mathematics

MDPI

Article

Dual-Domain Image Encryption in Unsecure Medium—A Secure Communication Perspective

Hemalatha Mahalingam [1], Thanikaiselvan Veeramalai [2,*], Anirudh Rajiv Menon [2], Subashanthini S. [3] and Rengarajan Amirtharajan [4,*]

[1] Department of Electrical and Computer Engineering, King Abdulaziz University, Jeddah 22254, Saudi Arabia
[2] School of Electronics Engineering, Vellore Institute of Technology, Vellore 632014, India
[3] School of Information Technology, Vellore Institute of Technology, Vellore 632014, India
[4] School of Electrical & Electronics Engineering, SASTRA Deemed University Thanjavur, Thanjavur 613401, India
* Correspondence: thanikaiselvan@vit.ac.in (T.V.); amir@ece.sastra.edu (R.A.); Tel.: +91-4362-264101 (R.A.)

Abstract: With the growing demand for digitalization, multimedia data transmission through wireless networks has become more prominent. These multimedia data include text, images, audio, and video. Therefore, a secure method is needed to modify them so that such images, even if intercepted, will not be interpreted accurately. Such encryption is proposed with a two-layer image encryption scheme involving bit-level encryption in the time-frequency domain. The top layer consists of a bit of plane slicing the image, and each plane is then scrambled using a chaotic map and encrypted with a key generated from the same chaotic map. Next, image segmentation, followed by a Lifting Wavelet Transform, is used to scramble and encrypt each segment's low-frequency components. Then, a chaotic hybrid map is used to scramble and encrypt the final layer. Multiple analyses were performed on the algorithm, and this proposed work achieved a maximum entropy of 7.99 and near zero correlation, evidencing the resistance towards statistical attacks. Further, the keyspace of the cryptosystem is greater than 2^{128}, which can effectively resist a brute force attack. In addition, this algorithm requires only 2.1743 s to perform the encryption of a 256 $\times$ 256 sized 8-bit image on a host system with a Windows 10 operating system of 64-bit Intel(R) Core(TM) i5-7200U CPU at 2.5 GHz with 8 GB RAM.

Keywords: encryption; chaotic maps; integer wavelet transform; statistical attack; keyspace

MSC: 94A60

check for updates

Citation: Mahalingam, H.; Veeramalai, T.; Menon, A.R.; S., S.; Amirtharajan, R. Dual-Domain Image Encryption in Unsecure Medium—A Secure Communication Perspective. *Mathematics* **2023**, *11*, 457. https://doi.org/10.3390/math11020457

Academic Editor: Leimin Wang

Received: 17 December 2022
Revised: 5 January 2023
Accepted: 10 January 2023
Published: 15 January 2023

1. Introduction

COVID-19 and the subsequent lockdown made professionals and laypeople use mobile phones as gadgets for telephony and handy computers for daily activities such as attending online meetings, academic classes and discussions, writing documents, writing surveillance, etc. Specifically, the transfer of images plays a vital role in social media. Hence, the safe transmission of these images, as essential information carriers, in an unsecure medium is critical for ensuring confidentiality. Confidentiality protects the disclosure of data and can be achieved through image encryption. Third parties can glean confidential information if such images are intercepted, proving dangerous. This prolific interception has happened due to the immense development of wireless technology.

On the other hand, it is necessary to safeguard the gadget and its essential content. Hence, security solutions must be utilized. Encryption especially must be deployed in the gadget to preserve the data's confidentiality. Though the communication channel has authentication and security mechanisms, it is recommended to have indigenous encryption schemes for data security that resist third-party service providers and other channel intruders.

The Internet of Things becomes an Interconnection of Threats due to the overwhelming heterogeneous connectivity in networks. This opens a backdoor for attackers, intruders, and masqueraders to make the network system vulnerable. Nowadays, real-time monitoring has become an essential part of any industry. A camera interfaced with an edge device is utilized for monitoring purposes where the privacy of the images and videos are to be preserved. Image security offers confidentiality to the image and frames, such as video, to secure them from attackers. Though cloud storage offers good authentication, the cloud service provider can access the content against the individual's privacy policy. Hence, the image could be encrypted in the edge device and stored in the cloud, ensuring adequate data security. Furthermore, due to technological advancements, data are now easily transmitted over long distances. Hence, there is a need for an efficient encryption and scrambling technique that prevents third parties from understanding the communicated information [1,2].

Image encryption is a technique that modifies an image into an output that cannot be discerned by a third-party user [2]. This multimedia data can be personal, public, governmental, or military. Since governmental or military data are susceptible, they face significant threats while being transmitted online. Thus, information security over the internet is one of the most researched topics to protect individuals' privacy. Research has taken many data protection measures to transmit multimedia data securely over the internet for national security and safety. Data security protects data from unauthorized and illegal access [3,4] as these sensitive and critical data must be securely transmitted over the channel. For this purpose, network-specific data encryption is not trustworthy and consistent. Hence, we need another method to protect these sensitive data when transmitted over the channel [4]. The confidential data should not be able to be retrieved even when an unauthorized party has accessed the network routes. One of the most basic data security techniques is cryptography, in which researchers directly convert sensitive data into cryptosystems and transmit them through the channel. However, due to the rapidly changing types of multimedia files and increased computational power, cryptosystems can be easily hacked into, and sensitive data are very vulnerable.

Since the 1990s, chaotic systems have exhibited the randomness, unpredictability, and sensitivity of generated keys to the initial value required to design an effective cryptosystem. Furthermore, chaotic maps reduce the correlation between adjacent pixels; hence, they are useful in scrambling. Cryptography shows how chaos-based image encryption can be helpful because of the initial conditions' susceptibility [5]. These maps can be categorized into one-dimensional and higher-dimensional maps. However, the chaotic trajectories and parametric and commencing values of 1D chaotic maps such as Logistic maps, Sine maps, and tent maps can be extracted relatively easily [6,7]. Suneja et al. observed the advantages of higher dimensionality in chaos-based systems, where they conducted some augmentations on implementing 1D chaotic maps to combat these earlier method limitations [8].

Hua et al. [9] and Gao et al. [10] used a hybrid consisting of two existing 1D and 3D chaotic maps. Both methods proposed new 1D maps combined in series and parallel. The former had two parameters but became non-uniform for a range of those parameters. The latter showed substantial uniformity but relied only on one parameter. Compared to existing maps, the range across which chaotic behavior is depicted, randomness and susceptibility to commencing values is much better for the earlier maps. A significant improvement was seen for maps functioning over more than a single dimension. These chaotic maps, such as the Duffing map, Chirikov–Taylor map, Kaplan–Yorke map, Henon map, etc., have been preferred due to their complex structures and better chaotic performances. For example, Zhu et al. [11] implemented a 2D chaotic system for image encryption based on a scheme that couples a Sine map and a Logistic map-modulated Sine map. Theoretical analyses and simulations have shown the efficacy of such a higher dimensional map [12–20].

Wavelet transforms are used to operate on an image in the time-frequency domain. In a wavelet transform, a discrete signal is subjected to a high pass filter and a low pass filter and downsampled, the outputs of which are high (HP) and low (LP) frequency signals, both having half as many samples as the input signal [17–19,21]. The method proposed in [20] shows how a Discrete Wavelet Transform (DWT) can split an image into its wavelet sub-bands. Another image can then be embedded directly into the decomposed sub-bands for watermarking. Similarly, scrambling an image's sub-bands can lead to a decreased correlation between the pixels in a lower number of iterations. The Lifting Wavelet Transform (LWT) is a reversible wavelet transform that is computationally efficient, incorporates floating-point operations, and is easily implementable on hardware [19]. The Lifting Scheme is also faster and more efficient than other schemes. For instance, it has advantages over DWT, such as the transform's ability to retain invertibility despite a local modification.

Other advantages of the Lifting Scheme are that the inverse and forward transforms have precisely the same complexity, require less memory, and can be used on arbitrary geometries. In addition, this transform divides the input signal into odd-positioned and even-positioned sample sequences. Afterward, the transform is 'lifted' to a transform with the required properties by using one or more lifting filter operations, P and/or U. It analyses and depicts the various multi-resolution aspects of the LWT to propose an image compression system that consumes less power and operates at high-speeds and is known in the available literature on image security research [21–53].

The special contributions of this research work are listed below:

1. The usage of lift wavelet transform as the intermediate stage of the ciphering process to effectively reduce the operational time;
2. The development of a hybrid chaotic system to offer high keyspace;
3. The design of bit-level diffusion to break the pixel dependencies.

The rest of the paper is arranged as follows: Section 2 deals with LWT's related works and the proposed algorithm's chaotic maps. The chaotic maps used are a variety of 1D and 2D maps; the Gauss map, Piecewise Linear map, Tent map, Sine map, and Circle map are the 1D chaotic maps used, and the Kaplan–Yorke map, Chirikov–Taylor map, Duffing map, and Henon map [23–27] are the 2D maps used. Section 3 depicts the novel hybrid chaotic map proposed, Section 4 demonstrates its encryption and decryption methodology, and Sections 5 and 6 deal with its results and performance analysis, and provide a conclusion.

2. Background

This section explains the necessary preliminaries, such as lift wavelet transform, chaotic maps, and their characteristics. These concepts were adopted in the proposed methodology to accomplish the encryption process.

2.1. *Lifting Wavelet Transform*

Lifting wavelet transform [19] comprises three steps: Split, Predict, and Update, as depicted in Figure 1. Each step is explained below.

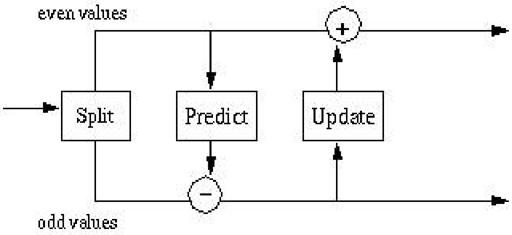


Figure 1. The architecture of the LWT sequence.

2.1.1. Split

Here, the entire signal is split into two sequences, the low-resolution part λ_j, which consists of pixels in even coordinates, and the high-resolution part γ_j (pixels in odd coordinates). This creates a checkerboard-like pattern, as shown in Figure 2.

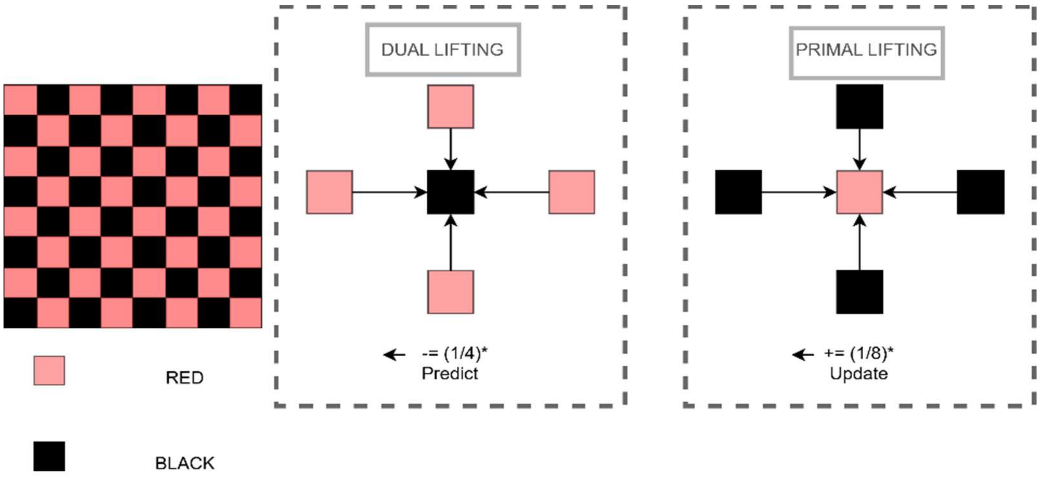


Figure 2. Primal and dual lifting with prediction and updating in the checkered pattern (* is 1/4 Predict Equation (1)).

2.1.2. Predict

A more powerful prediction mechanism will allow data representation more compactly. This mechanism is also known as dual lifting. Here, the high-resolution part of the data γ_j is predicted from its low-resolution part λ_j. When the signals have high degrees of correlation, such a prediction will be very accurate, hence only the portion of γ_j that is different from its prediction (the prediction error) needs to be stored.

Thus, γ_j is replaced with γ_j-P(λ_j), where P is the prediction operator. P is usually a linear interpolation operation. The closer P(λ_j) is to λ_j, the greater compression and efficiency of the transform. The values of γ_j are predicted by averaging the immediate horizontal and vertical neighbors in λ_j and replaced by their prediction errors:

$$\gamma_j = \gamma_j - \frac{1}{4}(Im(r-1,c) + Im(r,c-1) + Im(r,c+1) + Im(r+1,c)) \tag{1}$$

where $Im(r, c)$ is the value of the image pixel at position (r, c). The new γ_j represents the high-frequency component of the signal.

2.1.3. Update

This new representation, also known as primal lifting, causes a change in specific basic properties, such as the signal's mean. To preserve this property, a primal lifting step is required to update; hence, λ_j is updated with data computed from the new sequence γ_j.

Thus, λ_j is supplanted by λ_j + U(γ_j), where U is an updating operator. U is also a linear interpolation operation. The new λ_j represents the low-frequency component/approximation of the signal.

$$\lambda_j = \lambda_j - \frac{1}{8}(Im(r-1,c) + Im(r,c-1) + Im(r,c+1) + Im(r+1,c)) \tag{2}$$

Hence, the final output is a more compact representation of the signal's low-frequency and high-frequency components.

In Figure 2, the red squares represent λ_j and the black squares represent γ_j. The next resolution level can be found by splitting the λ_j sequence and repeating the same operations while considering the checkerboard to be rotated 45°.

λ_j is split into even (λ_{j-1}) and odd samples (γ_{j-1}) represented by blue and yellow squares, respectively, in Figure 3.

$$\lambda_{j-1} = \lambda_{j-1} + \frac{1}{8}(Im(r-1,c-1) + Im(r-1,c+1) + Im(r+1,c-1) + Im(r+1,c+1)) \tag{3}$$

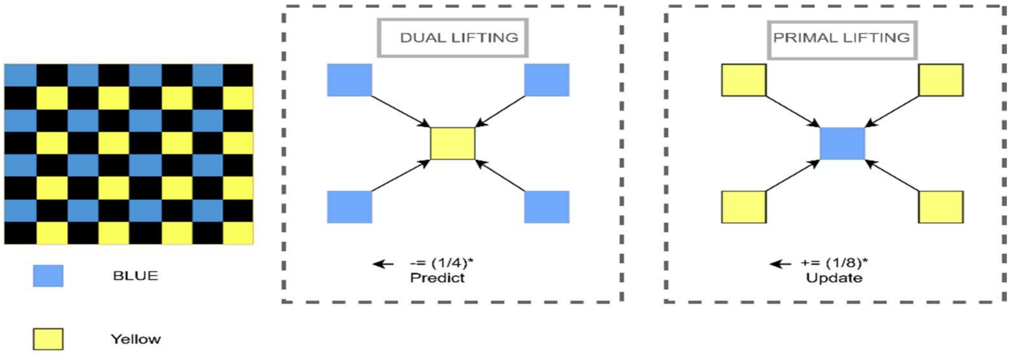


Figure 3. Next resolution level prediction and updating (* is 1/8 Update Equation (3) & 1/4 Predict Equation (4)).

Similarly, γ_{j-1} is given as:

$$\gamma_{j-1} = \gamma_{j-1} + \frac{1}{4}(Im(r-1,c-1) + Im(r-1,c+1) + Im(r+1,c-1) + Im(r+1,c+1)) \tag{4}$$

The values represented by the red and blue squares are low-frequency components, while those represented by the black and yellow squares are high-frequency components. For subsequent resolution, the blue squares, i.e., λ_{j-1}, must be operated on similarly.

Row processing is the one-dimensional decomposition of an image using LWT that splits the image into high and low-frequency parts. After the two-dimensional decomposition or row-column processing, the image is split into four frequency sub-bands, LL, LH, HL, and HH, as shown in Figure 4. LL is the most sensitive part of the decomposed image and contains most of the image's information. The LWT decomposition of Lena is shown in Figure 5.

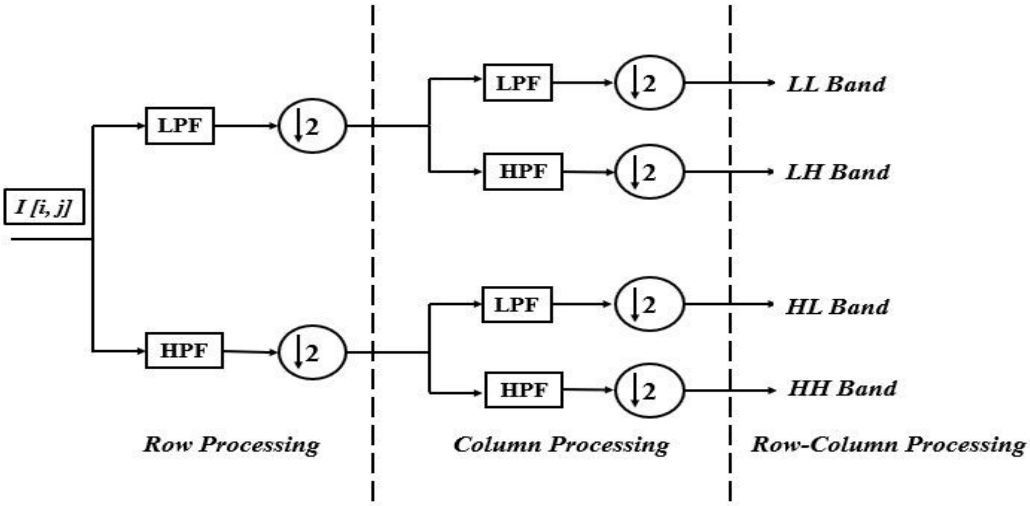


Figure 4. Lifting Wavelet Transform Decomposition.

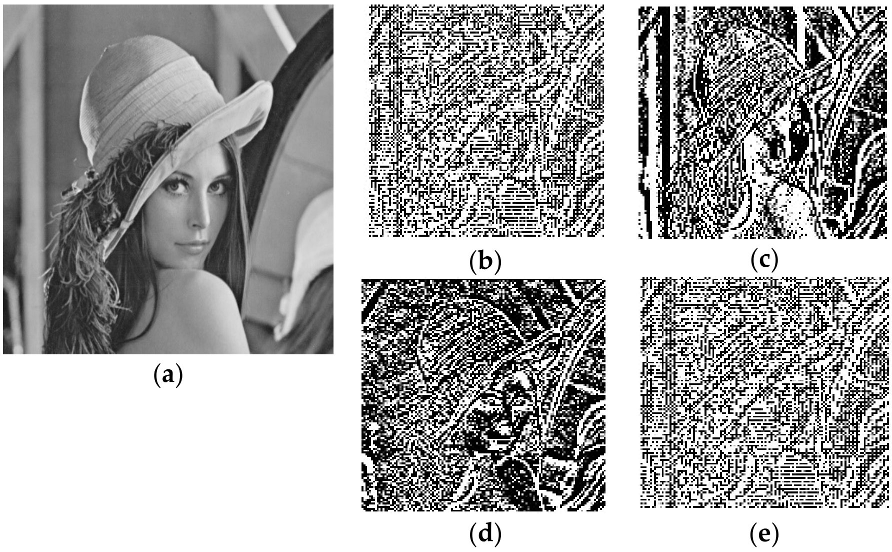


Figure 5. (**a**) Input Lena Image; (**b**) LL, (**c**) HL, (**d**) LH, and (**e**) HH sub-bands of decomposed Lena image after LWT.

2.2. Chaotic Maps

A chaotic map is an iteratively evolving function that exhibits chaotic behavior. The chaotic map also exhibits susceptibility to its initial conditions and the parameters its equation depends on. These properties are useful in image encryption and can significantly reduce the correlation between an image's pixels, thus successfully morphing it into an indiscernible image. There exist chaotic maps that operate over multiple dimensions as well. At the same time, the chaotic systems introduced in this paper are all classical discrete chaotic systems. Many high-performance discrete chaotic systems have been studied recently, such as in [36–53]. The chaotic maps used in the proposed scheme are described below.

2.2.1. Kaplan–Yorke Map

The Kaplan–Yorke map is a 2D chaotic map that maps a point in the (x, y) plane given below, and its behavior can be analyzed.

$$\begin{aligned} K(i+1) &= 2(K(i)) \\ Y(i+1) &= a(Y(i) + cos(4\pi K(i)) \end{aligned} \tag{5}$$

We use only the K dimension to generate a 1D chaotic array, but we use both dimensions in the proposed hybrid map. A plot of 2000 iterations of the Kaplan–Yorke map with a = 0.2 is shown in Figure 6. The initial K-value $K0$ has been chosen as depicted in Section 2.2, and its bifurcation diagram is shown in Figure 7c.

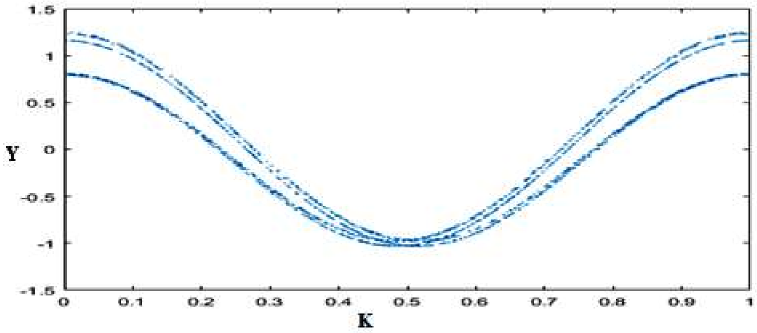


Figure 6. The plot of the Kaplan–Yorke map for 1000 iterations with a = 0.2.

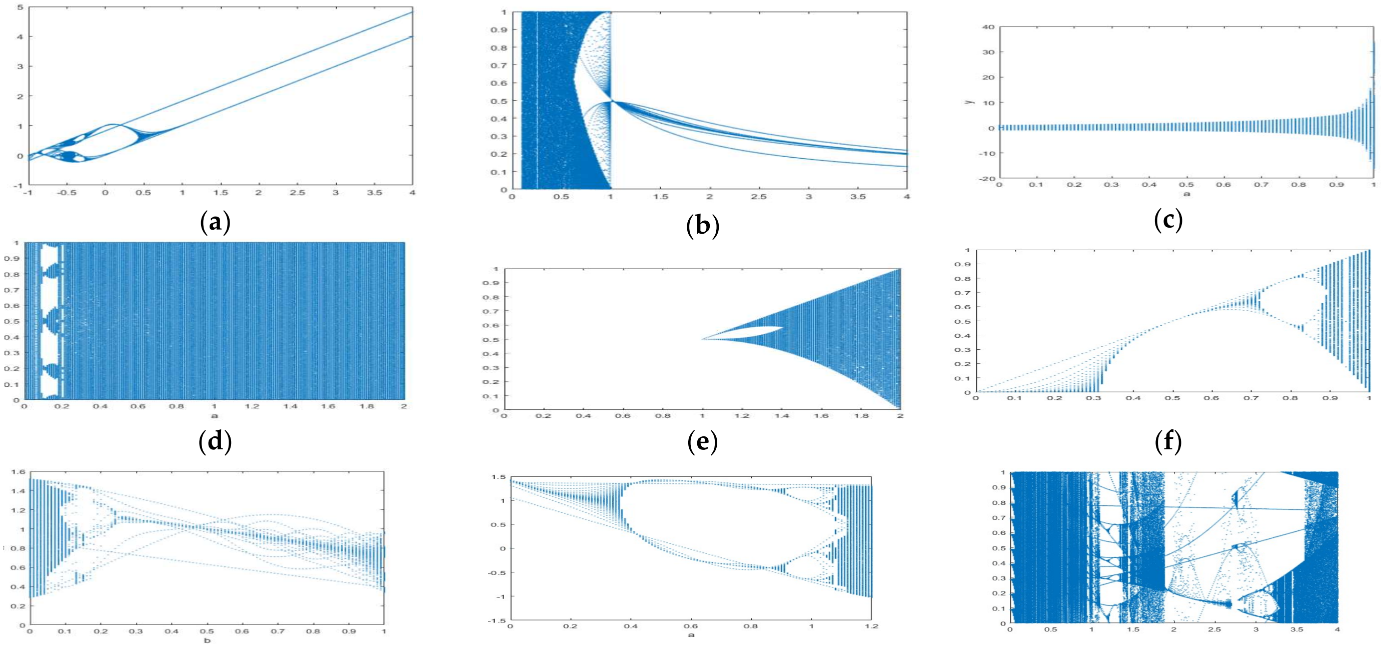


Figure 7. Bifurcation diagrams of (**a**) Gauss map (g vs. α); (**b**) PWLCM map (C vs. a); (**c**) Kaplan–Yorke map (y vs. α); (**d**) Chirikov–Taylor map (C vs. α); (**e**) Tent map (T vs. μ); (**f**) Sine map from 0 to 1 (s vs. α); (**g**) Duffing map (M vs. b); (**h**) Henon map (E vs. α); and (**i**) Circle map (O vs. K).

2.2.2. Gauss Map

The Gauss map is a 1D map that uses many control parameters to strengthen data security. The number of changing pixel rate (*NPCR*) and the unified averaged changed intensity (*UACI*) are values in [23] that show this characteristic. The Gauss map is also computationally faster. The equation for the Gauss map is as follows:

$$g_{i+1} = exp\left(-\alpha g_i^2\right) + \beta \tag{6}$$

To generate chaos, α = 6.2 and β = 0.5 have been chosen. The initial g value $g0$ has been chosen as depicted in Section 2.2. Figure 7a shows the bifurcation diagram of the Gauss map.

2.2.3. Piecewise Linear Map

The PWLCM is a 1D chaotic map. It is a uniform system with high ergodicity, invariant distribution, confusion, and determinacy suitable for information encryption [12,13]. The PWLCM equation is:

$$C(i+1) = \begin{cases} \frac{C(i)}{a} & if\ 0 \le C(i) < a \\ \frac{C(i)-a}{0.5-a} & if\ a \le C(i) < 0.5 \\ \frac{1-a-C(i)}{0.5-a} & if\ 0.5 \le C(i) < 1-a \\ \frac{1-C(i)}{a} & if\ 1-a \le C(i) < 1 \end{cases} \tag{7}$$

To generate chaos, a = 0.45 has been chosen. The initial C value C_0 has also been chosen, as depicted in Section 2.2. The bifurcation diagram of PWLCM is shown in Figure 7b.

2.2.4. Chirikov–Taylor Map

The Chirikov–Taylor map is a 2D map that represents the mapping of a square with the length of its side equal to 2π onto itself [27], given by:

$$\begin{gathered} C(i+1) = C(i) + T(i) + a\ sin\ (2\pi C(i)) \\ T(i+1) = C(i+1) - C(i) \end{gathered} \tag{8}$$

a = 0.6 and an initial T value of 0.2 have been chosen to generate chaos. The initial C value C_0 has been chosen, as depicted in Section 2.2. Figure 7d shows the bifurcation diagram of the Chirikov–Taylor map from 0 to 2.

2.2.5. Tent Map

The Tent map is a 1D chaotic map named after the tent-like shape of the $t(I + 1)$ vs. $t(i)$ graph. For the right values of μ (approaching 2), the map experiences transformations that cause it to stretch and fold. The map is also very susceptible to changes in its initial conditions for these values and displays an increase in the density of periodic and non-periodic points [24]. The equation for the tent map is given by:

$$\begin{aligned} T(i+1) &= \mu T(i) && if\ T(i) < 0.5 \\ T(i+1) &= \mu(1-(T(i)) && if\ T(i) \geq 0.5 \end{aligned} \tag{9}$$

To generate chaos, μ = 0.7 has been chosen. The initial T value T_0 has been chosen, as depicted in Section 2.2. Figure 7e shows the bifurcation diagram of the tent map.

2.2.6. Sine Map

The Sine map is a 1D chaotic map that shows high complexity and non-linearity. The equation of this system transforms inputs into the range [0, 1] [25] and is given by:

$$s(i+1) = a\ sin(\pi(s(i)) \tag{10}$$

The initial value $s0$ has been chosen as depicted in Section 2.2. Figure 7f shows the bifurcation diagrams of the Sine map from range 0 to 1.

2.2.7. Duffing Map

The Duffing map is a discretized version of the Duffing equation. It is a 2D system that displays either a periodic or chaotic property based on the parameter values [26]. The equation is given by:

$$\begin{gathered} M(i+1) = D(i) \\ D(i+1) = -b\ M(i) + a\ D(i) - D(i)^3 \end{gathered} \tag{11}$$

For generating chaos, a = 2.75, b = 0.2, and an initial D value of 0.3678 have been chosen. In addition, the initial M value M0 has been chosen, as depicted in Section 2.2. Figure 7g shows the Duffing map's bifurcation diagram.

2.2.8. Henon Map

Inspired by his study to show that dynamic systems defined by quadratic equations can be reduced to the study of area-preserved mapping, Henon proposed a 2D map:

$$\begin{gathered} E(i+1) = 1 - a(E(i)^2) + H(i) \\ H(i+1) = b(E(i)) \end{gathered} \tag{12}$$

a = 1.4, b = 0.3, and an initial H value of 0.3678 have been chosen for generating chaos. The choosing initial E value $E0$ is depicted in Section 2.2. Figure 7h shows the bifurcation diagram of the Henon map.

2.2.9. Circle Map

The Circle map is a 1D chaotic map for specific values of a (0, 1), and b; the map displays specific characteristics such as phase locking. For these values, the map behaves chaotically. The equation is as follows:

$$O(i+1) = O(i) + M - \frac{K}{2\pi} sin\ (2\pi O(i)) \tag{13}$$

K = 0.5 and M = 0.2 have been chosen to generate chaos. The initial O value $O0$ has been chosen as 0.489. Figure 7i shows the bifurcation diagram of the Circle map.

3. Proposed Hybrid Chaotic Map

The proposed hybrid chaotic map is a 3D map that combines three different chaotic maps non-linearly. The three maps used are the Chirikov–Taylor, Kaplan–Yorke, and Henon maps. The equation defining this system is given below.

$$x(i+1) = [K(x(i) + C(a, c\ sin(2\pi x(i)) - H(d,\ sin(2\pi x(i)\ \ ]mod\ 1 \tag{14}$$

where $C(a, b, X(i))$ is the Chirikov–Taylor map as a function of a and b (a and c are two of the parameters of the map), $K(x(i))$ is the Kaplan–Yorke function, and $H(d, x(i))$ is the Henon map as a function of d (d is the third parameter of the map). Here, mod refers to the modulus operation. The system equation thus becomes:

$$\begin{gathered} x(i+1) = \begin{bmatrix} 2x(i) + sin(2\pi x(i)) + y(i) + asin(2\pi sin(2\pi x(i))) \\ -1 + c(sin(2\pi x(i)))^2 - z(i) \end{bmatrix} mod\ 1 \\ y(i+1) = x(i+1) - sin(2\pi x(i)) \\ z(i+1) = d\ sin(2\pi x(i) + dx(i) + cos(4\pi x(i))) \end{gathered} \tag{15}$$

where a = 0.6, c = 1.4, and d = 0.3 have been chosen. The initial values $x0$, $y0$, and $z0$ have been chosen as shown in Section 3.2. The bifurcation diagram between x and parameters a, c, and d are given in Figure 8a–c, respectively. A uniform chaotic behavior is observed over a wide range of all the parameters. Hence, the proposed novel hybrid map is suitable for the final layer of scrambling in the proposed algorithm.

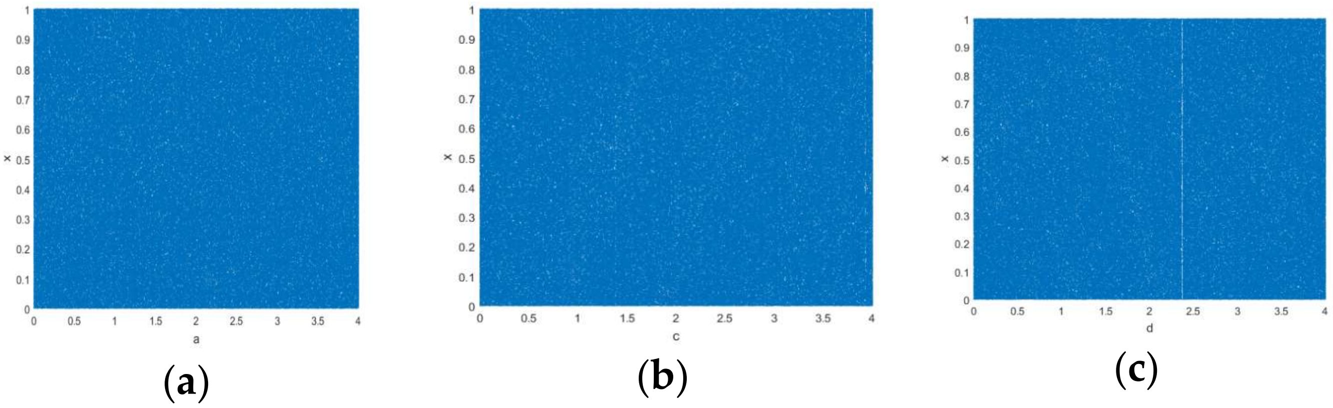


Figure 8. Bifurcation diagram of the proposed hybrid map (**a**) between x and a, (**b**) between x and c, and (**c**) between c and d.

3.1. Lyapunov Exponents for Proposed Map

Lyapunov exponents are a good measure of chaos in a system. For example, the Lyapunov exponent given by Equation (16) is used to study and analyze the rate of separation (expressed exponentially) of two orbits infinitesimally close at an initial time stage. It indicates a susceptibility or sensitivity to a variation in the initial conditions.

$$Ly(f(x)) = \lim_{n\to\infty}\left(\frac{1}{n}\right)\sum_{i=0}^{n-1} ln\left|\left(fi'(x)\right)\right| \tag{16}$$

where $fi'(x)$ is the derivative of the ith iterate $fi(x)$.

A positive Lyapunov exponent indicates an unstable orbit that shows chaotic properties. The points initially nearby will diverge regardless of how close they are. There will be a Lyapunov exponent for a 3D chaotic system, such as the proposed one, for every dimension, i.e., Lx, Ly, and Lz. A system can be declared chaotic if at least one of these dimensions displays a positive Lyapunov exponent over many iterations. Figure 9 illustrates how the Lyapunov exponents of the proposed hybrid map vary as iterations progress. Here, the 1st 500 iterations have been ignored to eschew the influence of the initial state. It is observed across 2000 iterations that Lx remains positive with a value of around 2. This implies that the proposed Hybrid Chaotic map is in a chaotic state.

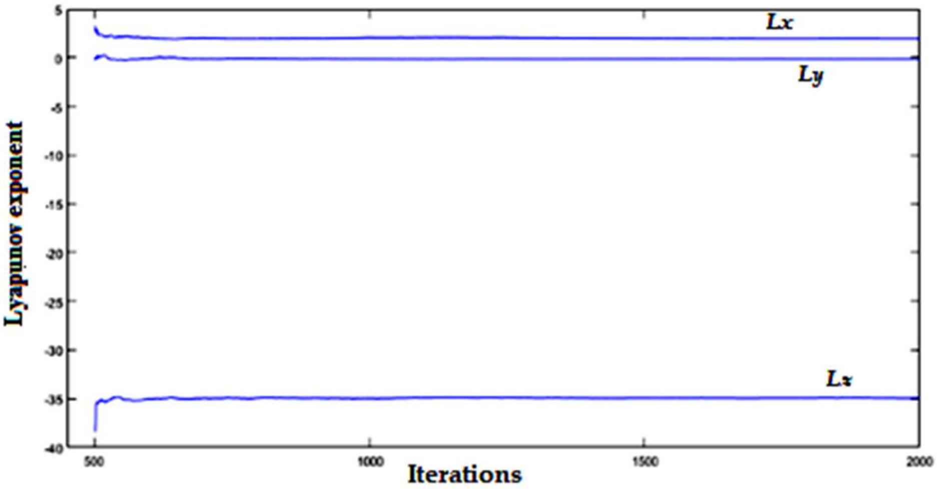


Figure 9. Lyapunov exponent spectrums of the proposed map.

All the chaotic maps' initial value x values are defined by XORing all the grey-scale image pixels and dividing by 255, as shown below.

$$x_o = \frac{x_o \oplus I(i,j)}{255} \tag{17}$$

where i and j vary from 1 to 256 each, i.e., covering all the image pixels.

3.2. Choosing Initial Value for Chaotic Maps

This prevents differential by changing the initial value, the encryption key, and the scrambling sequence whenever an original image is modified. For the proposed 3D hybrid chaotic map, the initial values of the other two dimensions, i.e., z and y, are determined in Equations (18) and (19), respectively.

$$z_o = \frac{1}{256\times 256\times 255}\left(\sum_{i=1}^{256}\sum_{j=1}^{256} I(i,j)\right) \tag{18}$$

$$y_0 = x_0 + z_0 \tag{19}$$

4. Proposed Methodology

The encryption and decryption algorithms for the proposed methodology are given below.

4.1. Encryption

The step-by-step process for image encryption is shown in Figure 10 and is explained in Algorithm 1.

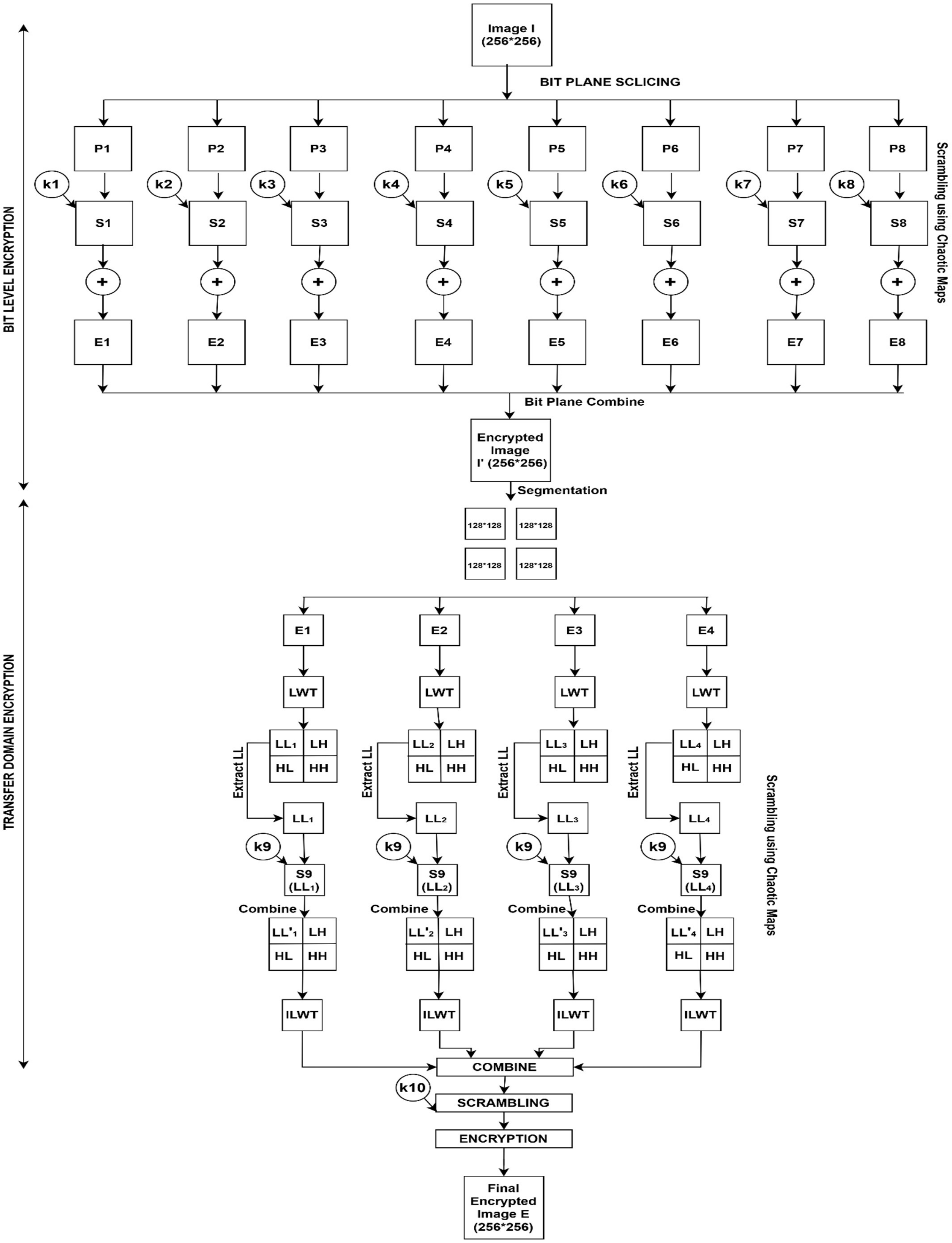


Figure 10. Block diagram of the proposed encryption method.

Algorithm 1 Encryption

Input: the 8-bit grayscale image of size 256 × 256
Output: the encrypted image of size 256 × 256
Step 1: Take an input image I and slice the image into constituent bit planes. For an 8-bit image, 8-bit planes will be formed.
Step 2: Determine the initial values of the chaotic maps from Section 3.2.
Step 3: For each sliced bit plane:

(a) Iterate the corresponding bit plane chaotic map equation with one using the modulo operation to form a 1D array of random/chaotic values of length equal to the number of pixels in each bit plane. For example, for a 256 × 256 image, the array's length will be 65,536.
(b) Sort the 1D array in ascending order.
(c) Scramble the 1D array by iterating through each bit plane pixel and assigning a new position to the pixel corresponding to the original index.
(d) Generate a key for the bit plane by carrying out a modulo operation between the original indices of the sorted array before scrambling and obtaining a binary key of length 65,536.
(e) Conduct a pixel-wise XOR operation between the generated key and the bit plane to give an encrypted bit plane.
(f) Repeat for the remaining planes with their corresponding chaotic maps.

Here, the Gauss map, PWLCM, Kaplan–Yorke map, Chirikov–Taylor map, Tent map, Sine map, Duffing map, and Henon map are used to scramble and encrypt planes 0 through 7, respectively.
Step 4: Recombine encrypted planes to form I'.
Step 5: Divide I' into four parts, $E1$, $E2$, $E3$, and $E4$, each of size 128 × 128.
Step 6: For each of the four sub-parts of size 128 × 128, perform the following operations:

(a) Perform forward LWT transform to obtain the four decomposed sub-bands LL, LH, HL, and HH.
(b) Take the LL sub-band and scramble it using the Circle map to form LL′.
(c) Perform the inverse IWT on LL′, LH, HL, and HH to obtain a scrambled sub-image of 128 × 128.
(d) Repeat step 6 for other subparts of the image I'.

Step 7: The 4 scrambled sub-images are recombined to give an R of size 256 × 256.
Step 8: The final layer of scrambling is done using the proposed Hybrid Chaotic map, whose initial value is found using equations in Section 3.2.
Step 9: Generate a key using step 3(d) using values in the range of 0–255 instead of in the range 0–1.
Step 10: The generated key from step 9 is XORed with R to give the final encrypted image E.

4.2. Decryption

The block diagram for the proposed decryption algorithm is given in Figure 11, and its step-by-step process is explained in Algorithm 2.

Algorithm 2 Decryption

Input: the encrypted image of size 256 × 256
Output: the 8-bit grayscale image of size 256 × 256
Step 1: Encrypted image E is XORed with the key used in the final encryption layer.
Step 2: Descramble the image using the same hybrid chaotic map. The pixels corresponding to the sorted generated chaotic array are placed in the positions corresponding to the indices of the sorted values in the generated unsorted 1D chaotic array to give E'.
Step 3: Divide E' into four parts, $E1'$, $E2'$, $E3'$ and $E4'$
Step 4: For each subpart $E1'$, $E2'$, $E3'$ and $E4'$:

(a) Perform forward LWT transform.
(b) Components corresponding to LL′ are descrambled similarly using the Circle map.
(c) The inverse LWT is performed for the part by replacing the LL′ component with the descrambled LL component.
(d) Repeat for remaining subparts.

Step 5: Recombine descrambled subparts to give a new D'.

Step 6: Slice new D' into constituent bit planes.

Step 7: For each bit plane:

(a) XOR with the same key with which they were encrypted.

(b) Descramble the image with the same chaotic map with which they were scrambled, as mentioned in step 2.

(c) Repeat step 7 for all other bit planes.

Step 8: Recombine the decrypted bit planes to recover the original image I.

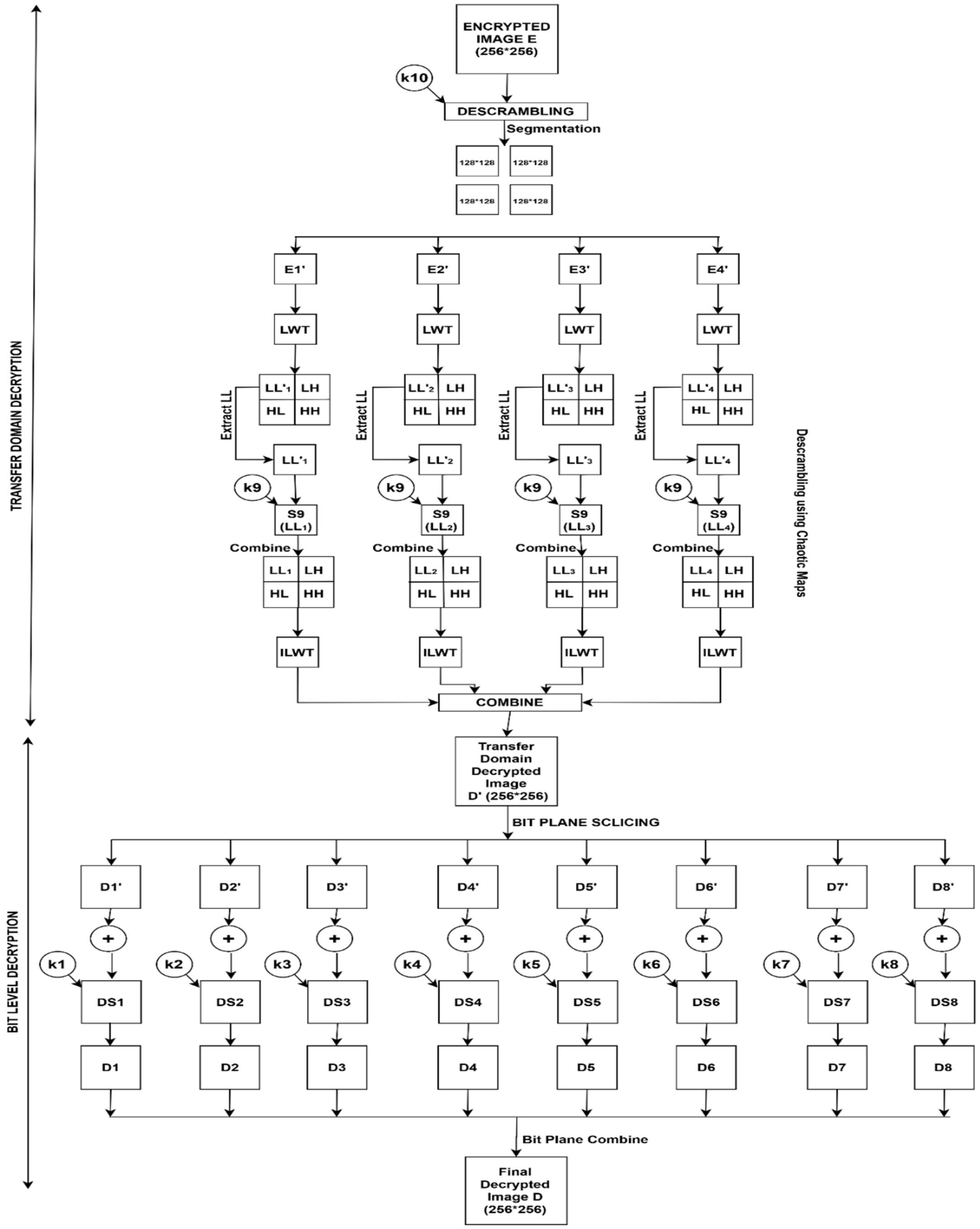

ENCRYPTED IMAGE E (256*256)
k10
DESCRAMBLING
Segmentation
128*128
E1'
E2'
E3'
E4'
LWT
LL'1 LH HL HH
Extract LL
LL'1
k9
S9 (LL1)
Combine
LL1 LH HL HH
ILWT
COMBINE
Descrambling using Chaotic Maps
TRANSFER DOMAIN DECRYPTION
Transfer Domain Decrypted Image D' (256*256)
BIT PLANE SCLICING
D1' D2' D3' D4' D5' D6' D7' D8'
+
k1 k2 k3 k4 k5 k6 k7 k8
DS1 DS2 DS3 DS4 DS5 DS6 DS7 DS8
D1 D2 D3 D4 D5 D6 D7 D8
Bit Plane Combine
Final Decrypted Image D (256*256)
BIT LEVEL DECRYPTION

Figure 11. Block diagram of the proposed decryption method.

5. Results and Performance Measures

The proposed algorithm is tested on ten grayscale images from the waterloo image database of sizes 256 × 256 (https://links.uwaterloo.ca/Repository.html (accessed on 10 December 2022)). Figure 12 shows the original, encrypted, and decrypted images of some input images from the database.

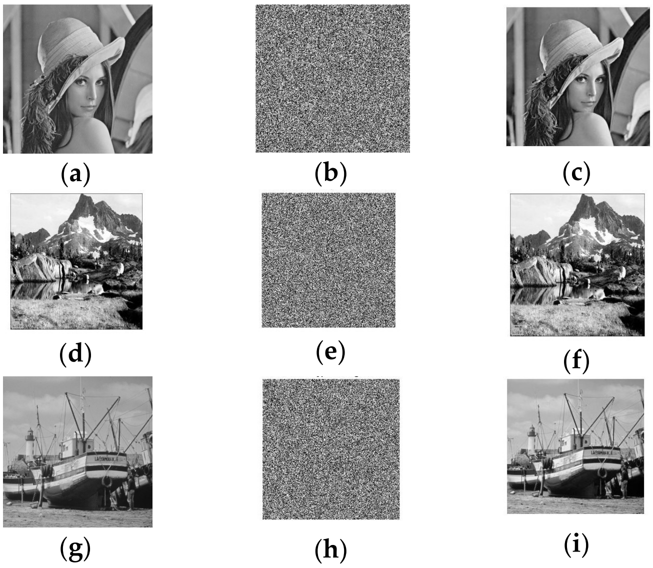


Figure 12. (**a**) Original Lena image, (**b**) Encrypted Lena image, (**c**) Decrypted Lena image, (**d**) Original mountain image, (**e**) Encrypted mountain image, (**f**) Decrypted mountain image, (**g**) Original boat image, (**h**) Encrypted boat image, and (**i**) Decrypted boat image.

5.1. Statistical Analysis

5.1.1. *MSE*, *PSNR*, and *SSIM*

The performance characteristics of the output encrypted image have been compared with the original image using the Peak Signal Noise Ratio (*PSNR*), Mean Square Error (*MSE*), and Structural Similarity Index Matrix (*SSIM*), which are shown below in Equations (20)–(22), where the sizes of the original (*I*) and the encrypted (*E*)images are $M \times N$.

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(I_{i,j} - E_{i,j}\right)^2 \tag{20}$$

$$PSNR = 10log_{10}\frac{255 * 255}{MSE}\text{dB} \tag{21}$$

$$SSIM(C,S) = \frac{(2\mu_I\mu_E + x_1) * (2\sigma_{IE} + x_2)}{\left(\mu_I^2 + \mu_E^2 + x_1\right) * \left(\sigma_I^2 + \sigma_E^2 + x_2\right)} \tag{22}$$

where:

μ_C is the mean pixel value of I; μ_S is the mean pixel value of E;

σ_C^2 is the variance of I; σ_S^2 is the variance of E;

σ_{CS} is the covariance of I and E;

$x_1 = (k_1L)^2$ and $x_2 = (k_2L)^2$ are used to compensate for the weak denominator; L is the dynamic range of pixel values;

k_1 = 0.01 and k_2 = 0.03.

A secure encrypted image should have a very low *PSNR* and *SSIM*. Table 1 shows the *PSNR* and *SSIM* of the encrypted images and the proposed method's original images. The average *PSNR* obtained is 8.8413 dB, and the *SSIM* for the ten images approaches zero, which proves that the encryption generated by the proposed method is robust and secure.

Table 1. *MSE*, *PSNR*, and *SSIM* values for the encrypted images.

S. No.	Image	*MSE*	*PSNR* (in dB)	*SSIM*
1.	Lena	7734.48	9.2465	0.0124
2.	Barb	7666.37	9.2849	0.0120
3.	Boat	8232.87	8.9753	0.0088
4.	Goldhill	8009.06	9.0950	0.0120
5.	Mandrill	6929.84	9.7236	0.0105
6.	Mountain	11,140.38	7.6618	0.0114
7.	Washsat	9365.16	8.4156	0.0094
8.	Peppers	9255.63	8.4667	0.0090
9.	Cameraman	9412.01	8.3940	0.0092
10.	Pirate	7907.29	9.1505	0.0110

5.1.2. Histogram Analysis

An image's histogram plots the number of pixels in an image having intensity values for every available intensity value. It is useful in evaluating image encryption schemes. A good image encryption system should convert an image into a non-discernible one with no defined features. This can be evaluated by checking the histogram of the encrypted image. An even histogram implies that the number of pixels for every intensity value is uniform, implying no discernible contrasts and features. Figure 13 shows the original, encrypted, and decrypted image histogram plots of the mandrill image for the proposed algorithm. Thus, the proposed algorithm can smoothen the original image histogram and protect the image's statistical information.

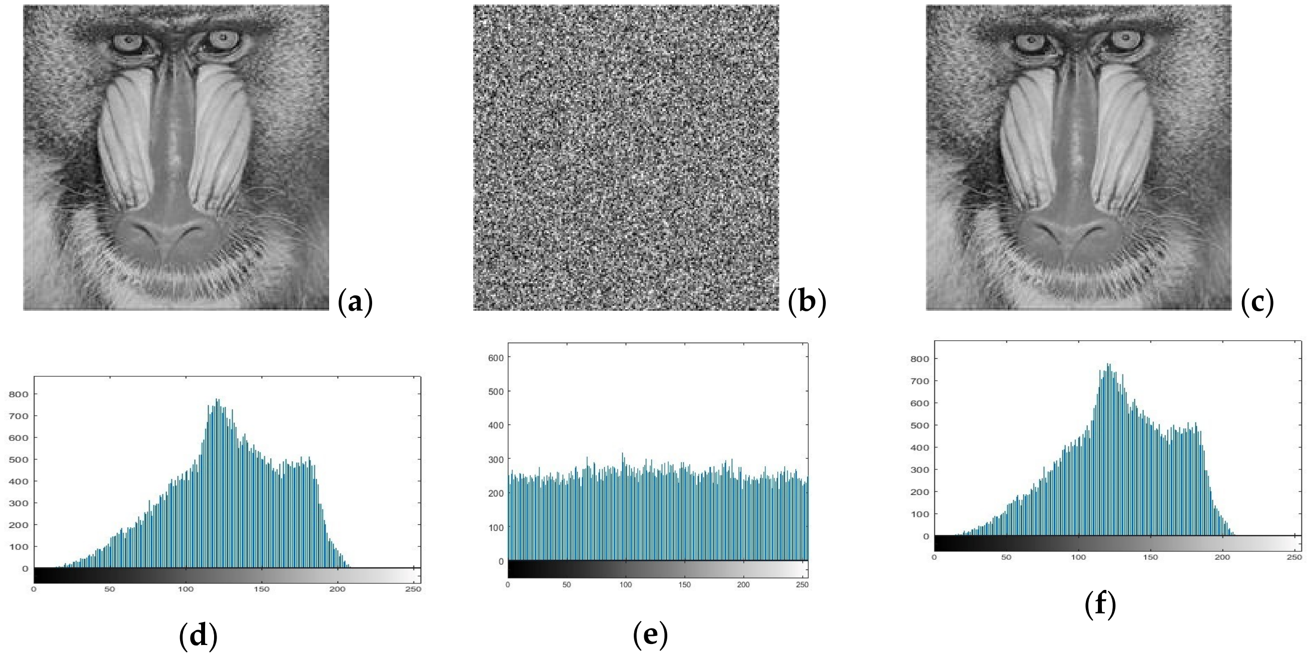


Figure 13. (**a**) Original mandrill image; (**b**) Encrypted mandrill image; (**c**) Decrypted mandrill image; (**d**) Histogram plot of (**a**); (**e**) Histogram plot of (**b**); (**f**) Histogram plot of (**c**).

5.1.3. Information Entropy

The information entropy (IFE) value gives information about the uncertainty of the pixels of an image. It illustrates the number of corresponding intensity levels that a pixel can adapt. It is given by Equation (23):

$$H = \sum_{n=0}^{255}\left(P(L=n)\log\frac{1}{P(L=n)}\right) \tag{23}$$

where n varies from 0 to 255 (the range of intensity values that grayscale pixels can take), and $P(L = n)$ is the percentage of pixels with intensity = n.

A high IFE score depicts an excellent randomness characteristic. For example, for a grayscale image with a data range of [0, 255], its maximum IFE is 8. An IFE score close to the maximum implies a highly random characteristic shown by the ciphered image.

The IFE values in Table 2 show that the proposed scheme provides an IFE close to 8. Thus, the randomness imparted by the scheme is strong enough to prevent the divulging of information. Table 3 compares the average IFE for the proposed method with the existing literature, showing promising results.

Table 2. Information entropy for encrypted images.

S. No.	Image	Entropy
1.	Lena	7.9976
2.	Barb	7.9974
3.	Boat	7.9970
4.	Goldhill	7.9970
5.	Mandrill	7.9970
6.	Mountain	7.9972
7.	Washsat	7.9971
8.	Peppers	7.9971
9.	Cameraman	7.9972
10.	Pirate	7.9973

Table 3. Comparison of average information entropy for encrypted images.

S. No.	Method	Entropy
1.	Proposed Method	7.9972
2.	[9]	7.9994
3.	[10]	7.9912
4.	[13]	7.9978
5.	[35]	7.998
6.	[40]	7.9993
7.	[41]	7.9998
8.	[42]	15.785
9.	[43]	7.9914
10.	[44]	7.9915
11.	[45]	7.9994
12.	[46]	7.9972
13.	[47]	7.9914
14.	[48]	7.9992
15.	[49]	7.9991
16.	[50]	7.9993
17.	[53]	7.9992

5.1.4. Correlation Test

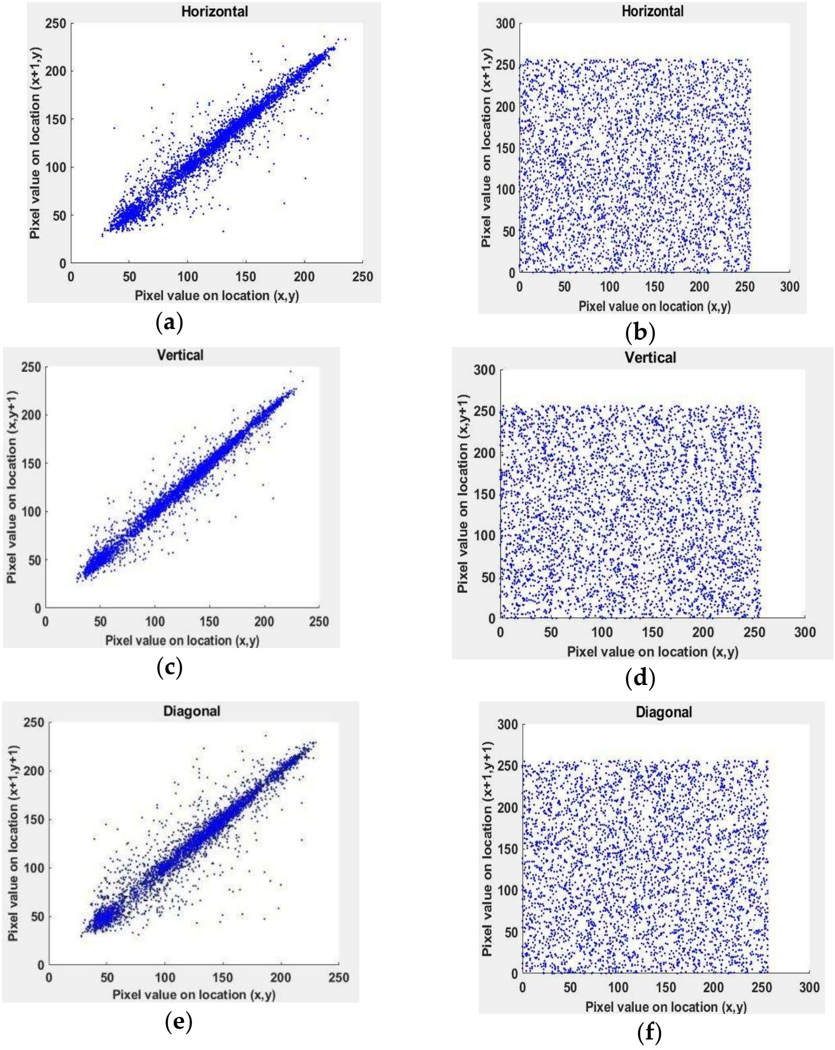
A good encryption scheme should preserve the statistical information of an image by reducing the correlation between the image's pixels [37–51]. The adjacent pixel correlation coefficients along the horizontal, vertical, and diagonal directions can be calculated using Equation (24), where μr and μc are the mean values along with the r (row) and c (column) coordinates of the ciphered image. σr and σc are the standard deviations, along with the r and c coordinates. E[.] is the expectation operation, and x and y are chosen appropriately depending on whether the horizontal, vertical, or diagonal correlation needs to be calculated. Table 4 compares the correlation coefficients of the proposed method and the three directions with other algorithms.

$$Correlation(x, y) = \frac{E[r - \mu_r][c - \mu_c]}{\sigma_r \sigma_c} \tag{24}$$

Table 4. Comparison of the correlation coefficients of various methods.

Method	**Image**	**Correlation Coefficients of the Literature**			**Correlation Coefficients of the Proposed Method**		
		Horizontal	**Vertical**	**Diagonal**	**Horizontal**	**Vertical**	**Diagonal**
[9]	Lena	−0.0685	0.0857	0.0059	−0.002153	−0.0000901	−0.0006059
	Goldhill	−0.0351	0.0556	0.0330	0.002091	0.001700	0.001184
[10]	Cameraman	0.0159	0.0093	0.0097	−0.001211	−0.006940	−0.0004557
[13]	Lena	0.0069	0.0047	0.0056	−0.002153	−0.0000901	−0.0006059
	Cameraman	0.0063	−0.0099	−0.0076	−0.001211	−0.006940	−0.0004557
	Baboon	−0.0063	0.0070	0.0051	0.005398	0.0002923	0.006103
	Boat	0.0033	−0.0069	0.0025	0.003136	−0.0005582	0.004891

The correlation distributions for neighboring adjacent pixels in all three directions for the original and encrypted Lena image are plotted in Figure 14. The dispersed correlation coefficients in the encrypted image plots compared to those of the original image demonstrates that the proposed scheme can significantly decrease the correlation between the pixels and prevent attackers from gleaning useful information where μ and σ are the mean value and standard deviation, respectively.



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

Figure 14. The correlation distribution of the Lena image. (**a**) Original image in the horizontal direction, (**b**) Encrypted image in the horizontal direction, (**c**) Original image in the vertical direction, (**d**) Encrypted image in the vertical direction, (**e**) Original image in a diagonal direction, and (**f**) Encrypted image in a diagonal direction.

5.2. Differential Attack Analysis

NPCR refers to the change rate in the pixel at every coordinate of the encrypted image when only one pixel at a randomly selected coordinate of the original image is altered. *UACI* measures the average difference between the intensities of the two encrypted images. The *NPCR* and *UACI* are defined in Equations (25)–(27).

$$NPCR = \left(\sum_{i=1}^{R} \sum_{j=1}^{C} Ed(i,j) \right) / R * C \tag{25}$$

$$UACI = \left(\sum_{i=1}^{R} \sum_{j=1}^{C} |E1(i,j) - E2(i,j)| \right) / 255 \times R \times C \tag{26}$$

$$Ed(i,j) = \begin{cases} 0, & if\ E1(i,j) = E2(i,j) \\ 1, & if\ E1(i,j) \neq E2(i,j) \end{cases} \tag{27}$$

where *R* and *C* are the dimensions of the images, *E*1 is the encrypted image of the original image, and *E*2 is the resultant image of the original image with a modification in one randomly selected pixel. The theoretical ideal value is 99.609% for *NPCR* and 33.46% for *UACI*.

Table 5 shows the *NPCR* and *UACI* values of the proposed algorithm and those of other algorithms. The values are more significant than the ideal values of *NPCR* and *UACI*; the proposed algorithm shows that it can withstand any differential attack [30–50].

Table 5. Comparison of average *NPCR* and *UACI* values.

S. No.	Method	*NPCR*	*UACI*
1.	Proposed Method	99.6230	33.4935
2.	[9]	99.6166	33.5033
3.	[10]	99.6110	33.4430
4.	[13]	99.6405	33.5175
5.	[30]	99.64	33.50
6.	[31]	99.62	33.44
7.	[32]	99.63	33.61
8.	[33]	99.61	33.47
9.	[34]	99.61	33.46
10.	[38]	99.6567	33.5078
11.	[39]	99.64	33.54
12.	[40]	99.61	33.50
13.	[41]	99.6060	33.5126
14.	[42]	99.6067	33.47
15.	[43]	99.6366	33.4586
16.	[44]	99.62	33.47
17.	[45]	99.6094	33.4635
18.	[46]	99.6056	33.4758
19.	[47]	99.6060	33.4689
20.	[48]	99.6132	33.4601
21.	[50]	99.6199	33.4773

5.3. Key Space Analysis and Key Sensitivity

An efficient encryption method should be susceptible to minute variations in the key. It should not be decrypted by a key that might even differ from the actual key. The smallest changes in the decryption key should lead to a significant alteration in the output. The key generated by the proposed chaotic map is directly dependent on parameters *a*, *c* and *d*. It has been observed that a change as small as 10–16 for *a* and 10–15 for *b* and *c* show a significantly different output. Hence, the proposed system is susceptible to minuscule variations in the key generated by the proposed hybrid map. Good encryption should have a large keyspace. Such a system would be able to withstand third-party manipulations such as brute force attacks satisfactorily. The keyspace for encryption due to the proposed hybrid map in Section 3 is affected by the parameters a, c, and d and the initial values $x0$, $b0$, and $p0$ (initial values for each chaotic map dimension). The precision for parameter a has been observed to be 10–16, while for parameters c, d, and initial values $x0$, $b0$, and $p0$, the precisions have been observed to be 10–15. Thus, the keyspace is calculated to be 1016 × 1075 × 256 = 1.044 × 2184. The keyspace is hence large enough to render a brute-force attack unfeasible.

5.4. Robustness

5.4.1. Cropping Attack

During transmission, data loss is expected, due to which images can experience cropping of portions. Hence, the encryption scheme should be able to withstand such an attack. To check this resistance to cropping, a 70 × 70 portion of the encrypted image is cropped, and then the decryption algorithm is applied to this modified image. It is observed that the resulting decrypted image still retains the necessary information despite the missing data in the encrypted image. Hence the scheme shows good resistance to such data loss, as depicted in Figure 15.

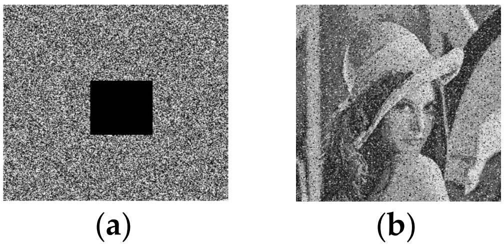


Figure 15. (**a**) 70 × 70 area cropped in the encrypted Lena image; (**b**) Decrypted image.

5.4.2. Noise Attack

It is to be expected that images will be subjected to noise. Hence, the encryption scheme should withstand such an attack and retain most of the image's essential features on the receiving end after decryption. The proposed algorithm has tested this by applying varying Gaussian noise scales to the encrypted image. The noise has been applied according to:

$$E' = E + \sigma G_n \tag{28}$$

where E is the encrypted image, G_n is the Gaussian noise (matrix, generated from a 0 mean and unit SD normal distribution, of dimensions equal to that of the image), σ is the noise intensity (the analysis has been done for σ values of 0.3, 0.6 and 1), and E' is the noisy encrypted image.

Figure 16 shows images decrypted from the noisy encrypted images for the three σ values. The decrypted images retain most of the important aspects and features of the original image, demonstrating that the proposed scheme shows good resistance to noise attacks.

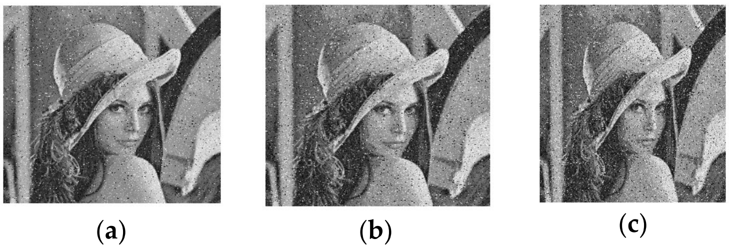


Figure 16. Decrypted image when Gaussian noise is applied to the image with σ values of (**a**) 0.3; (**b**) 0.6, and (**c**) 1.

5.5. Computational Complexity

The speed with which an image can be encrypted must also be considered. The speed depends on factors such as the computer specifications on which the algorithm is run. The average time required for the entire embedding and extraction combined when tested on a Windows 10 operating system with a 64-bit Intel(R) Core(TM) i5-7200U CPU at 2.5 GHz and 8 GB RAM is found to be 2.1743 s. The same time complexity can be given in Big-O notation as O(N) since the algorithm is operated sequentially for each pixel. This algorithm is compared against the encryption scheme proposed in [54], which takes 23.7 s to encrypt the image. This shows that the proposed encryption scheme is effective in terms of time consumption.

5.6. NIST SP 800-22

NIST Special Publication (SP) 800-22 is a statistical test suite for random and pseudorandom number generators (RNGs and PRNGs) used in cryptographic applications. The test suite consists of 22 statistical tests designed to provide a comprehensive evaluation of the statistical properties of an RNG or PRNG. The test suite is organized into three categories: empirical tests, which are based on the observed statistical properties of the output sequence; spectral tests, which analyze the distribution of the output sequence in the frequency domain; and statistical tests, which analyze the distribution of the output sequence in the time domain. The test suite is intended to be used as part of an overall testing strategy for RNGs and PRNGs and is not intended to be used as a standalone evaluation tool. In summary, NIST SP 800-22 is a comprehensive test suite for evaluating the statistical properties of RNGs and PRNGs used in cryptographic applications. The probability (P)-value, calculated for each of the 15 NIST Tests, varies between 0 and 1 and is given in Table 6. Shallow p-values indicate the absence of randomness in the stream of bits. The NIST SP 800-22 standard observes that if the p-value is greater than or equal to 0.001, the bits under testing can be considered to have uniform distribution and statistical strength.

Table 6. NIST SP 800-22 results.

Type of Test	p-Value	Conclusion
Frequency Test (Monobit)	0.4120	Random
Frequency Test within a Block	0.9606	Random
Run Test	0.6140	Random
Longest Run of Ones in a Block	0.1815	Random
Binary Matrix Rank Test	0.1327	Random
Discrete Fourier Transform (Spectral) Test	0.4770	Random
Non-Overlapping Template Matching Test	0.0257	Random
Overlapping Template Matching Test	0.1384	Random
Maurer's Universal Statistical test	0.8626	Random
Linear Complexity Test	0.4227	Random
Serial Test 1	0.9000	Random
Serial Test 2	0.7167	Random
Approximate Entropy Test	0.7484	Random
Cumulative Sums (Forward) Test	0.6990	Random
Cumulative Sums (Reverse) Test	0.5227	Random

6. Discussion of the Obtained Results

The proposed work has been compared with the existing works regarding entropy and differential analysis. Entropy has been considered a strong metric for evaluating the equidistribution property of encrypted pixels among the plane. When comparing, the proposed work yielded a maximum entropy of 7.9972, which is relatively higher than the works reported in [10,43,44,46,47] and comparatively similar to the works reported in [9,13,35,40,41,45,48–50,53]. From the results, it has been confirmed that the proposed work has secured adequate entropy. In addition, differential attack analysis was carried out to investigate the tolerance level of the proposed cryptosystem when it is subjected to cipher attacks. On observing the results from Table 5, this work outperforms the works mentioned in [9,10,33,34,40–42,45–48,50] in *NPCR* estimation. Similarly, the *UACI* estimated value is higher than the works mentioned in [9,10,13,31,33,34,42–48,50]. Other than this analysis, the avalanche effect of keys has been verified by conducting a key sensitivity analysis, thereby calculating the keyspace of 1.044×2^{184}, which is very much sufficient to resist a brute force attack. Further, cropping and correlation analyses were performed to evaluate the cryptosystem from the position of being attacked. Though the encrypted images were attacked through the earlier attacks, the algorithm was resistant to decrypting the image, evidencing the proposed algorithm's self-strength. The novelty and enhancements of the proposed work are presented below:

1. A hybrid chaotic system-driven cryptosystem was developed;
2. Lifting wavelet transform decomposition was adopted to achieve quantization error-free frequency separation;
3. Bit plane-based diffusion was used to break the pixel dependency in an effective way which resulted in a near-zero correlation on encrypted pixels;
4. Ten different keys were used to accomplish the encryption through which the keyspace was increased to 1.044×2^{184};
5. Segmentation of the intermediate cipher image helped to reduce the time consumption, which resulted in a time of 2.1743 s to encrypt a 256 × 256 × 8-bit image;
6. A maximum entropy of 7.9972 was achieved with an average *PSNR* of 8.84139.

7. Conclusions

Effective multilayer encryption augmented by a novel 3D hybrid chaotic map is proposed. The proposed map has a uniform chaotic characteristic over several parameters and a larger keyspace than existing chaotic maps. Multiple chaotic maps are used to scramble and encrypt, creating a robust encryption system in which each key generated by a chaotic map is different from the others. LWT is used in a portion of our procedure to perform frequency domain scrambling to overcome certain spatial domain scrambling limitations. Ten standard 256 × 256 × 8-bit depth grayscale test images were taken from the University of Southern California database to test the proposed encryption. Entropy and correlation analyses were considered as the primary focus for statistical validation, in which the average entropy was calculated as 7.9972 with a near 0 correlation. Further, an average *PSNR* of 8.8439 ensured that adequate encryption was achieved through the proposed cryptosystem. The avalanche effect was witnessed through key sensitivity analysis, and the average *NPCR* and *UACI* from differential analyses were determined as 99.6230 and 33.4935, respectively. This evidences that the proposed work is cryptographically strong. Moreover, the attack analysis results also confirm this algorithm's attack-resistant capability. Future work will be on implementing the chaos-cryptic solution on reconfigurable hardware to improve the speed of the operation by employing hardware concurrency.

Author Contributions: Conceptualization, H.M. and R.A.; methodology, T.V.; software, A.R.M.; validation, S.S. and A.R.M.; formal analysis, S.S.; investigation, A.R.M.; resources, S.S.; data curation, A.R.M. and T.V.; writing—original draft preparation, R.A. and T.V.; writing—review and editing, H.M. and R.A. visualization, A.R.M.; supervision, R.A. and T.V.; project administration, H.M. and R.A. funding acquisition, H.M. and R.A. All authors have read and agreed to the published version of the manuscript.

Funding: Institutional Fund Projects under grant no. (IFPIP-1113-144-1443). Ministry of Education and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

Data Availability Statement: Not applicable.

Acknowledgments: The authors gratefully acknowledge the technical and financial support provided by the Ministry of Education and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khan, M.; Shah, T. A Literature Review on Image Encryption Techniques. *3D Res.* **2014**, *5*, 29. [CrossRef]
2. Kaur, M.; Kumar, V. A Comprehensive Review on Image Encryption Techniques. *Arch. Comput. Methods Eng.* **2018**, *27*, 15–43. [CrossRef]
3. Jung, H. Data hiding scheme improving embedding capacity using mixed PVD and LSB on bit plane. *J. Real-Time Image Process.* **2018**, *14*, 127–136. [CrossRef]
4. Huang, H.-Y.; Chang, S.-H. A Lossless Data Hiding based on Discrete Haar Wavelet Transform. In Proceedings of the IEEE International Conference on Computer and Information Technology, Bradford, UK, 29 June–1 July 2010; pp. 1554–1559.
5. Tedmori, S.; Al-Najdawi, N. Image cryptographic algorithm based on the Haar wavelet transform. *Inf. Sci.* **2014**, *269*, 21–34. [CrossRef]
6. Li, C.; Lin, D.; Lu, J. Cryptanalyzing an Image-Scrambling Encryption Algorithm of Pixel Bits. *IEEE MultiMedia* **2017**, *24*, 64–71. [CrossRef]
7. Hua, Z.; Zhou, B.; Zhou, Y. Sine-Transform-Based Chaotic System with FPGA Implementation. *IEEE Trans. Ind. Electron.* **2018**, *65*, 2557–2566. [CrossRef]
8. Suneja, K.; Dua, S.; Dua, M. A Review of Chaos based Image Encryption. In Proceedings of the International Conference on Computing Method-Ologies and Communication (ICCMC), Erode, India, 27–29 March 2019; pp. 693–698.
9. Hua, Z.; Zhou, Y. One-Dimensional Nonlinear Model for Producing Chaos. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2017**, *65*, 235–246. [CrossRef]
10. Gao, S.; Wu, R.; Wang, X.; Wang, J.; Li, Q.; Wang, C.; Tang, X. A 3D model encryption scheme based on a cascaded chaotic system. *Signal Process.* **2023**, *202*, 108745. [CrossRef]
11. Zhu, H.; Zhao, Y.; Song, Y. 2D Logistic-Modulated-Sine-Coupling-Logistic Chaotic Map for Image Encryption. *IEEE Access* **2019**, *7*, 14081–14098. [CrossRef]
12. Tang, Z.; Yang, Y.; Xu, S.; Yu, C.; Zhang, X. Image Encryption with Double Spiral Scans and Chaotic Maps. *J. Secur. Commun. Netw.* **2019**, *2019*, 8694678. [CrossRef]
13. Ye, G.; Huang, X. A secure image encryption algorithm based on chaotic maps and SHA-3. *J. Secur. Commun. Netw.* **2016**, *9*, 2015–2023. [CrossRef]
14. Agarwal, S. A Review of Image Scrambling Technique Using Chaotic Maps. *Int. J. Eng. Technol. Innov.* **2018**, *8*, 77–98.
15. Sankpal, P.R.; Vijaya, P.A. Image Encryption Using Chaotic Maps: A Survey. In Proceedings of the 2014 Fifth International Conference on Signals and Image, Cherbourg, France, 20 June–2 July 2014; IEEE Conference Publication: Cherbourg, France, 2014.
16. Abdullah, H.N.; Abdullah, H.A. Image encryption using hybrid chaotic map. In Proceedings of the 2017 International Conference on Current Research in Computer Science and Information Technology (ICCIT), Dhaka, Bangladesh, 22–24 December 2017; pp. 121–125.
17. Banu, S.A.; Amirtharajan, R. A robust medical image encryption in dual domain: Chaos-DNA-IWT combined approach. *Med. Biol. Eng. Comput.* **2020**, *58*, 1445–1458. [CrossRef] [PubMed]
18. Rajini, G.K. A Comprehensive Review on Wavelet Transform and Its Applications. 2016. Available online: http://www.arpnjournals.org/jeas/research_papers/rp_2016/jeas_1016_5133.pdf (accessed on 10 December 2022).
19. Uytterhoeven, G.; Roose, D.; Bultheel, A. Wavelet Transforms Using the Lifting Scheme. ITA-Wavelets Report WP, 1. 1997. Available online: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=e154fe1bf4777fcad4bc70a7e01611e7a5c43e9c (accessed on 10 December 2022).
20. Luo, Y.; Wang, F.; Xu, S.; Zhang, S.; Li, L.; Su, M.; Liu, J. CONCEAL: A robust dual-color image watermarking scheme. *Expert Syst. Appl.* **2022**, *208*, 118133. [CrossRef]
21. Singh, S.P.; Bhatnagar, G. A simplified watermarking algorithm based on lifting wavelet transform. *Multimed. Tools Appl.* **2019**, *78*, 20765–20786. [CrossRef]
22. Jan, A.; Parah, S.A.; Hussan, M.; Malik, B.A. Double layer security using crypto-stego techniques: A comprehensive review. *Health Technol.* **2021**, *12*, 9–31. [CrossRef]
23. Salunke, S.; Ahuja, B.; Hashmi, M.F.; Marriboyina, V.; Bokde, N.D. 5D Gauss Map Perspective to Image Encryption with Transfer Learning Validation. *Appl. Sci.* **2022**, *12*, 5321. [CrossRef]
24. Li, C.; Luo, G.; Qin, K.; Li, C. An image encryption scheme based on chaotic tent map. *Nonlinear Dyn.* **2016**, *87*, 127–133. [CrossRef]
25. Hua, Z.; Zhou, Y. Image encryption using 2D Logistic-adjusted-Sine map. *Inf. Sci.* **2016**, *339*, 237–253. [CrossRef]
26. Khan, M.; Masood, F. A novel chaotic image encryption technique based on multiple discrete dynamical maps. *Multimed. Tools Appl.* **2019**, *78*, 26203–26222. [CrossRef]
27. Rakheja, P.; Yadav, S.; Tobria, A. A novel image encryption mechanism based on umbrella map and Yang-Gu algorithm. *Optik* **2022**, *271*, 170152. [CrossRef]
28. Banu, S.A.; Amirtharajan, R. Tri-level scrambling and enhanced diffusion for DICOM image cipher-DNA and chaotic fused approach. *Multimed. Tools Appl.* **2020**, *79*, 28807–28824. [CrossRef]
29. Banu, S.A.; Amirtharajan, R. Bio-inspired cryptosystem on the reciprocal domain: DNA strands mutate to secure health data. *Front. Inf. Technol. Electron. Eng.* **2021**, *22*, 940–956. [CrossRef]
30. Chen, Y.; Tang, C.; Ye, R. Cryptanalysis and improvement of medical image encryption using high-speed scrambling and pixel adaptive diffusion. *Signal Process.* **2019**, *167*, 107286. [CrossRef]

31. Guan, M.; Yang, X.; Hu, W. Chaotic image encryption algorithm using frequency-domain DNA encoding. *IET Image Process.* **2019**, *13*, 1535–1539. [CrossRef]
32. Belazi, A.; Talha, M.; Kharbech, S.; Xiang, W. Novel Medical Image Encryption Scheme Based on Chaos and DNA Encoding. *IEEE Access* **2019**, *7*, 36667–36681. [CrossRef]
33. Haghighi, B.B.; Taherinia, A.H.; Mohajerzadeh, A.H. TRLG: Fragile blind quad watermarking for image tamper detection and recovery by providing compact digests with optimized quality using LWT and GA. *Inf. Sci.* **2019**, *486*, 204–230. [CrossRef]
34. Stalin, S.; Maheshwary, P.; Shukla, P.K.; Maheshwari, M.; Gour, B.; Khare, A. Fast and Secure Medical Image Encryption Based on Non Linear 4D Logistic Map and DNA Sequences (NL4DLM_DNA). *J. Med. Syst.* **2019**, *43*, 267. [CrossRef]
35. Liu, T.; Banerjee, S.; Yan, H.; Mou, J. Dynamical analysis of the improper fractional-order 2D-SCLMM and its DSP implementation. *Eur. Phys. J. Plus* **2021**, *136*, 506. [CrossRef]
36. Liu, T.; Yan, H.; Banerjee, S.; Mou, J. A fractional-order chaotic system with hidden attractor and self-excited attractor and its DSP implementation. *Chaos Solitons Fractals* **2021**, *145*, 110791. [CrossRef]
37. Kaur, G.; Agarwal, R.; Patidar, V. Color image encryption system using combination of robust chaos and chaotic order fractional Hartley transformation. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 5883–5897. [CrossRef]
38. Yang, F.; Mou, J.; Ma, C.; Cao, Y. Dynamic analysis of an improper fractional-order laser chaotic system and its image en-cryption application. *Opt. Lasers Eng.* **2020**, *129*, 106031. [CrossRef]
39. Zefreh, E.Z. An image encryption scheme based on a hybrid model of DNA computing, chaotic systems and hash functions. *Multimed. Tools Appl.* **2020**, *79*, 24993–25022. [CrossRef]
40. Zhang, X.; Hu, Y. Multiple-image encryption algorithm based on the 3D scrambling model and dynamic DNA coding. *Opt. Laser Technol.* **2021**, *141*, 107073. [CrossRef]
41. Ravichandran, D.; Banu, S.A.; Murthy, B.; Balasubramanian, V.; Fathima, S.; Amirtharajan, R. An efficient medical image encryption using hybrid DNA computing and chaos in transform domain. *Med. Biol. Eng. Comput.* **2021**, *59*, 589–605. [CrossRef] [PubMed]
42. Patel, S.; Thanikaiselvan, V.; Pelusi, D.; Nagaraj, B.; Arunkumar, R.; Amirtharajan, R. Colour image encryption based on customized neural network and DNA encoding. *Neural Comput. Appl.* **2021**, *33*, 14533–14550. [CrossRef]
43. Zheng, J.; Liu, L. Novel image encryption by combining dynamic DNA sequence encryption and the improved 2D logistic sine map. *IET Image Process.* **2020**, *14*, 2310–2320. [CrossRef]
44. Li, J.; Chen, L.; Cai, W.; Xiao, J.; Zhu, J.; Hu, Y.; Wen, K. Holographic encryption algorithm based on bit-plane decomposition and hyperchaotic Lorenz system. *Opt. Laser Technol.* **2022**, *152*, 108127. [CrossRef]
45. Zhang, F.; Zhang, X.; Cao, M.; Ma, F.; Li, Z. Characteristic Analysis of 2D Lag-Complex Logistic Map and Its Application in Image Encryption. *IEEE MultiMedia* **2021**, *28*, 96–106. [CrossRef]
46. Lone, M.A.; Qureshi, S. RGB image encryption based on symmetric keys using Arnold transform, 3D chaotic map and affine hill cipher. *Optik* **2022**, *260*, 168880. [CrossRef]
47. Teng, L.; Wang, X.; Xian, Y. Image encryption algorithm based on a 2D-CLSS hyperchaotic map using simultaneous permutation and diffusion. *Inf. Sci.* **2022**, *605*, 71–85. [CrossRef]
48. Ye, G.; Wu, H.; Liu, M.; Shi, Y. Image encryption scheme based on blind signature and an improved Lorenz system. *Expert Syst. Appl.* **2022**, *205*, 117709. [CrossRef]
49. Sridevi, A.; Sivaraman, R.; Balasubramaniam, V.; Sreenithi; Siva, J.; Thanikaiselvan, V.; Rengarajan, A. On Chaos based duo confusion duo diffusion for colour images. *Multimed. Tools Appl.* **2022**, *81*, 16987–17014. [CrossRef]
50. Zhang, Y.; Xie, H.; Sun, J.; Zhang, H. An efficient multi-level encryption scheme for stereoscopic medical images based on coupled chaotic system and Otsu threshold segmentation. *Comput. Biol. Med.* **2022**, *146*, 105542. [CrossRef] [PubMed]
51. Tutueva, A.; Nepomuceno, E.G.; Moysis, L.; Volos, C.; Butusov, D. Adaptive Chaotic Maps in Cryptography Applications. In *Cybersecurity. Studies in Big Data*; Abd El-Latif, A.A., Volos, C., Eds.; Springer: Cham, Switzerland, 2022; Volume 102. [CrossRef]
52. Tutueva, A.V.; Moysis, L.; Rybin, V.G.; Kopets, E.E.; Volos, C.; Butusov, D.N. Fast synchronization of symmetric Hénon maps using adaptive symmetry control. *Chaos Solitons Fractals* **2022**, *155*, 111732. [CrossRef]
53. Anushiadevi, R.; Amirtharajan, R. Separable reversible data hiding in an encrypted image using the adjacency pixel difference histogram. *J. Inf. Secur. Appl.* **2023**, *72*, 103407. [CrossRef]
54. Zhang, X.; Wang, X. Multiple-image encryption algorithm based on DNA encoding and chaotic system. *Multimed. Tools Appl.* **2018**, *78*, 7841–7869. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.