alsehibani, A.-A.S., 2008. An optimized implementation of the s-box using residue of prime numbers. Int. J. Comput. Sci. Network Sec. 8 (4), 304–309.
Agrawal, S., Panda, R., Mishro, P.K., Abraham, A., 2022. A novel joint histogram equalization based image contrast enhancement. J. King Saud Univ.-Comput. Informat. Sci. 34 (4), 1172–1182.
Akraam, M., Rashid, T., Zafar, S., 2022. An image encryption scheme proposed by modifying chaotic tent map using fuzzy numbers. Multimedia Tools Appl., 1–19
Alarifi, A., Sankar, S., Altameem, T., Jithin, K., Amoon, M., El-Shafai, W., 2020. A novel hybrid cryptosystem for secure streaming of high efficiency h. 265 compressed videos in iot multimedia applications. IEEE Access 8, 128548–128573.
Alawida, M., Teh, J.S., Mehmood, A., Shoufan, A., et al., 2022. A chaos-based block cipher based on an enhanced logistic map and simultaneous confusion-diffusion operations. J. King Saud Univ.-Comput. Informat. Sci. 34 (10), 8136–8151.
Alqahtani, F., Amoon, M., El-Shafai, W., 2022. A fractional fourier based medical image authentication approach. CMC-Comput. Mater. Continua 70 (2), 3133–3150.
Amdouni, R., Gafsi, M., Guessmi, R., Hajjaji, M.A., Mtibaa, A., Bourennane, E.-B., 2022. High-performance hardware architecture of a robust encryption block-cipher algorithm based on different chaotic maps and dna sequence encoding. Integration.
Amina, S., Mohamed, F.K., 2018. An efficient and secure chaotic cipher algorithm for image content preservation. Commun. Nonlinear Sci. Numer. Simul. 60, 12–32.
Bedoui, M., Mestiri, H., Bouallegue, B., Hamdi, B., Machhout, M., 2022. An improvement of both security and reliability for aes implementations. J. King Saud Univ.-Comput. Informat. Sci. 34 (10), 9844–9851.
Cao, C., Sun, K., Liu, W., 2018. A novel bit-level image encryption algorithm based on 2d-licm hyperchaotic map. Signal Process. 143, 122–133.
Chai, X., 2017. An image encryption algorithm based on bit level brownian motion and new chaotic systems. Multimedia Tools Appl. 76, 1159–1175.
Chai, X., Fu, J., Gan, Z., Lu, Y., Zhang, Y., 2022a. An image encryption scheme based on multi-objective optimization and block compressed sensing. Nonlinear Dyn. 108 (3), 2671–2704.
Chai, X., Wang, Y., Chen, X., Gan, Z., Zhang, Y., 2022b. Tpe-gan: thumbnail preserving encryption based on gan with key. IEEE Signal Process. Lett. 29, 972–976.
Chen, J., Chen, L., Zhou, Y., 2020. Cryptanalysis of a dna-based image encryption scheme. Inf. Sci. 520, 130–141.
Diab, H., 2018. An efficient chaotic image cryptosystem based on simultaneous permutation and diffusion operations. IEEE Access 6, 42227–42244.
Diaconu, A.-V., 2014. An image encryption algorithm with a chaotic dynamical system based sudoku grid. In: 2014 10th International Conference on Communications (COMM). IEEE, pp. 1–4.

El-Shafai, W., Khallaf, F., El-Rabaie, E.-S.M., El-Samie, F.E.A., 2021. Robust medical image encryption based on dna-chaos cryptosystem for secure telemedicine and healthcare applications. J. Ambient Intell. Humanized Comput. 12, 9007–9035.
El-Shafai, W., Almomani, I.M., Alkhayer, A., 2021. Optical bit-plane-based 3d-jst cryptography algorithm with cascaded 2d-frft encryption for efficient and secure hevc communication. IEEE Access 9, 35004–35026.
El-Shafai, W., Khallaf, F., El-Rabaie, E.-S.M., El-Samie, F.E.A., 2022a. Proposed 3d chaos-based medical image cryptosystem for secure cloud-iomt ehealth communication services. J. Ambient Intell. Humanized Comput., 1–28
El-Shafai, W., Aly, M., Algarni, A., Abd El-Samie, F.E., Soliman, N.F., 2022b. Secure and robust optical multi-stage medical image cryptosystem. CMC-Comput. Mater. Continua 70 (1), 895–913.
Faragallah, O.S., Alzain, M.A., El-Sayed, H.S., Al-Amri, J.F., El-Shafai, W., Afifi, A., Naeem, E.A., Soh, B., 2018. Block-based optical color image encryption based on double random phase encoding. IEEE Access 7, 4184–4194.
Faragallah, O.S., Afifi, A., El-Shafai, W., El-Sayed, H.S., Naeem, E.A., Alzain, M.A., Al-Amri, J.F., Soh, B., Abd El-Samie, F.E., 2020a. Investigation of chaotic image encryption in spatial and frft domains for cybersecurity applications. IEEE Access 8, 42491–42503.
Faragallah, O.S., AlZain, M.A., El-Sayed, H.S., Al-Amri, J.F., El-Shafai, W., Afifi, A., Naeem, E.A., Soh, B., 2020b. Secure color image cryptosystem based on chaotic logistic in the frft domain. Multimedia Tools Appl. 79, 2495–2519.
Faragallah, O.S., Afifi, A., El-Shafai, W., El-Sayed, H.S., Alzain, M.A., Al-Amri, J.F., Abd El-Samie, F.E., 2020c. Efficiently encrypting color images with few details based on rc6 and different operation modes for cybersecurity applications. IEEE Access 8, 103200–103218.
Faragallah, O.S., El-sayed, H.S., Afifi, A., El-Shafai, W., 2021. Efficient and secure opto-cryptosystem for color images using 2d logistic-based fractional fourier transform. Opt. Lasers Eng. 137, 106333.
Fridrich, J., 1998. Symmetric ciphers based on two-dimensional chaotic maps. Int. J. Bifurcat. Chaos 8 (06), 1259–1284.
Gondal, M.A., Anees, A., 2013. Analysis of optimized signal processing algorithms for smart antenna system. Neural Comput. Appl. 23, 1083–1087.
Haar, A., 1909. Zur theorie der orthogonalen funktionensysteme. Georg-August-Universitat, Gottingen..
Hoang, T.M., 2022. A novel design of multiple image encryption using perturbed chaotic map. Multimedia Tools Appl. 81 (18), 26535–26589.
Holschneider, M., 1988. On the wavelet transformation of fractal objects. J. Stat. Phys. 50, 963–993.
Hosny, K.M., Kamal, S.T., Darwish, M.M., 2023. A novel color image encryption based on fractional shifted gegenbauer moments and 2d logistic-sine map. Visual Comput. 39 (3), 1027–1044.
Hua, Z., Zhou, Y., 2016. Image encryption using 2d logistic-adjusted-sine map. Inf. Sci. 339, 237–253.
Hua, Z., Zhou, Y., Pun, C.-M., Chen, C.P., 2015. 2d sine logistic modulation map for image encryption. Inf. Sci. 297, 80–94.
Huang, X., Ye, G., 2014. An efficient self-adaptive model for chaotic image encryption algorithm. Commun. Nonlinear Sci. Numer. Simul. 19 (12), 4094–4104.
Khan, M.A.M., Azam, N.A., Hayat, U., Kamarulhaili, H., 2023. A novel deterministic substitution box generator over elliptic curves for real-time applications. J. King Saud Univ.-Comput. Informat. Sci. 35 (1), 219–236.
Lai, Q., Hu, G., Erkan, U., Toktas, A., 2023. A novel pixel-split image encryption scheme based on 2d salomon map. Expert Syst. Appl. 213, 118845.
Liao, X., Kulsoom, A., Ullah, S., 2016. A modified (dual) fusion technique for image encryption using sha-256 hash and multiple chaotic maps. Multimedia Tools Appl. 75 (18), 11241–11266.
Liu, X., Tong, X., Wang, Z., Zhang, M., 2022. A new n-dimensional conservative chaos based on generalized hamiltonian system and its' applications in image encryption. Chaos Solitons Fractals 154, 111693.
Masood, F., Boulila, W., Alsaeedi, A., Khan, J.S., Ahmad, J., Khan, M.A., Rehman, S.U., 2022. A novel image encryption scheme based on arnold cat map, newton-leipnik system and logistic gaussian map. Multimedia Tools Appl. 81 (21), 30931–30959.
Mondal, B., Mandal, T., 2017. A light weight secure image encryption scheme based on chaos & dna computing. J. King Saud Univ.-Comput. Informat. Sci. 29 (4), 499–504.
Nan, S.-X., Feng, X.-F., Wu, Y.-F., Zhang, H., 2022. Remote sensing image compression and encryption based on block compressive sensing and 2d-lcccm. Nonlinear Dyn. 108 (3), 2705–2729.
Pak, C., Huang, L., 2017. A new color image encryption using combination of the 1d chaotic map. Signal Process. 138, 129–137.
Parameshachari, B., Kiran, R.P., Rashmi, P., Supriya, M., Rajashekarappa, M., Panduranga, H., 2019. Controlled partial image encryption based on lsic and chaotic map. In: ICCSP, pp. 60–63.
Ping, P., Fan, J., Mao, Y., Xu, F., Gao, J., 2018. A chaos based image encryption scheme using digit-level permutation and block diffusion. IEEE Access 6, 67581–67593.
Rehman, M.U., Shafique, A., Khalid, S., Hussain, I., 2021. Dynamic substitution and confusion-diffusion-based noise-resistive image encryption using multiple chaotic maps. IEEE Access 9, 52277–52291.
Rehman, M.U., Shafique, A., Ghadi, Y.Y., Boulila, W., Jan, S.U., Gadekallu, T.R., Driss, M., Ahmad, J., 2022. A novel chaos-based privacy-preserving deep learning model for cancer diagnosis. IEEE Trans. Network Sci. Eng. 9 (6), 4322–4337.
Sharma, M., 2020. Image encryption based on a new 2d logistic adjusted logistic map. Multimedia Tools Appl. 79 (1–2), 355–374.
Tong, X.-J., Zhang, M., Wang, Z., Liu, Y., Xu, H., Ma, J., 2015. A fast encryption algorithm of color image based on four-dimensional chaotic system. J. Vis. Commun. Image Represent. 33, 219–234.
Wang, X., Guan, N., 2022. 2d sine-logistic-tent-coupling map for image encryption. J. Ambient Intell. Humanized Comput., 1–21
Wang, S., Peng, Q., Du, B., 2022. Chaotic color image encryption based on 4d chaotic maps and dna sequence. Opt. Laser Technol. 148, 107753.
Ye, G., Huang, X., 2017. An efficient symmetric image encryption algorithm based on an intertwining logistic map. Neurocomputing 251, 45–53.
Zhu, H., Zhao, Y., Song, Y., 2019. 2d logistic-modulated-sine-coupling-logistic chaotic map for image encryption. IEEE Access 7, 14081–14098.
Zhu, H., Ge, J., Qi, W., Zhang, X., Lu, X., 2022. Dynamic analysis and image encryption application of a sinusoidal-polynomial composite chaotic system. Mathe. Comput. Simul. 198, 188–210.