HDIEA: high dimensional color image encryption architecture using five-dimensional Gauss-logistic and Lorenz system

Bharti Ahuja, Rajesh Doriya, Sharad Salunke, Mohammad Farukh Hashmi, Aditya Gupta & Neeraj Dhanraj Bokde

To cite this article: Bharti Ahuja, Rajesh Doriya, Sharad Salunke, Mohammad Farukh Hashmi, Aditya Gupta & Neeraj Dhanraj Bokde (2023) HDIEA: high dimensional color image encryption architecture using five-dimensional Gauss-logistic and Lorenz system, Connection Science, 35:1, 2175792, DOI: 10.1080/09540091.2023.2175792

To link to this article: https://doi.org/10.1080/09540091.2023.2175792

CONNECTION SCIENCE
2023, VOL. 35, NO. 1, 2175792
https://doi.org/10.1080/09540091.2023.2175792

RESEARCH ARTICLE

OPEN ACCESS Check for updates

HDIEA: high dimensional color image encryption architecture using five-dimensional Gauss-logistic and Lorenz system

Bharti Ahuja[a], Rajesh Doriya[a], Sharad Salunke[b], Mohammad Farukh Hashmi[c], Aditya Gupta[d] and Neeraj Dhanraj Bokde[e]

[a]Department of Information Technology, National Institute of Technology Raipur, Raipur, India; [b]Department of Electronics and Communication Engineering, Amity University Madhya Pradesh, Gwalior, India; [c]Department of Electronics and Communication Engineering, NIT Warangal, Warangal, India; [d]Department of Information and Technology, University of Agder, Kristiansand, Norway; [e]Center for Quantitative Genetics and Genomics, Aarhus University, Aarhus, Denmark

ABSTRACT

The work presented here is a high dimensional color image encryption architecture (HDIEA) founded on the Lorenz-Gauss-Logistic (LGL) encryption algorithm. The primary objective is to demonstrate that both the proposed novel five-dimensional (5D) Gauss-Logistic and four-dimensional (4D) Lorenz system are operating in a hyper-chaotic condition. The visual study of their most important characteristics, such as the sensitivity of the starting value of both maps and the Lyapunov exponent of the 5D Gauss Logistic map, is carried out. The Runge–Kutta technique is used to discretise the Lorenz system in order to construct a pseudo-random sequence generation for the control parameter that has a greater degree of randomness. The 5D Gauss-Logistic system is then selected to serve as the principal hyper-chaotic mapping scheme. The simulation results demonstrate that the suggested image encryption method is successful according to the NIST test and has powerful anti-attack, a larger key space as large as 2^{847}, which is prone to multiple attacks, and key sensitivity capabilities. Also, the pixel correlation reached -0.0019, -0.0016, and -0.0069, while the information entropy was at 7.9996. This demonstrates the excellent scrambling effect of the proposed approach, which is capable of greatly improving the color image security performance.

ARTICLE HISTORY
Received 21 October 2022
Accepted 28 January 2023

KEYWORDS
Image encryption; Gauss map; logistic map; Lorenz system; hyperchaotic map; Lyapunov exponent

1. Introduction

More than one trillion photographs were taken in the year 2020. This is despite the fact that the COVID-19 virus disrupted important formal and informal occasions. As of 2022, the number of users of Instagram, a social network that focuses on the sharing of images, has surpassed 2 billion. Banking, academia, health research, aviation, the defense and even politics are just some of the numerous fields that are making ex use of digital images. When we want to communicate visual information with one another, we may quickly transfer it

CONTACT Aditya Gupta ✉ aditya.gupta@uia.no Department of Information and Technology, University of Agder, 4604 Kristiansand, Norway

over the internet using either a computer or mobile device. However, unauthorised individuals may readily get the images as well, which poses a significant risk to the information exchange of images (Lin & Li, 2021). Unauthorised cryptanalysis also poses a risk to the security of the images themselves. More crucially, certain images may contain issues pertaining to national defense and violates individual's right to privacy. For example, satellite surveillance and bio-metric identification both fall within this category. As a result of this, the subject of how to effectively secure the digital images during transmission has drawn a significant amount of interest from academics and industry professionals all over the world (Ferdush et al., 2021).

Image encryption utilises a far bigger quantity of data and a significantly higher level of redundancy when compared to the conventional approach of encrypting text. The starting circumstances in terms of constant numbers and distinct parameters of the chaotic system have a major impact on the system's sensitivity as well as its dependency on those values. As a result, numerous different chaos based approaches for image security have been developed, one after the other (Wang et al., 2022), in an effort to make the internet a more secure place. Image encryption based on chaos has gained popularity in past few years due to its many advantages in cryptography, including ergodicity, unpredictable nature, pseudo-randomness, and highly sensitive to variables and initial condition.

Fridrich was the first person to suggest applying a scrambling-diffusion structure based on the concept of chaos (Fridrich, 1998). Except chaos, there are also other approaches, such as those that are based on block scrambling, bit-level scrambling, the DNA rule, matrix manipulation, and tensor theory (Hosny et al., 2021; Shahna & Mohamed, 2021; Wang et al., 2022; Wang & Gao, 2020). Shahna et al. (Shahna & Mohamed, 2021) suggested a method using double scrambling for image, which used dual scrambling on bit level and pixel level as well, to provide a greater level of security during the permutation process. Matrix semi-tensor product technique was developed by Wang et al. (Wang & Gao, 2020) as a means of diffusing a image in order to obtain an encrypted image.

Pixel-level approaches execute encryption and decryption on pixels at the permutation stage, i.e. an image is viewed as a collection of pixels. Different permutation methods are utilised by different authors in the earlier literatures. Sorting algorithms, cyclic shift, pixel switching mechanisms, and parallel permutation are examples (Ahmad & Hwang, 2016; Fu et al., 2017; Kaur & Singh, 2021; Song et al., 2020; Song et al., 2022a; Song et al., 2023; Wang et al., 2019). However, the pixel value and histogram statistics stay unaltered throughout the sorting method, cyclic shift, and pixel swapping procedures (Chen et al., 2021). But parallel permutation helps in reducing encryption time.

Many people started looking at high-dimensional (HD) chaotic attractors after the invention of chaos theory. These HD chaotic attractors include systems like 4D chaotic attractor subsystems (Liu et al., 2019b; Yan et al., 2023) and 5D chaotic attractor subsystems (Koyuncu et al., 2019). Fractional-order chaotic systems (Liu et al., 2019a) and hidden attractors (Goufo & Franc, 2019) have also been the subject of substantial research in recent years. Linear or nonlinear state feedback controllers have the potential to construct a variety of various sorts of 4D chaotic systems in typical 3D chaotic attractors. The computational complexity of the 4D hyperchaotic system is higher, and it possesses equal to or greater than two positive Lyapunov exponents.

In most cases, image encryption techniques relying on chaos are able to build chaotic ciphers, which are then utilised for the purpose of swapping the locations or values of the pixels present in the source image. A 2D Arnold chaos was utilised to build a 3D Arnold chaos, which was subsequently employed in image encryption (Khade & Narnaware, 2012). The findings demonstrate that the strategy is both quick and risk-free. An image cryptosystem was developed by Elghandour et al. (Elghandour et al., 2022) employing a 2D piecewise chaotic map. Here in the beginning, the simple image gets jumbled up by employing the logistic map (confusion) and piecewise chaotic map which can yield chaotic sequences. Ping et al. came up with the idea of applying Henon chaos to the image encryption system, and they demonstrated that the encryption approach could withstand a selective plaintext attack (Ping et al., 2018). An another image encryption approach based on the 3D chaotic system was also presented by Haroun et al. (Haroun & Aaron Gulliver, 2015). The image encryption methods described above make use of chaos theory. These methods rely on low-dimensional chaotic systems (LDCS) with at most one positive Lyapunov exponent. These types of chaotic systems have a number of benefits, including an easy-to-implement format; few control criteria, and a straightforward design. However, LDCS are easy to exploit because of their lack of structure. If the encryption is modified such that it uses high-dimensional chaotic systems (HDCS) rather than LDCS, then the encryption will be more successful in terms of security.

The 4D chaotic cryptosystem was suggested by Wang et al. (2022), and its purpose is to construct four chaos patterns using DNA approach. A new encryption method was also presented by Lin et al. (Lin & Li, 2021), and it was founded on the Lorenz map and RSA algorithm. Here, the RSA technique is utilised to construct the starting values of the Lorenz system, and the key stream is formed in an iterative manner. Then the data are masked through the use of additive mode diffusion so that the position of the pixel as well as its grey value may be altered. After that, the procedure for finite field diffusion is carried out to accomplish the concealment of the image information. The said technique required to be repeated twice to diffuse the pixel information throughout the complete cipher image (Lin & Li, 2021). A scanning sequence approach for preserving color image relying on the 3D-Lorenzo chaotic map was presented by Jawad et al. (Jawad, 2021). Here the scan pattern approach is utilised in order to generate three distinct masks, one of which is utilised for each channel comprising the colored image. When ciphering the image, these masks took into account the space of the shuffling pixels, which serve as input elements for the 3D-Lorenzo chaotic map.

Su et al. (Su & Wang, 2022) presented a proposal for a 4D autonomous dynamic system and conducted an analysis of the dynamic features. Here the point of equilibrium and the dissipation of the system is calculated first, and then proceed to the non-dynamic behaviour of the system by using the bifurcation diagram. In the course of the investigation, it was analysed that the presence of a wide parameter value range causes the system to remain in a hyperchaotic condition.

1.1. Motivation

Security system or algorithms having small key space are prone to multiple attacks. Since LDCS based methods offers small key space, this research proposed to build an image encryption algorithm based on the HDCS i.e. HDIEA. Simultaneously, low-dimensional chaos map architectures are considerably simpler, since there are fewer system constituents. Using chaotic signal estimation techniques, system features and beginning values may also be predicted for LDCS. On the other hand, HDCS displays exceptional chaotic behaviour as well as a complex architecture (Li et al., 2019).

1.2. Contribution

In light of the findings cited above, this article makes a suggestion about the integration of two hyper chaotic maps for the purpose of image encryption. These maps are a novel 5D Gauss Logistic map and a 4D Lorenz system. The phrases that follow describe the most important developments and contributions made by this work.

- With the addition of the Gauss Logistic approach, the structure becomes more complicated, and the chaotic performance is significantly enhanced.
- The Lyapunov exponential spectrum is used in the analysis in order to assess the efficiency of the 5D Gauss Logistic system. By doing sensitivity analysis on the starting value of the chaotic system, the performance reveals that the system offers beneficial chaotic features, ergodicity, and a broad hyperchaotic range.
- The correlation coefficients of the encrypted images are quite low; indicating extreme key sensitivity towards variables and secure mechanism.
- The technique allows a significantly large key space up to 2^{847}, which is sufficiently enough to resist a crypto attack. Furthermore, the suggested algorithm's keyspace is considerably superior to that of numerous literatures.
- For the purpose of demonstrating that the suggested method's effectiveness, several security and performance evaluations have been carried out including successful NIST's randomisation test.
- The outcomes clearly show that the LGL cryptosystem is significantly more effective and secure than the various image cryptosystem that are currently in use, and this conclusion was reached by comparing the relevant quality metrics of the encrypted image to the evaluation indicators of the decrypted image.

The suggested approach HDIEA may find applications in a variety of industries, including the protection of smart city surveillance data such as road traffic visual data, smart hospital biological image data, in the interest of national security such as military or SAR data, biometric data in personal identification, and for the variety of communication applications available.

The following is the hierarchical organisation of the paper: Section 2 gives insights about the 1D Gauss map, 1D logistic map and 4D Lorenz system. In Section 3, we have discussed the suggested 5D Gauss Logistic system, as well as its Lyapunov exponent analysis, and proposed encryption decryption method. Security analysis is examined in 4th section, followed by conclusion in Section 5.

2. Preliminaries

Nonlinear dynamical systems can be subdivided further into a simpler category known as chaotic systems. These systems may have very few interacting fragments, and those fragments may follow relatively simple laws, but they all have a highly sensitive dependency on the starting conditions. Despite their predictable simplicity, these systems are capable of producing behaviour that is both completely unexpected and radically different over time (chaotic). The chaotic map is type of transformation function that may be used to visualise chaotic activity in either continuous or discrete time. It does this by plotting the parameters against the time in either continuous or discrete form. This section examines the theoretical foundations of the three utilised chaotic maps.

2.1. Logistic map

The logistic function is represented mathematically by a differential equation that considers time to be a continuous variable. Instead of using a linear difference equation, the logistic map looks at discrete time steps using a nonlinear difference equation. Because it can map the value of the population at any given time step to the value of the population at the subsequent time step, Logistic map is expressed as (Ahuja & Doriya, 2022).

$$y_{i+1} = a \times y_i \times (1 - y_i) \tag{1}$$

Equation (1) describes the principles that govern the system, which may also be referred to as its dynamics: here, y stands for the population at any given time i and a stands for the growth rate.

2.2. Gauss map

This map is a non-linear iterated function of realistic intervals that has real parameters b and c and may be formally written as (Rahmawati & Liantoni, 2018):

$$y_{n+1} = \exp(-b \times y_n \times y_n) + c \tag{2}$$

The width of the Gauss or Gaussian curve is connected to the parameter b, while the height of the curve is related to the value c. Although the behaviour of the Gauss map is comparable to that of the logistic map, the dynamics connected with the Gauss map are more intricate due to the fact that it has two parameters. Although the majority of the attributes of the logistic map are also found in the Gauss map. But the Logistic map does not display some characteristics of the Gauss map, such as period un-doubling and bi-stability.

2.3. Lorenz map

A classic example of a chaotic structure is the Lorenz Hyperchaotic system (LHS), which is characterised as follows:

$$\left.\begin{aligned} \frac{dX}{dt} &= a(Y - X) + W \\ \frac{dY}{dt} &= X(c - Z) - Y \\ \frac{dZ}{dt} &= XY - bZ \\ \frac{dW}{dt} &= -YZ + dW \end{aligned}\right\} \tag{3}$$

The variable $a, b, c,$ and d act as the controlling parameters in Equation (3). Generally, Hyperchaotic behaviour will be exhibited by the system when $-1.52 \leq d \geq 0.06$. In an expanded form of the 3D differential equations, the Lorenz system can be represented by the 4D differential chaotic equation (Bisht et al., 2020; Tang et al., 2022). Additionally discrete continuous chaotic systems like the Lorenz Hyperchaotic system often requires the use of the Runge–Kutta technique. In the process of resolving nonlinear ordinary differential equations, this essential iterative approach is used. Equation (4) provides an illustration of the Runge–Kutta technique for the fourth order that may be stated as follows:

$$Q_{i+1} = Q_i + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4) \tag{4}$$

Here,

$$\begin{aligned} K_1 &= f(P_i, Q_i) \\ K_2 &= f\left(P_i + \frac{h}{2}, Q_i + \frac{h}{2}K_1\right) \\ K_3 &= f\left(P_i + \frac{h}{2}, Q_i + \frac{h}{2}K_2\right) \\ K_4 &= f(P_i + h, Q_i + hK_3) \end{aligned}$$

where P, h, and Q are time, time interval and function value, respectively.

3. Proposed methodology

3.1. Gauss-Logistic map

The notion of the 5D Gauss Logistic System, which will be described further in this article, has been built by us using the formulae that are presented further down in this paragraph. In Equation (5), the Gauss map is represented by the first two equations, whereas the Logistic map is described by the last three equations.

$$\left.\begin{aligned} x_{i+1} &= e^{(-r' {x_i}^2)} + t + q{y_i}^2 x_i + p{z_i}^3 \\ y_{i+1} &= e^{(-r' {y_i}^2)} + t + q{z_i}^2 y_i + p{x_i}^3 \\ z_{i+1} &= rz_i(1 - z_i) + q{x_i}^2 z_i + p{y_i}^2 \\ w_{i+1} &= rw_i(1 - w_i) + q{s_i}^2 w_i + p{z_i}^2 \\ s_{i+1} &= rs_i(1 - s_i) + q{x_i}^2 s_i + p{w_i}^2 \end{aligned}\right\} \tag{5}$$

In Equation (5), r and r' are control parameters and $p, q,$ and t are constant.

3.2. Lyapunov exponent of Gauss-logistic map

The sensitive dependency on the beginning circumstances is one characteristic that describes the quality of chaos. In chaotic systems a very little shift in the starting condition might result in significantly different results for the dynamic. A Lyapunov exponent is a number that provides an estimate of the behaviour of a chaotic system and provides information about how sensitive a system is. Therefore, it provides with added information concerning the system's butterfly effect (Su & Wang, 2022).

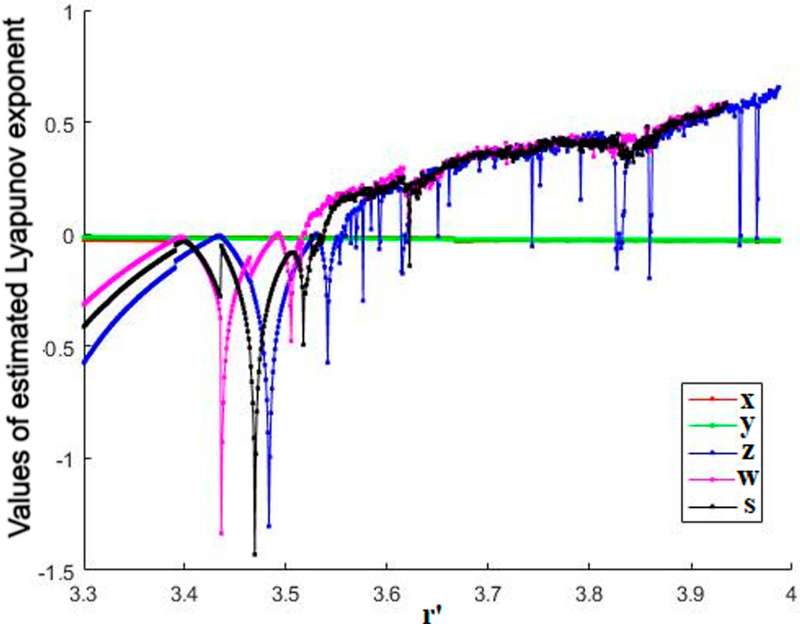


Figure 1. Lyapunov Exponent of proposed Gauss-logistic map. (With reference to Equation 5; The Gauss map is represented by the first two equations with the colors green and red, whereas the Logistic map is represented by the last three equations with the colors pink, black, and blue.)

Figure 1 provides a representation of the Lyapunov exponent for the Gauss Logistic map. When 3.35 $\leq r$ or $r' \leq$ 4.9, the system exhibits Hyperchaotic behavior, as shown in Lyapunov Graph (see Figure 1). Figure 1 demonstrates that the maps are suitable for the task of cryptography of data that is transferred over an unsecured network.

3.3. Sensitivity analysis of initial value of hyperchaotic system

When considering the Hyperchaotic system, it is important to keep in mind that the beginning circumstances have a substantial impact on the chaotic performance. Figure 2 is drawn when the 4D LHS uses the parameters $X_0 = 1.1$, $Y_0 = 2.2$, $Z_0 = 3.3$, and $W_0 = 4.4$ to perform the evaluation of the initial value's sensitivity. On the other hand, Figure 3 is drawn when the 5D Gauss-Logistic Hyperchaotic system uses the parameters $x_0 = 0.3250$, $y_0 = 0.4250$, $z_0 = 0.5250$, $w_0 = 0.4350$ and $s_0 = 0.5350$.

3.4. Encryption algorithm

In order to implement a novel symmetric image encryption strategy, this algorithm makes use of the 5D Gauss Logistic Map (refer Algorithm 1) in conjunction with Lorenz system (refer Algorithm 2) as shown in Figure 4, it is discussed in more detail below;

Step 1: The color image that has dimensions of $3 \times M \times N$ is broken down into three sub images of P_J*(J ε(R,G,B))*, and each sub image has dimensions of $M \times N$.

To perform the color image encryption operation the step 2 to step 7 will be used for each channel *(R, G, B)* separately.

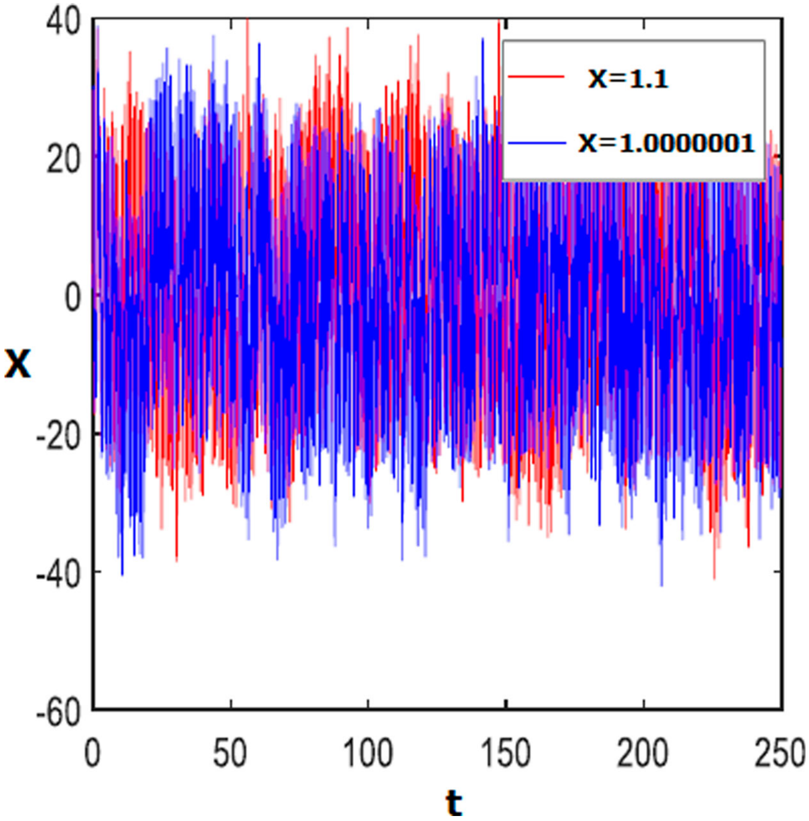


Figure 2. Sensitivity analysis of 4D Lorenz hyperchaotic system.

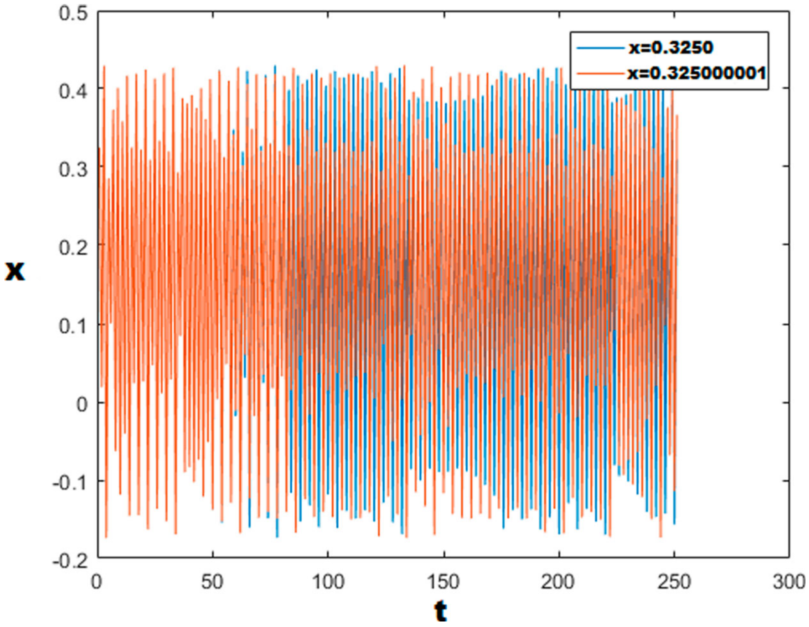


Figure 3. Sensitivity analysis of 5D Gauss-logistic hyperchaotic system.

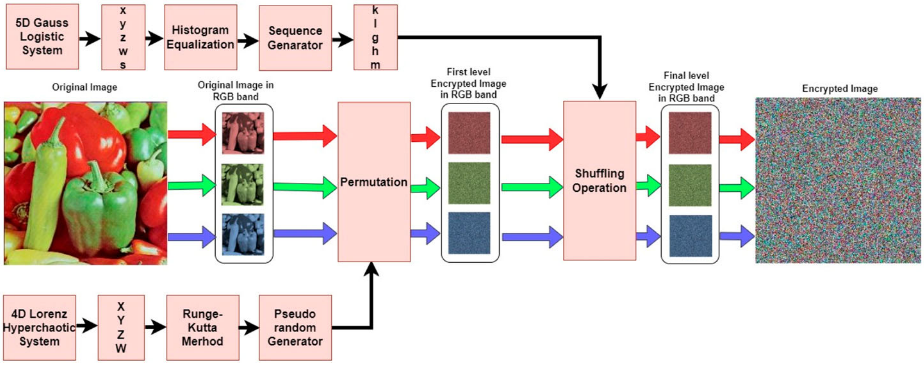


Figure 4. Proposed LGL encryption algorithm.

Step 2: By 4D Lorenz hyper chaotic system, substitute parameters X_0, Y_0, Z_0, and W_0 into Equations (3) and (4) to generate pseudorandom sequence S (given by Equation 6) and convert the generated values into the range of 0–255.

$$S = mod(floor((s + 100) \times 10^{10}), 10 \times \max(M, N)) + 1 \tag{6}$$

Step 3: Record the plain image as P and perform the permutation operation with sequence S and get first-level encrypted image I.

Step 4: By 5D Gauss Logistic system, substitute parameters x_0, y_0, z_0, w_0 and s_0 into Equation (5) to generate x, y, z, w, and s values and apply histogram equalisation using Equation (7) described below;

$$\left.\begin{array}{c} x = ceil((x \times A)mod\ M \\ y = ceil((y \times B)mod\ M \\ z = ceil((z \times C)mod\ M \\ w = ceil((w \times D)mod\ M \\ s = ceil((s \times E)mod\ M \end{array}\right\} \tag{7}$$

Step 5: Select p, q, and r, random numbers. Further by using sequence generator generates five sequences k, l, g, f and m with the help of x, y, z, w, and s.

Step 6: Record the image I, apply shuffling operation using k, l, g, and f sequences in row and column, to obtain shuffled image.

Step 7: After that XOR the shuffle image with m sequence and get the final level encrypted image V.

Step 8: After all three R, G and B encryption operation combine all the channels to get the colored encrypted image V_J $(J \in (R, G, B))$.

3.5. Decryption algorithm

The image decryption procedure (refer Algorithm 3 and 4 for pseudocode) is explained below, and Figure 5 depicts the image decryption process.

Step 1: The encrypted color image that has dimensions of $3 \times M \times N$ is broken down into three sub images of V_J $(J\, \varepsilon (R,G,B))$ and each sub image has dimensions of $M \times N$.

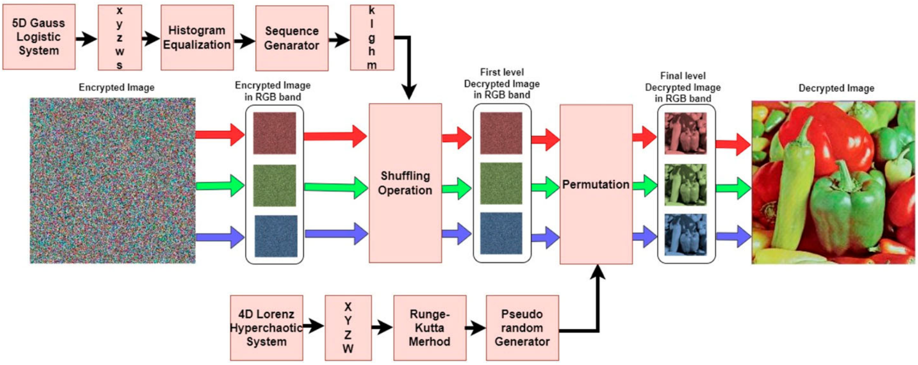


Figure 5. Proposed LGL decryption algorithm.

To perform the color image encryption operation the Step 2–7 will be used for each channel *(R,G,B)*separately.

Step 2: By 5D Gauss Logistic system, substitute parameters x_0, y_0, z_0, w_0 and s_0 into Equations (5) to generate *x, y, z, w,* and *s* values and apply histogram equalisation using Equation (7).

Step 3: Select *p, q,* and *r*, random numbers and using these numbers sequence generator generates five sequences *k, l, g, f* and *m* with the help of *x, y, z, w,* and *s*.

Step 4: Read the image *V*, apply shuffling operation using *k, l, g,* and *f* sequences in row and column, to obtain shuffled image.

Step 5: After that XOR the shuffle image with *m* sequence and get the first level decrypted image *F*.

Step 6: By 4D Lorenz hyper chaotic system, substitute parameters X_0, Y_0, Z_0, and W_0 into Equations (3) and (4) to generate pseudorandom sequence *S* (given by Equation (6)) and transform the produced numbers (0–255).

Step 7: Read the image *F* and implement the permutated sequence *S* and get final level decrypted image *H*.

Step 8: After all three *R*, *G*, and *B* encryption operation combine the all channels to get the colored decrypted image H_J $(J \in (R, G, B))$.

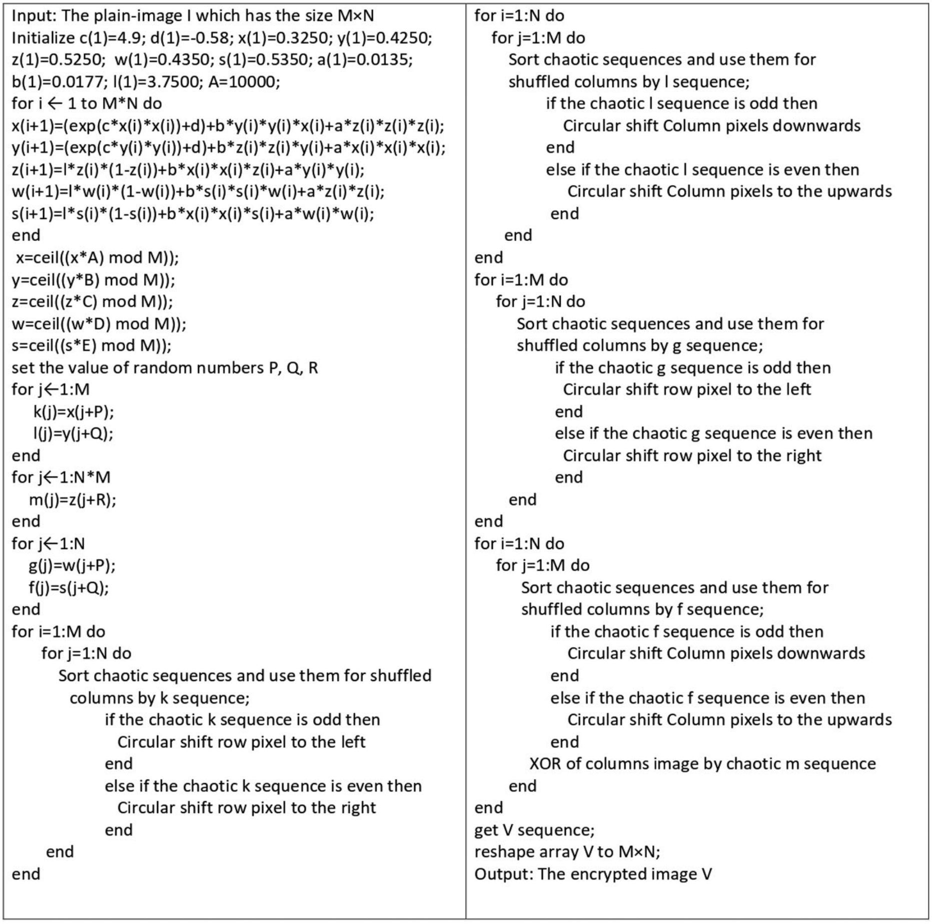
4. Analysis of simulation results

Throughout the course of this investigation, the High Dimensional Encryption method was constructed by making use of the MATLAB 2016 software platform. In this experiment, we employed Windows 10, Intel Core i5, and 8GB RAM configuration.

In the algorithm, 4D LHS uses the parameters $X_0 = 1.1$, $Y_0 = 2.2$, $Z_0 = 3.3$, and $\mathrm{W}_0 = 4.4$ of the initial value and controlling parameters are $a = 10$, $b = 8/3$, $c = 28$ and $d = -1$. On the other hand, the 5D Gauss-Logistic Hyperchaotic system uses the parameters $x_0 = 0.3250$, $y_0 = 0.4250$, $z_0 = 0.5250$, $w_0 = 0.4350$ and $s_0 = 0.5350$ of the initial values, and control parameters are $p = 0.0135$, $q = 0.0177$, $r = 3.75$, $r' = 4.9$ and $t = -0.58$.

This section contains in-depth security assessments as well as experimental data that may be used to assess the effectiveness of the proposed method. A series of conventional tests are performed on a number of images collected from a recognised database (Nilsback

Algorithm 1: Pseudocode for 5D Gauss logistic encryption method.



```
Input: The plain-image I which has the size M×N
Initialize c(1)=4.9; d(1)=-0.58; x(1)=0.3250; y(1)=0.4250;
z(1)=0.5250;  w(1)=0.4350; s(1)=0.5350; a(1)=0.0135;
b(1)=0.0177; l(1)=3.7500; A=10000;
for i ← 1 to M*N do
x(i+1)=(exp(c*x(i)*x(i))+d)+b*y(i)*y(i)*x(i)+a*z(i)*z(i)*z(i);
y(i+1)=(exp(c*y(i)*y(i))+d)+b*z(i)*z(i)*y(i)+a*x(i)*x(i)*x(i);
z(i+1)=l*z(i)*(1-z(i))+b*x(i)*x(i)*z(i)+a*y(i)*y(i);
w(i+1)=l*w(i)*(1-w(i))+b*s(i)*s(i)*w(i)+a*z(i)*z(i);
s(i+1)=l*s(i)*(1-s(i))+b*x(i)*x(i)*s(i)+a*w(i)*w(i);
end
 x=ceil((x*A) mod M));
y=ceil((y*B) mod M));
z=ceil((z*C) mod M));
w=ceil((w*D) mod M));
s=ceil((s*E) mod M));
set the value of random numbers P, Q, R
for j←1:M
    k(j)=x(j+P);
    l(j)=y(j+Q);
end
for j←1:N*M
   m(j)=z(j+R);
end
for j←1:N
   g(j)=w(j+P);
   f(j)=s(j+Q);
end
for i=1:M do
      for j=1:N do
         Sort chaotic sequences and use them for shuffled
           columns by k sequence;
                if the chaotic k sequence is odd then
                  Circular shift row pixel to the left
                end
                else if the chaotic k sequence is even then
                  Circular shift row pixel to the right
                end
       end
end
for i=1:N do
   for j=1:M do
      Sort chaotic sequences and use them for
      shuffled columns by l sequence;
            if the chaotic l sequence is odd then
               Circular shift Column pixels downwards
            end
            else if the chaotic l sequence is even then
               Circular shift Column pixels to the upwards
             end
      end
end
for i=1:M do
    for j=1:N do
       Sort chaotic sequences and use them for
       shuffled columns by g sequence;
             if the chaotic g sequence is odd then
               Circular shift row pixel to the left
             end
             else if the chaotic g sequence is even then
               Circular shift row pixel to the right
             end
       end
end
for i=1:N do
    for j=1:M do
        Sort chaotic sequences and use them for
        shuffled columns by f sequence;
             if the chaotic f sequence is odd then
               Circular shift Column pixels downwards
             end
             else if the chaotic f sequence is even then
               Circular shift Column pixels to the upwards
             end
          XOR of columns image by chaotic m sequence
       end
end
get V sequence;
reshape array V to M×N;
Output: The encrypted image V
```

& Zisserman, n.d.) (SIPI Image Database, n.d.). The results of these tests are explained in the subsections that accompany.

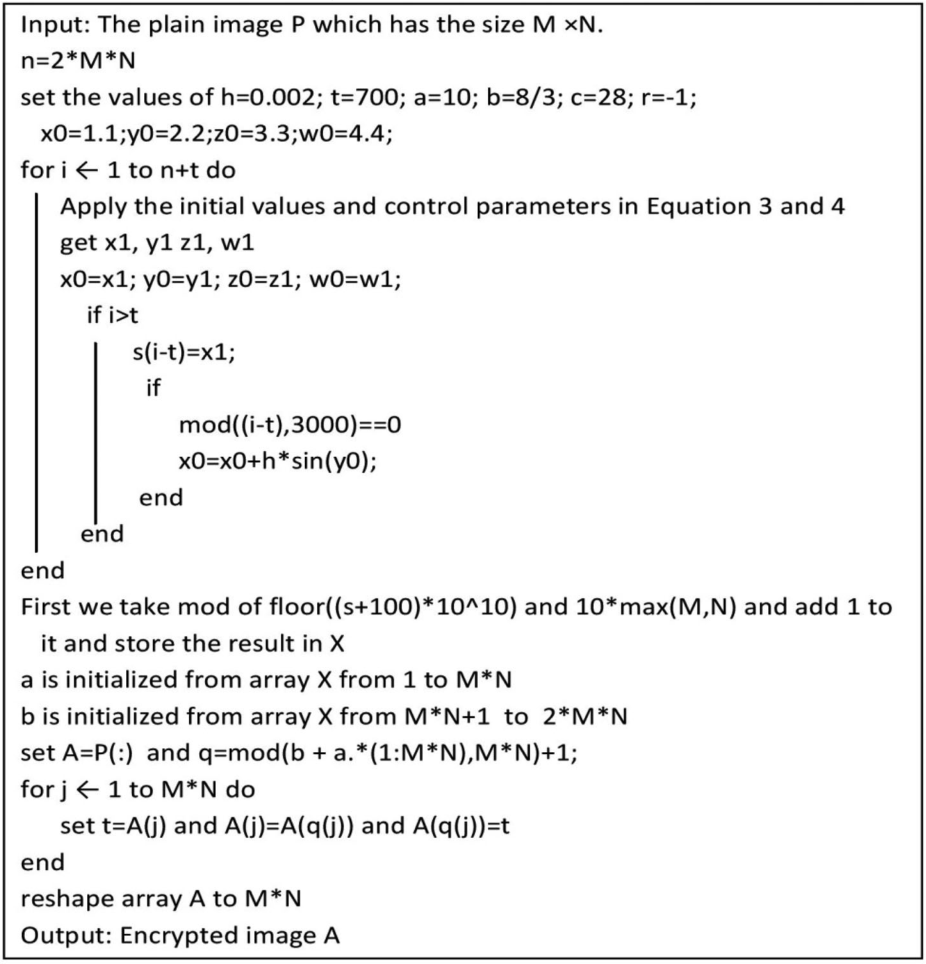
4.1. Statistical analysis

This section is devoted to in-depth statistical analysis, such as histogram analysis, correlation coefficient, and information entropy. It is essential to pay attention to the statistical characteristics of encrypted images. If the encrypted image still reveals a certain statistical rule, the adversary will probably try to crack the encryption using that rule (Song et al., 2022b; Zheng et al., 2022). A 256 × 256 Pepper color image is being used to illustrate the statistical analysis. Figure 6 also shows various original and encrypted test images for demonstration.

4.1.1. Histogram analysis

The frequency of each gray value may be finding out by looking at the histogram of the image pixels. Since Figure 7 indicates that the pixel distribution in the plain text image follows a regular pattern, the histogram for the original image should have many peaks. Evenly

Algorithm 2: Pseudocode for Lorenz encryption method.



```
Input: The plain image P which has the size M ×N.
n=2*M*N
set the values of h=0.002; t=700; a=10; b=8/3; c=28; r=-1;
  x0=1.1;y0=2.2;z0=3.3;w0=4.4;
for i ← 1 to n+t do
    Apply the initial values and control parameters in Equation 3 and 4
    get x1, y1 z1, w1
    x0=x1; y0=y1; z0=z1; w0=w1;
      if i>t
          s(i-t)=x1;
           if
              mod((i-t),3000)==0
              x0=x0+h*sin(y0);
           end
      end
end
First we take mod of floor((s+100)*10^10) and 10*max(M,N) and add 1 to
  it and store the result in X
a is initialized from array X from 1 to M*N
b is initialized from array X from M*N+1  to  2*M*N
set A=P(:)  and q=mod(b + a.*(1:M*N),M*N)+1;
for j ← 1 to M*N do
    set t=A(j) and A(j)=A(q(j)) and A(q(j))=t
end
reshape array A to M*N
Output: Encrypted image A
```

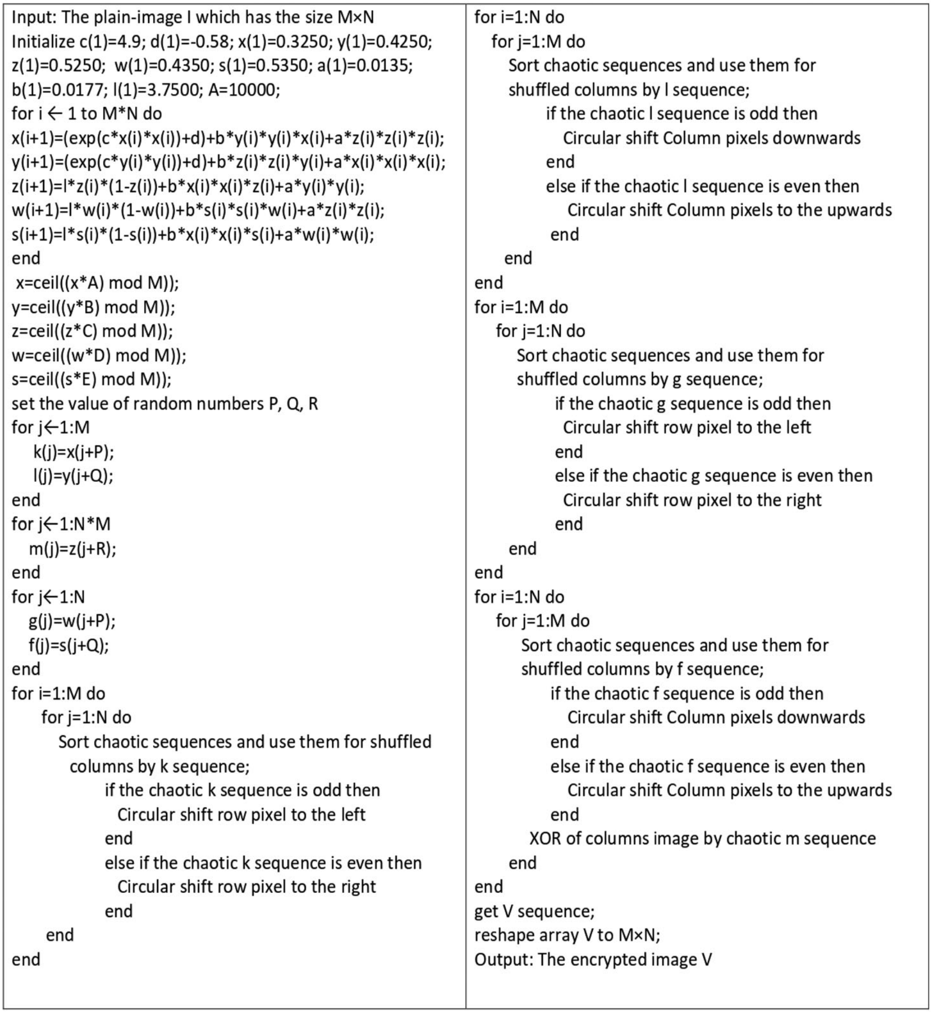
distribution of encrypted image histogram indicates that the encryption was effective. Also, smoother histograms reveal grey values closer to the average. The discrete image that was encrypted using the suggested encryption technique indicates that the ciphered image has a uniform pixel distribution even though there is no distribution parameters specified for the image. Furthermore, the Chi-square test is used as quantitative metric to demonstrate the attained uniformity (Ravichandran et al., 2016), and the results are shown in Table 1 for different encrypted images.

All of the *p*-values for the encrypted image were found to be > 0.05 (5% significant), demonstrating that the suggested encryption technique accepts the null hypothesis and establishing the histogram's uniformity. The outcomes demonstrate that the suggested cryptosystem performs well against statistical attacks.

4.1.2. Image pixel correlation analysis

The correlation coefficient measures the degree to which neighbouring image pixels have a linear association with one another. Ordinarily, an image has a substantial connection

Algorithm 3: Pseudocode for 5D Gauss logistic decryption method.



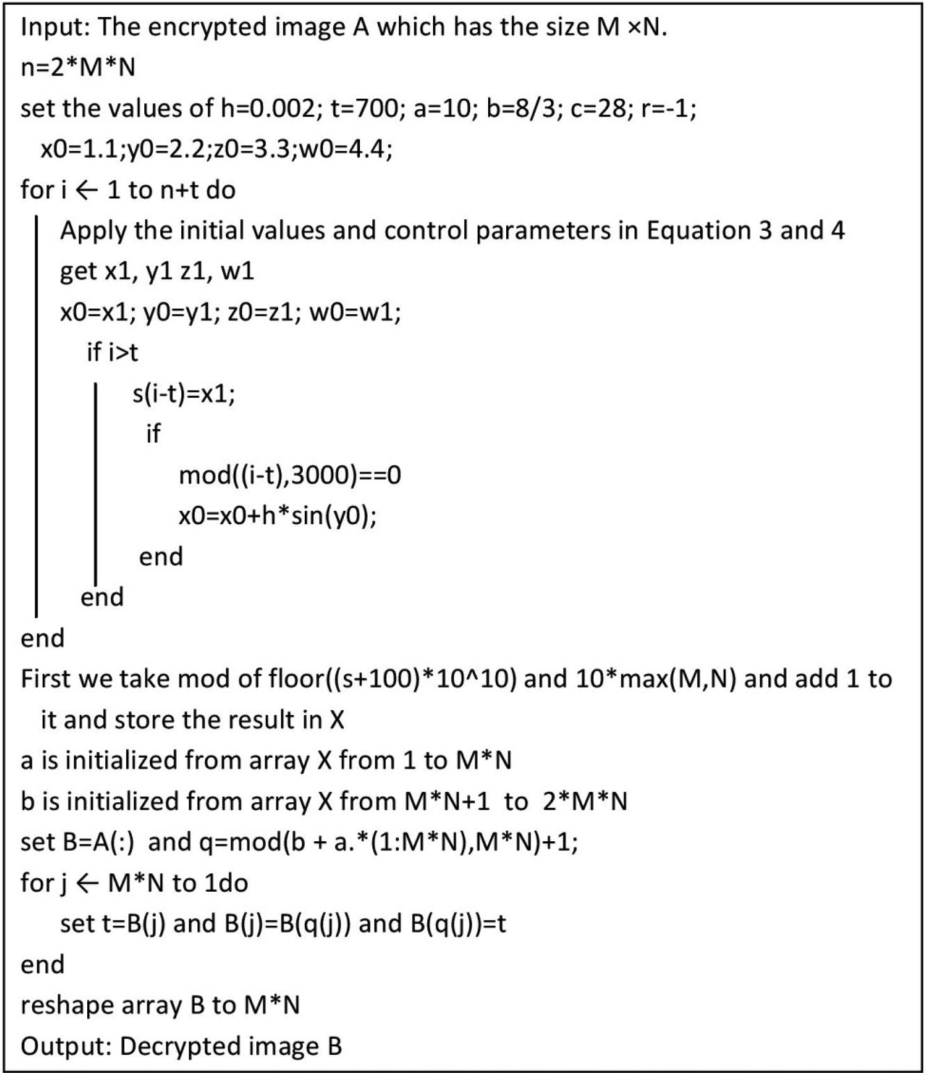
```
Input: The plain-image I which has the size M×N
Initialize c(1)=4.9; d(1)=-0.58; x(1)=0.3250; y(1)=0.4250;
z(1)=0.5250;  w(1)=0.4350; s(1)=0.5350; a(1)=0.0135;
b(1)=0.0177; l(1)=3.7500; A=10000;
for i ← 1 to M*N do
x(i+1)=(exp(c*x(i)*x(i))+d)+b*y(i)*y(i)*x(i)+a*z(i)*z(i)*z(i);
y(i+1)=(exp(c*y(i)*y(i))+d)+b*z(i)*z(i)*y(i)+a*x(i)*x(i)*x(i);
z(i+1)=l*z(i)*(1-z(i))+b*x(i)*x(i)*z(i)+a*y(i)*y(i);
w(i+1)=l*w(i)*(1-w(i))+b*s(i)*s(i)*w(i)+a*z(i)*z(i);
s(i+1)=l*s(i)*(1-s(i))+b*x(i)*x(i)*s(i)+a*w(i)*w(i);
end
 x=ceil((x*A) mod M));
y=ceil((y*B) mod M));
z=ceil((z*C) mod M));
w=ceil((w*D) mod M));
s=ceil((s*E) mod M));
set the value of random numbers P, Q, R
for j←1:M
    k(j)=x(j+P);
    l(j)=y(j+Q);
end
for j←1:N*M
   m(j)=z(j+R);
end
for j←1:N
   g(j)=w(j+P);
   f(j)=s(j+Q);
end
for i=1:M do
     for j=1:N do
        Sort chaotic sequences and use them for shuffled
          columns by k sequence;
              if the chaotic k sequence is odd then
                Circular shift row pixel to the left
              end
              else if the chaotic k sequence is even then
                Circular shift row pixel to the right
              end
      end
end
for i=1:N do
   for j=1:M do
      Sort chaotic sequences and use them for
      shuffled columns by l sequence;
            if the chaotic l sequence is odd then
               Circular shift Column pixels downwards
            end
            else if the chaotic l sequence is even then
               Circular shift Column pixels to the upwards
            end
     end
end
for i=1:M do
   for j=1:N do
      Sort chaotic sequences and use them for
      shuffled columns by g sequence;
            if the chaotic g sequence is odd then
              Circular shift row pixel to the left
            end
            else if the chaotic g sequence is even then
              Circular shift row pixel to the right
            end
      end
end
for i=1:N do
   for j=1:M do
      Sort chaotic sequences and use them for
      shuffled columns by f sequence;
           if the chaotic f sequence is odd then
              Circular shift Column pixels downwards
           end
           else if the chaotic f sequence is even then
              Circular shift Column pixels to the upwards
           end
        XOR of columns image by chaotic m sequence
      end
end
get V sequence;
reshape array V to M×N;
Output: The encrypted image V
```

among adjacent pixels in horizontal, vertical, and diagonal dimensions, but in cipher images, there must be no correlation among adjacent pixels in any direction (Veena & Ramakrishna, 2021). The equation for the computation may be stated as (Dhopavkar et al., 2022):

$$\rho = \frac{\left(\left(n\sum xy\right) - \left(\sum x\right)\left(\sum y\right)\right)}{\left(\sqrt[2]{\left(\left(n\sum x2\right)} - \left(\sum x\right)^2 \times \left(\left(n\sum y^2\right) - (y)^2\right)\right)} \tag{8}$$

The value of the correlation coefficient ranges from minus one to plus one. Encryption algorithms work more effectively on the cipher text image when there is a weak correlation between the pixels that are adjacent to one another. On the other hand, the encryption technique works less effectively when there is an intense correlation between the pixels that are adjacent to one another. The correlation between two random sequences is closer to zero, which signifies that the impact of encryption is improved. A value of that is less

Algorithm 4: Pseudocode for Lorenz decryption method.



```
Input: The encrypted image A which has the size M ×N.
n=2*M*N
set the values of h=0.002; t=700; a=10; b=8/3; c=28; r=-1;
  x0=1.1;y0=2.2;z0=3.3;w0=4.4;
for i ← 1 to n+t do
    Apply the initial values and control parameters in Equation 3 and 4
    get x1, y1 z1, w1
    x0=x1; y0=y1; z0=z1; w0=w1;
      if i>t
          s(i-t)=x1;
           if
               mod((i-t),3000)==0
               x0=x0+h*sin(y0);
          end
      end
end
First we take mod of floor((s+100)*10^10) and 10*max(M,N) and add 1 to
  it and store the result in X
a is initialized from array X from 1 to M*N
b is initialized from array X from M*N+1  to  2*M*N
set B=A(:)  and q=mod(b + a.*(1:M*N),M*N)+1;
for j ← M*N to 1do
    set t=B(j) and B(j)=B(q(j)) and B(q(j))=t
end
reshape array B to M*N
Output: Decrypted image B
```

than 1 shows the existence of differences between the original and encrypted versions of the image.

Figure 8 shows the correlation coefficient that may be found between the encrypted and original image (which depicts Colored Pepper). The correlation coefficients of several images when seen in the horizontal, vertical, and diagonal orientations are calculated and shown in Tables 2 and 3 depicts the correlation coefficient of colored Baboon, Aeroplane and Pepper image in the RGB components. The results of the correlation coefficient calculations for test images encrypted using various methods are compared in Table 4. In Table 5, comparison of correlation coefficients of colored test images in the RGB components with various methods are illustrated. The findings demonstrate that the suggested encryption technique is capable of efficiently fending off statistical attacks and breaks the high correlation that was present in the original image.

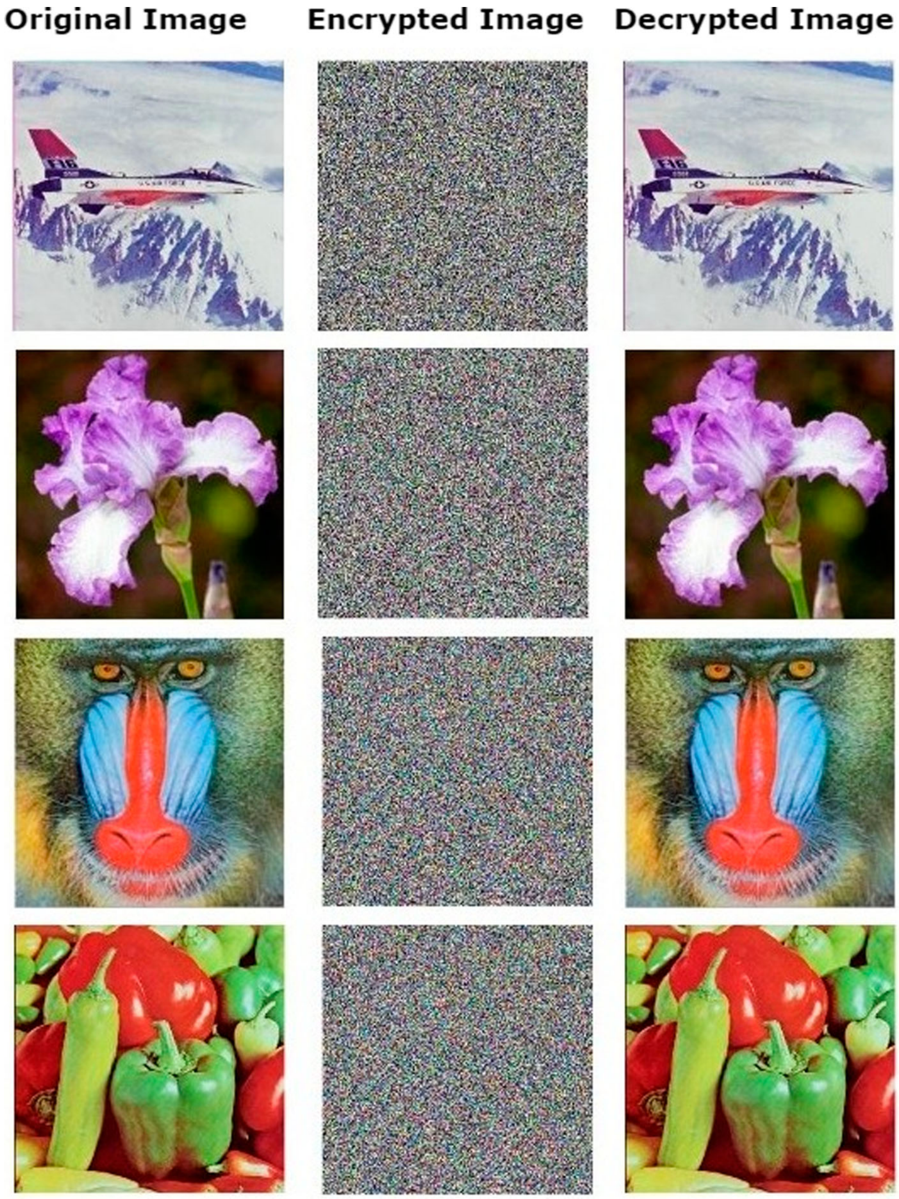


Figure 6. Original, encrypted, and decrypted colored test images with proposed algorithm.

4.1.3. Information entropy

The entropy of information is a crucial quantity that accurately reflects the information's degree of unpredictability. The information entropy of any image may be determined using the equation (Li et al., 2022) stated below:

$$I(s) = \sum_{i=1}^{n} p(s_i) \log \frac{1}{p(s_i)} \tag{9}$$

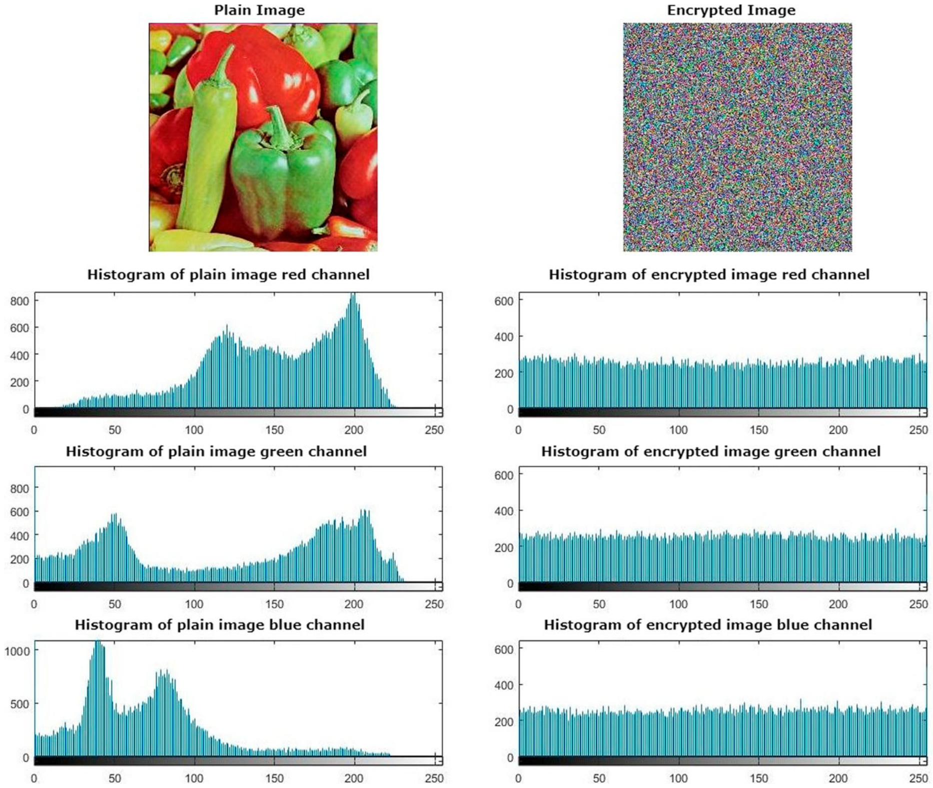


Figure 7. Comparative histogram of HDIEA.

Table 1. Histogram uniformity evaluation by chi-square test.

Images	*p* values	Decision
Baboon	0.54312	Accept
Lena	0.45393	Accept
Flower	0.75234	Accept
Pepper	0.21427	Accept
Aeroplane	0.65492	Accept
Tree	0.18762	Accept
House	0.76437	Accept
Buttercup Flower	0.87641	Accept

where $p(s_i)$ is the probability of the presence of pixel and n is the number of gray levels that the pixel contains (Gupta & Vijay, 2022). The gray level of an 8-bit image has a value of $2^8 = 256$, and its n value is also 256. In this scenario, the occurrence probability of all gray levels is equal to 1/256, and the optimal information entropy $I(s)$is equal to 8. This is the case if the encrypted image is perfectly uniform. The entropy values of the several test images are shown in Table 6. In Table 7, an example of a 256 × 256 pixel Lena image is used to illustrate how the information entropy of various types of literature varies. The findings demonstrate that the suggested algorithm provides a high level of security.

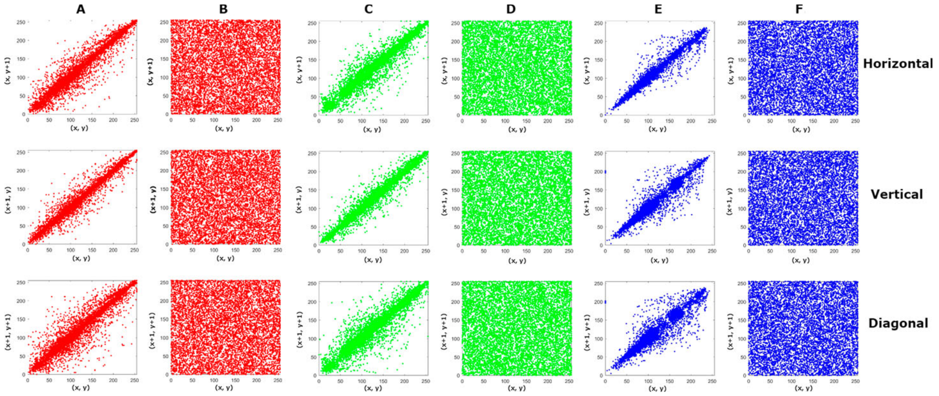


Figure 8. The pixel distribution of different (RGB) components of pepper original and encrypted image in horizontal, vertical and diagonal directions.

Table 2. Correlation coefficients of original and encrypted test images.

	Correlation coefficients for original image			Correlation coefficients for encrypted image		
Images	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Baboon	0.9631	0.9440	0.9213	−0.0080	−0.0116	0.0291
Lena	0.9647	0.9802	0.9463	−0.0019	−0.0016	−0.0069
Flower	0.9890	0.9913	0.9829	−0.0129	0.0069	0.0120
Pepper	0.9244	0.9733	0.9083	0.0012	−0.0122	−0.0007
Aeroplane	0.9253	0.9248	0.8910	−0.0080	0.0381	−0.0047
Tree	0.9689	0.9531	0.9434	−0.0051	0.0231	0.0379
House	0.9812	0.9514	0.9376	−0.0244	0.0196	−0.0065
Buttercup Flower	0.9963	0.9963	0.9933	−0.0098	0.0024	−0.0112

Table 3. Correlation coefficients of colored test images in the RGB components.

Methods	Correlation direction	R	G	B
Baboon	Horizontal	−0.0104	0.0025	−0.0077
	Vertical	−0.0175	−0.0010	−0.0171
	Diagonal	−0.0041	−0.0040	0.0035
Aeroplane	Horizontal	−0.0028	−0.0131	−0.0067
	Vertical	−0.0383	0.0074	0.0151
	Diagonal	0.0137	−0.0035	−0.0057
Pepper	Horizontal	0.0072	−0.0046	0.0018
	Vertical	−0.0013	−0.0079	0.0243
	Diagonal	−0.0009	−0.0077	−0.0047

4.1.4. Local entropy

The aforementioned entropy measure, known as “global entropy” in the cryptosystem, may sometimes mislead the true randomness of images. Extremely high entropy levels that are close to their maximum, as evaluated by the global Shannon technique, may not always represent real randomness. This is due to the fact that two images, such as one that is random and another that is perceptible and recognised, might have the same global entropy value. In order to overcome the issue of the global entropy measurement, Wu et al. (2013) suggested the local Shannon entropy metric for testing the randomness of cipher images. It is

Table 4. Comparison of correlation coefficients of colored test images.

		Correlation coefficient		
Image encryption algorithms	Images	Horizontal	Vertical	Diagonal
Proposed method	Lena	−0.0019	−0.0016	−0.0069
	Baboon	−0.0080	−0.0116	0.0291
	Pepper	0.0012	−0.0122	−0.0007
Ref (Jawad, 2021)	Lena	0.00091	0.00082	0.00065
	Baboon	−0.00092	−0.00078	0.00076
	Pepper	−0.00081	0.00083	−0.00070
Ref (Jarjar et al., 2022)	Baboon	−0.0007	−0.0004	0.0001
	Pepper	−0.0002	0.0006	0.0002
Ref (Feixiang et al., 2021)	Lena	0.0103	0.0049	0.0072
Ref (X. Wang & Yang, 2021)	Lena	−0.0009	−0.0003	0.0010
Ref (Khalil et al., 2021)	Lena	0.0023	−0.0012	−0.0001
Ref (Khedmati et al., 2020)	Lena	0.0034	0.0011	0.0012
Ref (Lin & Li, 2021)	Lena	−0.0328	0.0105	−0.0330
	Baboon	−0.0179	−0.0060	0.0181
	Pepper	−0.0195	−0.0101	−0.0109
Ref (Yan et al., 2023)	Lena	−0.0021	0.0051	0.0068

calculated by averaging the local entropy values of a random selection of non-overlapping image blocks.

It is expressed mathematically as;

$$H_{n,T_B}(s) = \sum_{i=1}^{n} \frac{H(S_i)}{n} \tag{10}$$

where T_B is the local block size and n is the number of blocks represented by S_i. The (n, T_B) local Shannon entropy metric is highly excellent at catching local image block unpredictability, which the global Shannon entropy score may often miss. In the experiment, the parameter n is set to 30, which is the minimum number of randomly chosen non-overlapping image segments necessary, in accordance with the advice (Wu et al., 2013) . Local entropy results for different encrypted images are shown in Table 8.

4.2. Differential attacks

In a differential attack, in particular, two encrypted images are compared in order to investigate the connection that exists between the original and encrypted image that corresponds to it (Xu et al., 2022). Two popular measures of an object's ability to withstand an attack are referred to as the Number of Pixels Change Rate (NPCR) and the Unified Average Changing Intensity (UACI). Changing a pixel value in the original image allows NPCR to measure the rate of change of pixel values in an encrypted image, while UACI examines the average changing intensity between the original and encrypted image. NPCR and UACI are expressed as (Abdullah & Abdullah, 2019).

$$\text{NPCR} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} K(i,j) \times 100\% \tag{11}$$

$$\text{UACI} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{|a_1(i,j) - a_2(i,j)|}{255} \times 100\% \tag{12}$$

Table 5. Comparison of correlation coefficients of colored test images in the R G B components.

Methods	Image	Correlation direction	R	G	B
Proposed method	Lena	Horizontal	−0.0106	−0.0002	−0.0082
		Vertical	−0.0033	−0.0138	−0.0020
		Diagonal	−0.0061	0.0006	−0.0131
	Baboon	Horizontal	−0.0104	0.0025	−0.0077
		Vertical	−0.0175	−0.0010	−0.0171
		Diagonal	−0.0041	−0.0040	0.0035
Ref (Shahna & Mohamed, 2021)	Lena	Horizontal	0.0005	−0.004	0.0034
		Vertical	0.001	−0.001	−0.002
		Diagonal	0.0005	0.0008	−0.0019
	Baboon	Horizontal	0.0014	0.0068	0.0006
		Vertical	0.0014	−0.003	−0.005
		Diagonal	0.0029	−0.0023	−0.0058
Ref (Hosny et al., 2021)	Lena	Horizontal	0.0064	0.0009	0.0091
		Vertical	0.0160	0.0034	−0.0045
		Diagonal	−0.0026	0.0125	−0.0090
	Baboon	Horizontal	−0.0213	0.0126	−0.0102
		Vertical	0.0072	0.0120	0.0015
		Diagonal	0.0011	−0.0133	0.0025
Ref (Wu et al., 2018)	Lena	Horizontal	0.0137	−0.0246	−0.0137
		Vertical	−0.0237	−0.0170	0.0023
		Diagonal	0.0109	−0.0133	−0.0013
Ref (Girdhar & Kumar, 2018)	Lena	Horizontal	−0.0001	−0.0011	−0.0010
		Vertical	0.0026	0.0009	−0.0030
		Diagonal	−0.0053	0.0026	−0.0051
	Baboon	Horizontal	−0.0017	0.0028	0.0041
		Vertical	−0.0007	0.0039	0.0061
		Diagonal	0.0015	0.0015	0.0025
Ref (Zhang et al., 2020)	Lena	Horizontal	0.0014	0.0033	0.0021
		Vertical	0.0048	−0.0006	0.0002
		Diagonal	0.0002	0.0048	−0.0040
	Baboon	Horizontal	0.001391	−0.008134	−0.008891
		Vertical	0.004650	0.000829	0.000056
		Diagonal	0.000334	0.005334	0.001710
Ref (Chai et al., 2019)	Lena	Horizontal	−0.0029	−0.0032	0.0040
		Vertical	0.0013	−0.0032	−0.0018
		Diagonal	−0.0026	−0.0039	0.0012
Ref (Liu et al., 2022)	Lena	Horizontal	−0.0046	−0.0015	0.0091
		Vertical	0.0072	0.0056	−0.0076
		Diagonal	0.0009	−0.0125	−0.0145
Ref (Li et al., 2022)	Baboon	Horizontal	0.0043	0.0019	0.0024
		Vertical	0.0023	0.0033	0.0023
		Diagonal	0.0029	−0.0030	0.0001

In Equation (12), a_1 represents the cipher image, whereas a_2 represents the modified cipher image that results when one of the pixel values in the original image is altered. The NPCR and UACI values for different size of the test images are shown in Table 9.

NPCR and UACI often fall around 99.95% and 33.52%, respectively. Among all possible values, these are the ones that come closest to matching the theoretical ones. Table 10 shows a comparison between the recommended method and other algorithms found in the literature.

4.3. Key space analysis

It is very necessary for a powerful encryption algorithm to have the ability to survive attacks that use brute force. When the key space is greater than 2^{100}, it is generally acknowledged

Table 6. Information entropy results for the proposed algorithm on different test images.

Images	Entropy plain image	Entropy encrypted image
Baboon	7.6792	7.9982
Lena	7.7599	7.9997
Iris Flower	7.7164	7.9983
Pepper	7.6629	7.9971
Aeroplane	6.6587	7.9980
Tree	7.5371	7.9971
House	7.0686	7.9992
Buttercup flower	7.6364	7.9996

Table 7. Comparison of Information entropy with different literature.

	Information entropy			
Methods	R	G	B	Mean
Proposed method	7.9995	7.9997	7.9996	7.9996
Ref (ul Haq & Shah, 2021)	7.9967	7.9973	7.9970	7.9970
Ref (Liu et al., 2020)	7.9917	7.9912	7.9917	7.9915
Ref (Hosny et al., 2021)	7.9974	7.9976	7.9974	7.9975
Ref (Girdhar & Kumar, 2018)	7.9974	7.9969	7.9979	7.9974
Ref (Chai et al., 2019)	7.9973	7.9969	7.9971	7.9971
Ref (Es-Sabry et al., 2022)	7.997080	7.997886	7.997364	7.99744
Ref (Shahna & Mohamed, 2021)	N/A	N/A	N/A	7.998967
Ref (Lin & Li, 2021)	N/A	N/A	N/A	7.9993
Ref (Jawad, 2021)	N/A	N/A	N/A	7.9984

N/A – not available.

Table 8. Local Entropy results for the proposed algorithm on different test images.

Images	Local Entropy
Baboon	7.9032
Lena	7.9014
Iris Flower	7.9021
Pepper	7.9012
Aeroplane	7.9024
Tree	7.9031
House	7.9023
Buttercup Flower	7.9027

among the community of security professionals that brute force attacks may be successfully resisted. The starting values of the chaotic maps and the parameter of the chaotic maps each make up one component of the key space in the method that has been proposed. When attempting to quantify the complete keyspace with the aid of the IEEE floating-point norm (Zefreh, 2020), Equation (13) is beneficial. Comparison is shown in Table 11 (Proposed vs other literature).

$$\begin{aligned}\text{Keyspace} &= 10^{15} \times 10^{15} \times 10^{15} \times 10^{15} \times 10^{15} \times 10^{15} \times 10^{15} \times 10^{15} \times 10^{15} \times 10^{15} \\ &\times 10^{15} \times 10^{15} \times 10^{15} \times \ 10^{15} \times 10^{15} \times 10^{15} \times 10^{15} = \ 10^{255} \approx 2^{847} \qquad (13)\end{aligned}$$

Table 9. NPCR AND UACI values of different size test images.

	256× 256		512× 512		1024 × 1024	
Images	NPCR	UACI	NPCR	UACI	NPCR	UACI
Baboon	99.61	33.32	99.66	33.39	99.66	33.44
Lena	99.63	33.35	99.65	33.42	99.66	33.45
Iris Flower	99.62	33.34	99.64	33.39	99.65	33.46
Pepper	99.65	33.38	99.66	33.40	99.66	33.47
Aeroplane	99.65	33.37	99.66	33.41	99.66	33.46
Tree	99.62	33.35	99.63	33.42	99.64	33.47
House	99.62	33.34	99.65	33.42	99.65	33.45
Buttercup Flower	99.64	33.39	99.65	33.41	99.66	33.46

Table 10. Comparison among the suggested algorithm and the algorithms in literature based on NPCR and UACI.

	256× 256 Lena Image		512× 512 Lena Image	
Methods	NPCR	UACI	NPCR	UACI
Proposed	99.63	33.35	99.65	33.42
Ref (Yan et al., 2023)	99.6220	33.48	99.61	33.43
Ref (Shahna & Mohamed, 2021)	99.60	33.4407	N/A	N/A
Ref (Dhopavkar et al., 2022)	N/A	N/A	99.6189	32.9215
Ref (Bhat et al., 2022)	N/A	N/A	99.60	33.70
Ref (Rahman et al., 2022)	99.814	0.33625	N/A	N/A

N/A – not available.

Table 11. Comparison of keyspace in different literatures.

Key space	Proposed	Ref (Elghandour et al., 2022)	Ref (Ahuja & Doriya, 2021)	Ref (Chai et al., 2021)	Ref (Li et al., 2019)	Ref (Hosny et al., 2021)	Ref (Jawad, 2021)	Ref (Jarjar et al., 2022)	Ref (Yan et al., 2023)
	2^{847}	2^{500}	2^{200}	2^{512}	2^{576}	2^{116}	2^{156}	2^{180}	2^{207}

4.4. Key sensitivity analysis

The "key sensitivity study" compares two cipher images produced by encrypting the same plain image with a key that has been drastically varied. If there are major differences between the two cipher images, then the image encryption method has a high key sensitivity; if there are just subtle differences, then the method has a low key sensitivity. A high level of key sensitivity is essential for a reliable image encryption system. Even with a very little change in the key, the method that provides a very high level of security cannot be broken. Even with a relatively little adjustment, the original image is unable to be decrypted when the parameters of the 5D Gauss Logistic map are altered throughout the decryption process. For this experimentation, we have examined the key sensitivity by using the Pepper image, with $x_0 = 0.2350 + 10^{-16}$. It is observed that with a change in x_0, the decrypted image is blurred. The results of the tests are shown in Figure 9, and it may be observed that even a small change in the key hinders the process of successful decryption. So, it can be shown that the proposed method of encryption is very sensitive to the key.

Additionally, to provide the desired level of security, numerous encryption images w.r.t. one image should be created in response to slightly variable key values. A statistic known as the cipher-text difference rate (CDR) is commonly employed to examine the sensitivity

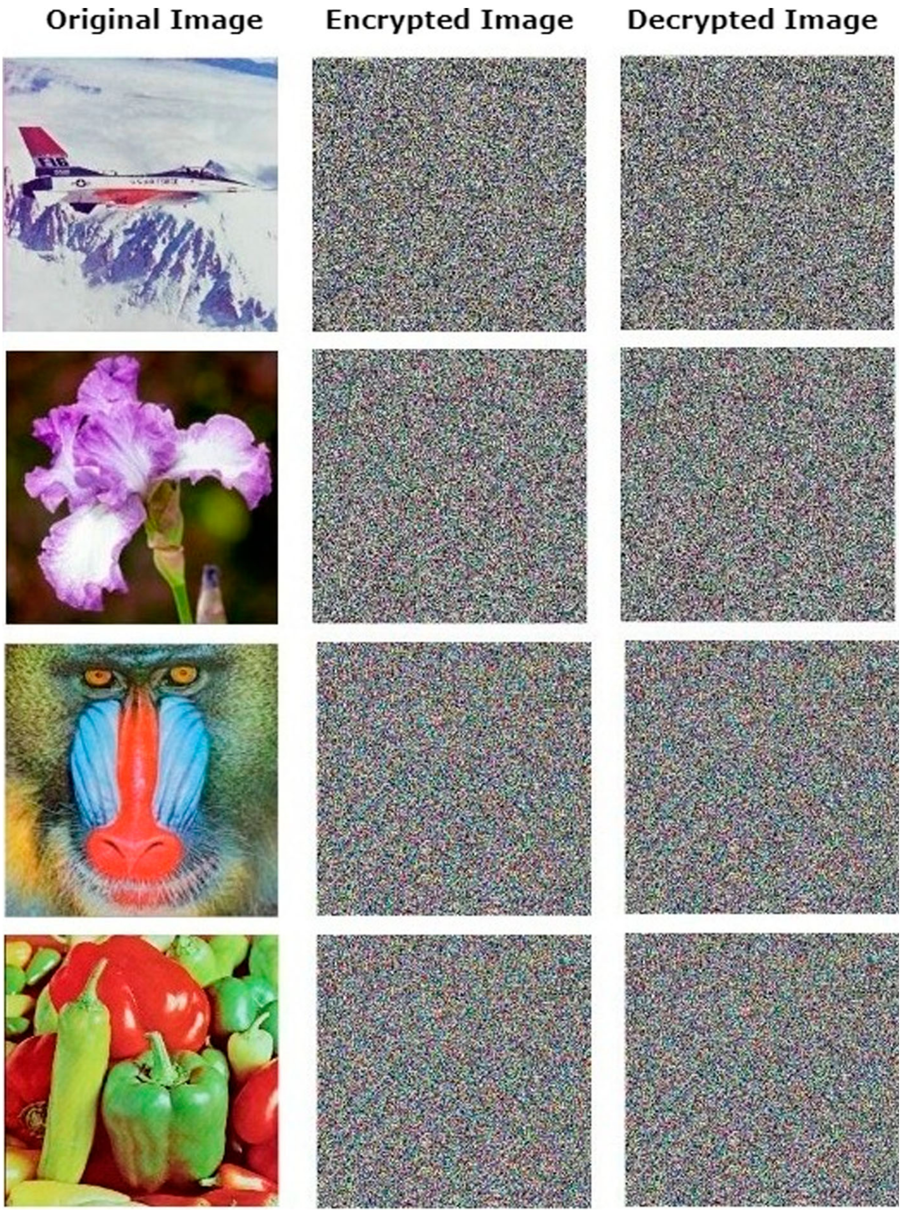


Figure 9. Key sensitivity analysis of the proposed algorithm on test images.

of secret keys (Yavuz, 2019). Table 12 shows the CDRs generated for encryptions as a result of changing secret keys as a percentage. In general, a CDR of more than 99% is considered adequate key sensitivity for an encryption scheme (Yavuz, 2019). Considering the data in Table 12, we can infer that the proposed cryptosystem has sufficient key sensitivity to fulfill the aforementioned condition. Figure 10 also displays the results of encrypting a pepper image using a secret key that has been slightly modified. The details of the subfigures in Figure 10 are as follows:

Table 12. CDRs estimation as a result of changing secret keys for encryptions.

Altered secret key	Slight change in	CDR (%)
x_0	$x_0{}^1$	99.59
	$x_0{}^2$	99.61
y_0	$y_0{}^1$	99.60
	$y_0{}^2$	99.62
z_0	$z_0{}^1$	99.58
	$z_0{}^2$	99.61
w_0	$w_0{}^1$	99.61
	$w_0{}^2$	99.61
s_0	$s_0{}^1$	99.62
	$s_0{}^2$	99.60

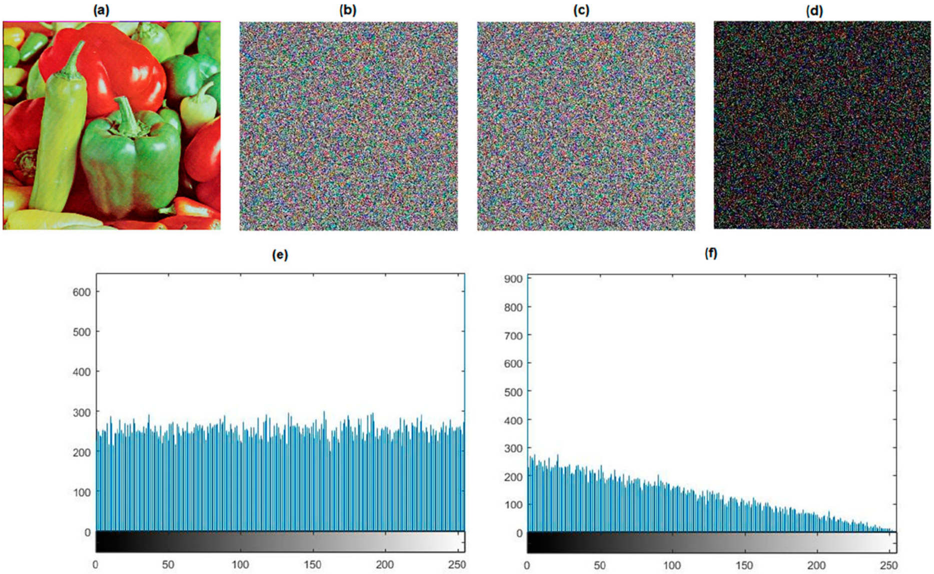


Figure 10. Key sensitivity analysis for encrypted pepper image with a slight change in one of the secret keys.

(a) The original test image.
(b) Encrypted image *AC* with the secret key $x_0{}^1 = 0.235$.
(c) Encrypted image with a slight difference in one of the secret keys *AC′* (secret key $x_0{}^1 = 0.2350000000000001$)
(d) Absolute intensity differences ($|AC\text{-}AC'|$) of corresponding pixels of encrypted images
(e) The histogram of *AC′* image
(f) Histogram of intensity difference ($|AC\text{-}AC'|$).

4.5. NIST test

The unpredictability of the sequences produced by the 5D Gauss Logistic Hyperchaotic system was examined with the help of NIST SP800-22 (see Table 13). The NIST SP800-22 test provides information on the random qualities of the sequence. Every test result with a *P* value should fall anywhere between 0 and 1, which indicates that the chaotic sequence successfully passes the evaluation (Yang et al., 2020).

4.6. Image quality analysis

Image quality is measured by the peak signal to noise ratio (PSNR). It evaluates noise between plaintext and cipher images (Arif et al., 2022). To figure out the authenticity and strength of the proposed algorithm, the mean square error (MSE) by Equation (12), and PSNR by Equation (13) were calculated.

$$\mathrm{MSE} = \frac{1}{\mathrm{MN}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [f'(i,j) - f(i,j)]^2 \tag{14}$$

$$\mathrm{PSNR} = 10\log_{10}\left[\frac{256 \times 256}{\mathrm{MSE}}\right] \tag{15}$$

The dimension of the image is represented in the equations described above by the product $M \times N$, where M is the row of the matrix and N is the column of the matrix. In other words, the size of the image is the product of the matrix's row and its column (Liu & Ding, 2020).

The structural similarity index, also known as the SSIM index, is another statistic that assesses the overall quality of the image. Calculating the relationship between an original image and a reconstructed one could well be done with the help of SSIM. It is recommended that the SSIM be characterised as (Liu & Ding, 2020);

$$\begin{aligned}\mathrm{SSIM}(p,q) &= f(l(p,q)c(p,q)s(p,q))\\ &= [l(p,q)]^{\alpha}[c(p,q)]^{\beta}[s(p,q)]^{\gamma}\end{aligned} \tag{16}$$

Brightness, contrast, and structure are all controlled by the contrast function, which is denoted by the letters $l(p,q)c(p,q)s(p,q)$. The equation, which adjusts the relative significance of these three module, has the terms, α, β and γ, and all are greater than 0. Assuming that α, β, and $\gamma = 1$.

The SSIM measurement function has a value range of [0,1] for its range of acceptable values. If the SSIM that is computed as 1, the image distortion is going to be minimal, and the decrypted image will then be same as original image, visually. And the suggested scheme also passed this test (SSIM = 1). This provides more evidence that the suggested algorithm

Table 13. Randomness test for the 5D Gauss logistic hyperchaotic system's sequences.

Test	*P* values	Results
Frequency	0.5659	Pass
Block frequency	0.6514	Pass
Cumulative sums forward & reverse	0.6782	Pass
Runs	0.8475	Pass
Rank	0.3785	Pass
The discrete Fourier transform test	0.5345	Pass
Overlapping template	0.2345	Pass
Approximate entropy	0.3234	Pass
Linear complexity	0.1145	Pass
Serial	0.1454	Pass

Table 14. Test scores for the proposed algorithm such as PSNR, SSIM, MSE.

Images	PSNR	SSIM	MSE
Baboon	∞	1	0
Lena	∞	1	0
Iris Flower	∞	1	0
Pepper	∞	1	0
Aeroplane	∞	1	0
Tree	∞	1	0
House	∞	1	0
Buttercup Flower	∞	1	0

successfully decrypts the cipher image in its entirety, indicating that the decryption effect is flawless. Image quality analysis through PSNR, SSIM, and MSE is shown in Table 14.

4.7. *Robustness analysis*

During the process of transmission, the image will be impacted by a number of different elements that cannot be avoided. In the communication system, for instance, noise may lead to unfavorable consequences such as distortion, deterioration, and pollution. Deciphering the noisy cipher text in order to reconstruct the original image is an additional challenge that must be overcome. As a consequence of this, any method that is employed to encrypt images should be adequately resistant to withstand attacks based on noise. During simulation, we utilise the 256 × 256 Pepper image to practice several degrees of cropping and noise attacks.

4.7.1. *Noise attack*

When evaluating the effectiveness of encryption schemes, anti-noise capability is an essential factor to take into account. The concept can be displayed in contexts of the following equation:

$$E_I{}' = E_I + KN \tag{17}$$

where E_I is the noise-free encrypted image and $\mathrm{E_I}'$ is noisy encrypted image, N represents supplemental noise, and K is the noise intensity constant. The encrypted image is also affected by noise. The following is an illustration of the notion that might be used in the event that additional noise is responsible for the destruction of the encrypted image. In this investigation, Salt and Pepper Noise attacks are used, and the noise intensity K for the suggested encryption method is varied between 0.01, 0.05, 0.001, and 0.005 respectively. Figure 11 displays the encrypted text visuals together with the decoded images that correspond to those images under various noise attacks and intensities. Even if the quality of the decrypted image is worse as the level of the noise gets higher, the technique can nevertheless withstand noise attacks across a larger spectrum of intensities. In Table 15 PSNR values of noisy encrypted images on noise attacks with different intensities are shown. As a direct consequence of this, the proposed method of encryption is more resistant against attacks that are based on salt and pepper noise.

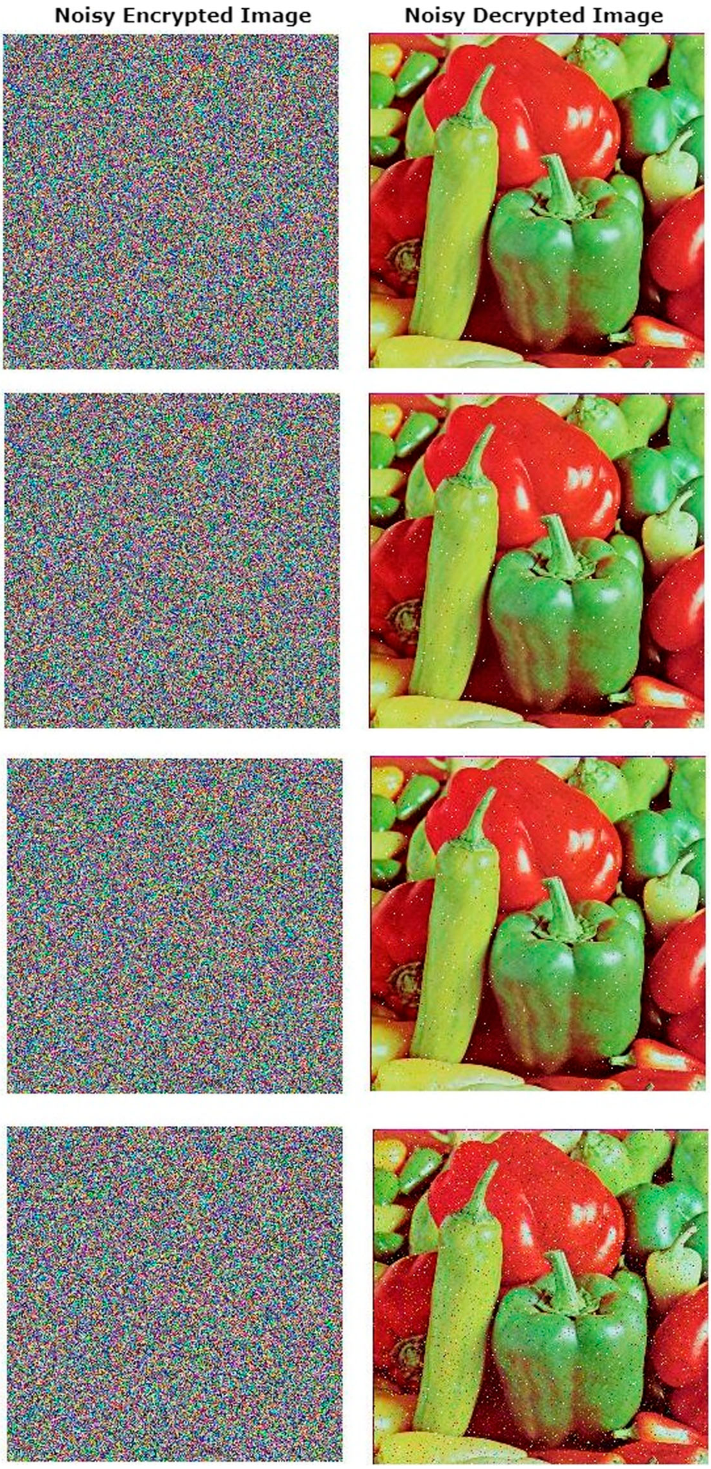


Figure 11. Noise attack on Pepper image.

Table 15. Noise attacks with different intensities.

Image	Noise density level	PSNR of the noisy encrypted image
Lena	0.001	28.476162
	0.005	26.940204
	0.01	25.556459
	0.05	20.633339

Table 16. Cropping attacks with different data loss pixel areas.

Image	Data loss of pixels area	PSNR of the cropped encrypted image
Lena	32 × 32 pixels area	32.327711
	64 × 64 pixels area	26.226746
	96 × 96 pixels area	22.739982
	128 × 128 pixels area	20.199788

4.7.2. Cropping attack

While communicating with image data, there is a high probability that some of the image data will be lost. The image that has been encrypted, using the suggested encryption technique, kept all the vital information intact. To infer that a section of the encrypted image has been removed during cropping, we might suppose that the corresponding pixel is blank. The encrypted image and its matching decoded image are shown side by side in Figure 12, after being cropped to remove data from areas of 32 × 32 pixels, 64 × 64 pixels, 96 × 96 pixels, and 128 × 128 pixels, respectively. In Table 16, PSNR values of cropped encrypted images on cropping attacks with different pixels data loss are shown. Even while the decrypted image will become hazier when the cutting rate is increased, the essential details of the original image will still be visible that shows attacks may be easily avoided using the encryption scheme that has been presented.

4.8. Computation time and speed analysis

In the context of security, computation time is also an important aspect. The time of the suggested encryption system is evaluated for colored images. Table 17 depicts the outcomes of the observation of computational time of encryption process with different images. Time complexity of the proposed algorithm is calculated as Θ (4MN).

In this performance comparison, we not only compare the time cost, but also present the discussion of other factors, such as the operating system, the hardware environment, the programming language, and key space. In order to gain a better image performance evaluation, we also compare encryption throughput (ET) and number of cycles (NC) in Table 18. On comparison we observe that ET and cycle count results are not very promising but the key space is far better and the computation time is also satisfactory. In the future, improved speed performance with appropriate keyspace may be worked on.

4.9. Verification of performance of classification through transfer learning

In this part of the article, the performance of the classification is evaluated by deep learning classification utilising transfer learning. The original, encrypted, and decrypted versions of the iris images are shown in Figure 13. These images are then put through further testing

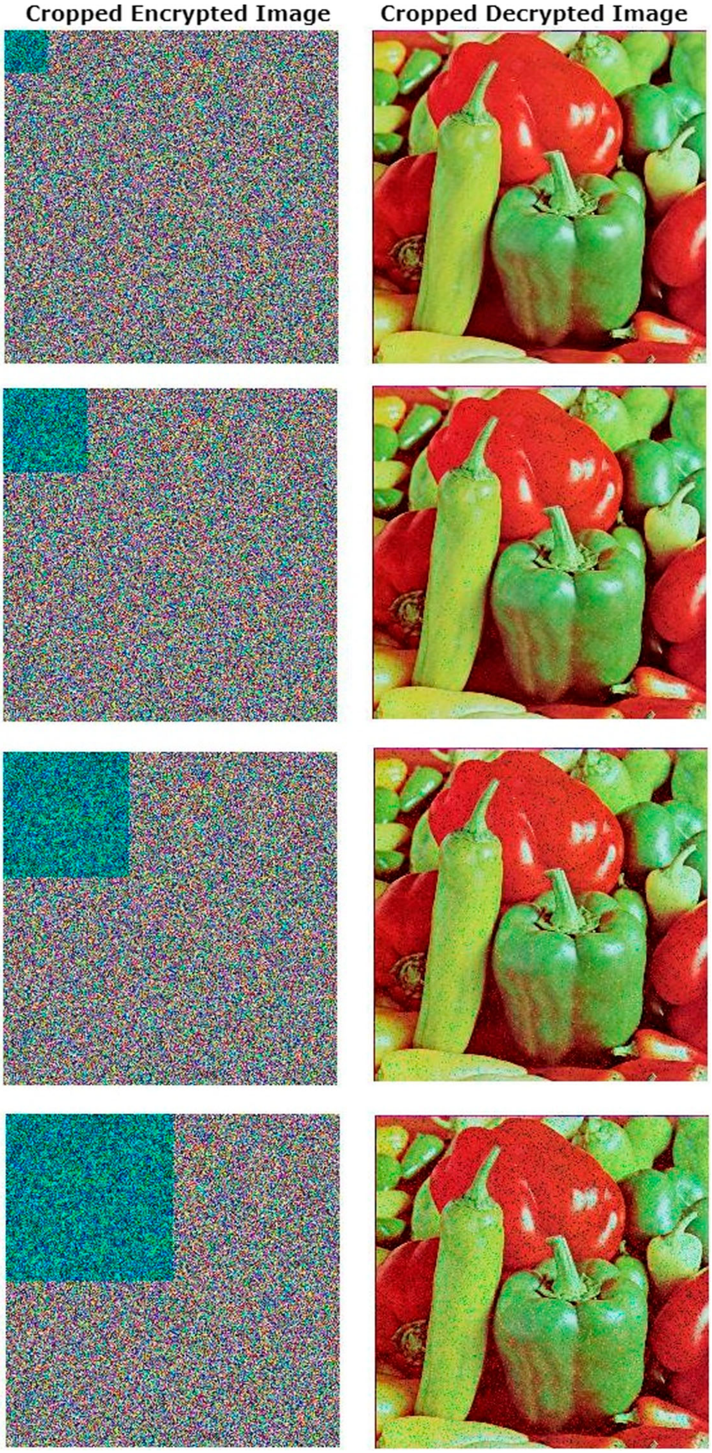


Figure 12. Cropping attack on Pepper image.

Table 17. Computational time for the proposed algorithm on different test images.

Images	Computational time (s)
Baboon	0.3221
Lena	0.3021
Iris Flower	0.3234
Pepper	0.3221
Aeroplane	0.3042
Tree	0.3025
House	0.3241
Buttercup Flower	0.3123

Table 18. Comparison of computational time and Speed analysis of encryption process in different literatures.

Methods	Image	CPU (GHz)	Language	Time	Keyspace	ET	NC
Proposed	256 × 256	1.6	MATLAB	0.322	2^{847}	0.115	13,853
Ref (Cun et al., 2021)	512 × 512	3	MATLAB	N/A	2^{231}	0.170	16,830
Ref (Xian et al., 2020)	256 × 256	3.2	MATLAB	N/A	2^{156}	0.275	11,089
Ref (Li et al., 2021)	512 × 512	1.4	MATLAB	0.138	2^{455}	1.811	1682
Ref (Shahna & Mohamed, 2021)	256 × 256	2.3	MATLAB	0.2410	2^{384}	N/A	N/A
Ref (Bhat et al., 2022)	512 × 512	1.80	MATLAB	0.70	N/A	N/A	N/A
Ref (Rahman et al., 2022)	512 × 512	N/A	MATLAB	0.45	2^{744}	N/A	N/A
Ref (Abduljabbar et al., 2022)	256 × 256	2.6	MATLAB	0.3493	2^{430}	N/A	N/A
Ref (Qian et al., 2021)	256 × 256	N/A	N/A	0.8314	2^{600}	N/A	N/A

N/A-not available.

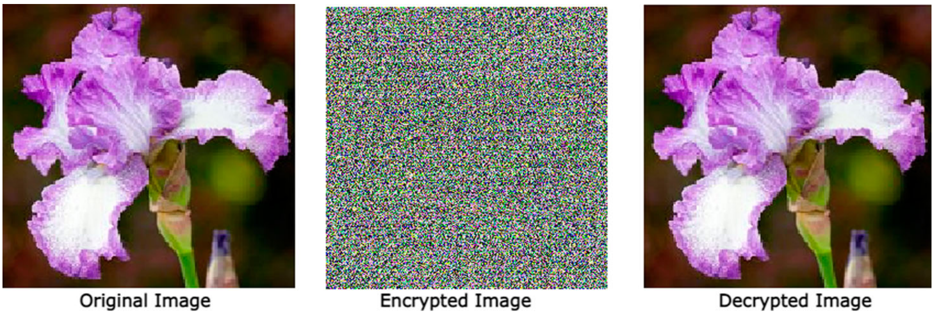


Figure 13. Original, encrypted, and decrypted Iris image.

to determine how accurately they can be classified. In this case, the simulation makes use of the AlaxaNet framework of transfer learning that is implemented on the deep learning designer of MATLAB 2021.

The categorisation of the iris image both before and after it was encrypted is shown in Figure 14. In addition, it is clear from the findings that the suggested approach successfully encrypted the images, and that the encrypted images were correctly categorised. The graph illustrating the relationship between accuracy and iteration for the iris image is shown in Figure 15. The accuracy of the iris image for the purpose of validation is 91.35%.

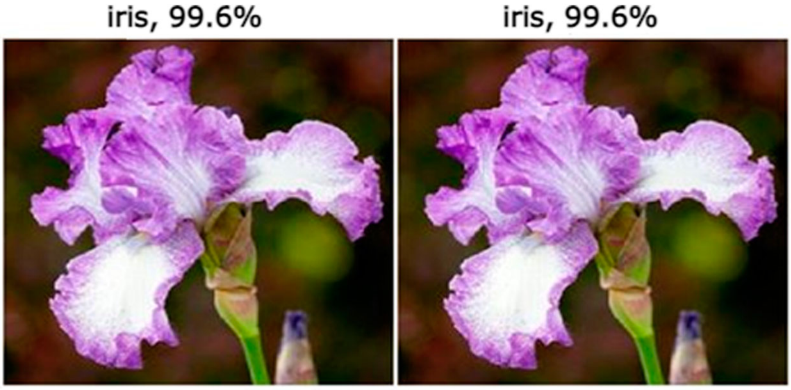


Figure 14. Iris image classification before and after encryption.

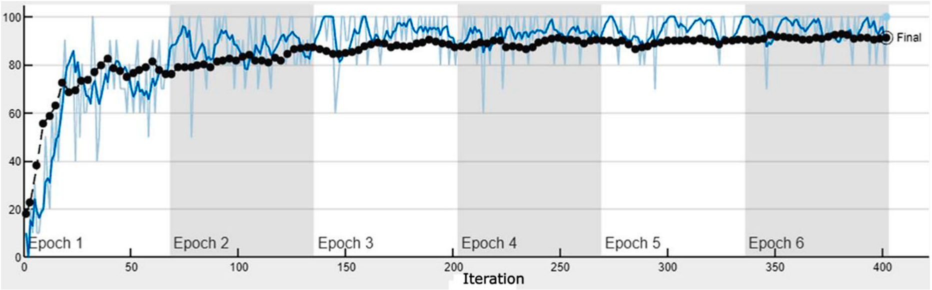


Figure 15. Accuracy versus iteration graph for the Iris image.

5. Conclusion

The Lorenz-Gauss-Logistic (LGL) encryption technique is developed and demonstrated in this work. The random key is created by the high dimensional Lorenz system, which is then utilised to generate a randomly generated numeric pattern for the controlling parameters. This pattern has a greater degree of unpredictability. Following this, the 5D Gauss-Logistic Hyperchaotic system is chosen to act as the principal Hyperchaotic map technique. Numerous security studies and the method's visual effect on decryption demonstrate the algorithm's superiority and robustness in comparison to competing algorithms. The visual examination of their most essential properties, such as the sensitivity of the beginning value of both maps and the Lyapunov exponent of 5D Gauss Logistic map, is carried out which proves the suitability of both the maps for encryption application. The simulations found that the LGL cryptographic system's pixel correlation attained a range of −0.0019, −0.0016, and −0.0069, as well as 7.9996 information entropy, indicating that the HDIEA's encryption approach had a significant scattering effect. Further the algorithm's UACI and NPCR scores are so high (respectively at 99.63% and 33.35%), this also suggests that it does an exceptional encryption performance. Also the recovered image is found identical to the original image which shows the accuracy of structural similarity. The visual decryption effect of the method as well as the visible results of numerous different anti-attack tests demonstrates that the method has a high anti-attack strength and is very resilient in comparison to other algorithms.

High-dimensional chaotic maps feature a greater number of variables or parameters, resulting in a wider chaotic space. However, such a complicated dynamical system may be challenging to build for real-time applications. Although this difficulty might be overcome by using compression methods to create a lightweight system.

We will continue to investigate and enhance the suggested model and algorithm in the future to create it more lightweight cryptosystem. As for the next version of this cipher is concerned, we plan to implement preferred encryption for the video surveillance problem.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Bharti Ahuja http://orcid.org/0000-0003-2978-6310
Sharad Salunke http://orcid.org/0000-0002-8452-5597
Aditya Gupta http://orcid.org/0000-0003-3128-2517

References

Abduljabbar, Z. A., Abduljaleel, I. Q., Ma, J., Al Sibahee, M. A., Nyangaresi, V. O., Honi, D. G., Abdulsada, A. I., & Jiao, X. (2022). Provably secure and fast color image encryption algorithm based on S-boxes and hyperchaotic map. *IEEE Access*, *10*, 26257–26270. https://doi.org/10.1109/ACCESS.2022.3151174.

Abdullah, H. A., & Abdullah, H. N. (2019). FPGA implementation of color image encryption using a new chaotic map. *Indonesian Journal of Electrical Engineering and Computer Science*, *13*(1), 129–137. https://doi.org/10.11591/ijeecs.v13.i1.pp129-137.

Ahmad, J., & Hwang, S. O. (2016). A secure image encryption scheme based on chaotic maps and affine transformation. *Multimedia Tools and Applications*, *75*(21), 13951–13976. https://doi.org/10.1007/s11042-015-2973-y.

Ahuja, B., & Doriya, R. (2021). A novel hybrid compressive encryption cryptosystem based on block quarter compression via DCT and fractional Fourier transform with chaos. *International Journal of Information Technology (Singapore)*, *13*(5), 1837–1846. https://doi.org/10.1007/s41870-021-00759-y

Ahuja, B., & Doriya, R. (2022). Bifold-crypto-chaotic steganography for visual data security. *International Journal of Information Technology (Singapore)*, *14*(2), 637–648.

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

Bhat, J., Saqib, M., & Moon, A. H. (2022). Fuzzy extractor and chaos enhanced elliptic curve cryptography for image encryption and authentication. *International Journal of System Assurance Engineering and Management*, *13*(2), 697–712. https://doi.org/10.1007/s13198-021-01330-5

Bisht, A., Dua, M., Dua, S., & Jaroli, P. (2020). A color image encryption technique based on Bit-level permutation and alternate logistic maps. *Journal of Intelligent Systems*, *29*(1), 1246–1260. https://doi.org/10.1515/jisys-2018-0365.

Chai, X., Fu, X., Gan, Z., Lu, Y., & Chen, Y. (2019). A color image cryptosystem based on dynamic DNA encryption and chaos. *Signal Processing*, *155*, 44–62. https://doi.org/10.1016/j.sigpro.2018.09.029.

Chai, X., Wu, H., Gan, Z., Han, D., Zhang, Y., & Chen, Y. (2021). An efficient approach for encrypting double color images into a visually meaningful cipher image using 2D compressive sensing. *Information Sciences*, *556*, 305–340. https://doi.org/10.1016/j.ins.2020.10.007

Chen, J., Chen, L., & Zhou, Y. (2021). Cryptanalysis of image ciphers with permutation-substitution network and chaos. *IEEE Transactions on Circuits and Systems for Video Technology*, *31*(6), 2494–2508. https://doi.org/10.1109/TCSVT.2020.3021908.

Cun, Q., Tong, X., Wang, Z., & Zhang, M. (2021). Selective image encryption method based on dynamic DNA coding and new chaotic map. *Optik*, *243*(April), 167286. https://doi.org/10.1016/j.ijleo.2021.167286

Dhopavkar, T. A., Nayak, S. K., & Roy, S. (2022). IETD: A novel image encryption technique using tinkerbell map and duffing map for IoT applications. *Multimedia Tools and Applications*, *81*(30), 43189–43228.

Elghandour, A., Salah, A., & Karawia, A. (2022). A new cryptographic algorithm via a two-dimensional chaotic map. *Ain Shams Engineering Journal*, *13*(1), 101489. https://doi.org/10.1016/j.asej.2021.05.004.

Es-Sabry, M., Akkad, N. E., Merras, M., Saaidi, A., & Satori, K. (2022). A new color image encryption algorithm using multiple chaotic maps with the intersecting planes method. *Scientific African*, *16*, e01217. https://doi.org/10.1016/j.sciaf.2022.e01217.

Feixiang, Z., Mingzhe, L., Kun, W., & Hong, Z. (2021). Color image encryption via Hénon-Zigzag map and chaotic restricted Boltzmann machine over blockchain. *Optics and Laser Technology*, *135*, 106610. https://doi.org/10.1016/j.optlastec.2020.106610.

Ferdush, J., Begum, M., & Uddin, M. S. (2021). Chaotic lightweight cryptosystem for image encryption. *Advances in Multimedia*, 1–16. https://doi.org/10.1155/2021/5527295.

Fridrich, J. (1998). Symmetric ciphers based on Two-dimensional chaotic maps. *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, *8*(6), 1259–1284.

Fu, C., Zheng, Y., Chen, M., & Wen, Z. K. (2017, October 27). *A color image encryption algorithm using a new 1-D chaotic map*. International Conference on Communication Technology Proceedings, ICCT, Chengdu, China. https://doi.org/10.1109/ICCT.2017.8359933.

Girdhar, A., & Kumar, V. (2018). A RGB image encryption technique using Lorenz and Rossler chaotic system on DNA sequences. *Multimedia Tools and Applications*, *77*(20), 27017–27039. https://doi.org/10.1007/s11042-018-5902-z.

Goufo, D., & Franc, E. (2019). On chaotic models with hidden attractors in fractional calculus above power law. *Chaos, Solitons and Fractals*, *127*, 24–30. https://doi.org/10.1016/j.chaos.2019.06.025.

Gupta, N., & Vijay, R. (2022). Hybrid image compression-encryption scheme based on multilayer stacked autoencoder and logistic map. *China Communications*, *19*(1)), 238–252. https://doi.org/10.23919/JCC.2022.01.017.

Haroun, M. F., & Aaron Gulliver, T. (2015). Real-time image encryption using a low-complexity discrete 3D dual chaotic cipher. *Nonlinear Dynamics*, *82*(3), 1523–1535 https://doi.org/10.1007/s11071-015-2258-z.

Hosny, K. M., Kamal, S. T., & Darwish, M. M. (2021). A color image encryption technique using block scrambling and chaos. *Multimedia Tools and Applications*, *81*, 505–525.

Jarjar, M., Hraoui, S., Najah, S., & Zenkouar, K. (2022). New technology of color image encryption based on chaos and two improved vigenère steps. *Multimedia Tools and Applications*, *81*(17), 24665–24689. https://doi.org/10.1007/s11042-022-12750-1

Jawad, L. M. (2021). A new scan pattern method for color image encryption based on 3D-Lorenzo chaotic map method. *Multimedia Tools and Applications*, *80*(24), 33297–33312 https://doi.org/10.1007/s11042-021-11295-z.

Kaur, R., & Singh, B. (2021). A novel approach for data hiding based on combined application of discrete cosine transform and coupled chaotic map. *Multimedia Tools and Applications*, *80*(10), 14665–14691. https://doi.org/10.1007/s11042-021-10528-5.

Khade, P. N., & Narnaware, P. M. (2012). 3D chaotic functions for image encryption. *International Journal of Computer Science Issues*, *9*(3), 323–328.

Khalil, N., Sarhan, A., & Alshewimy, M. A. M. (2021). An efficient color/grayscale image encryption scheme based on hybrid chaotic maps. *Optics and Laser Technology*, *143*, 107326. https://doi.org/10.1016/j.optlastec.2021.107326.

Khedmati, Y., Parvaz, R., & Behroo, Y. (2020). 2D hybrid chaos map for image security transform based on framelet and cellular automata. *Information Sciences*, *512*, 855–879. https://doi.org/10.1016/j.ins.2019.10.028.

Koyuncu, İ., Alçin, M., Tuna, M., Pehlivan, İ., Varan, M., & Vaidyanathan, S. (2019). Real-time high-speed 5-D hyperchaotic lorenz system on FPGA. *International Journal of Computer Applications in Technology*, *61*(3), 152–165. https://doi.org/10.1504/IJCAT.2019.102852

Li, N., Xie, S., & Zhang, J. (2022). A color image encryption algorithm based on double fractional order chaotic neural network and convolution operation. *Entropy*, *24*(7), 933. https://doi.org/10.3390/e24070933

Li, P., Xu, J., Mou, J., & Yang, F. (2019). Fractional-order 4D hyperchaotic memristive system and application in color image encryption. *Eurasip Journal on Image and Video Processing*, *2019*(22). https://doi.org/10.1186/s13640-018-0402-7

Li, Z., Peng, C., Tan, W., & Li, L. (2021). An effective chaos-based image encryption scheme using imitating Jigsaw method. *Complexity*, *2021*, 1–18. https://doi.org/10.1155/2021/8824915

Lin, R., & Li, S. (2021). An image encryption scheme based on Lorenz hyperchaotic system and RSA algorithm. *Security and Communication Networks*, *2021*. https://doi.org/10.1155/2021/5586959.

Liu, C., & Ding, Q. (2020). A color image encryption scheme based on a novel 3D chaotic mapping. *Complexity*, *2020* https://doi.org/10.1155/2020/3837209.

Liu, J., Tong, X., Wang, Z., Ma, J., & Yi, L. (2019a). An improved Rao-Nam cryptosystem based on fractional order hyperchaotic system and EDF-QC-LDPC. *International Journal of Bifurcation and Chaos*, *29*(9). https://doi.org/10.1142/S0218127419501220.

Liu, L., Du, C., Zhang, X., Li, J., & Shi, S. (2019b). Dynamics and entropy analysis for a new 4-D hyperchaotic system with coexisting hidden attractors. *Entropy*, *21*(3). https://doi.org/10.3390/e21030287.

Liu, X., Xiao, D., & Liu, C. (2020). Quantum image encryption algorithm based on bit-plane permutation and sine logistic map. *Quantum Information Processing*, *19*(8). https://doi.org/10.1007/s11128-020-02739-w

Liu, Y., Cen, G., Xu, B., & Wang, X. (2022). Color image encryption based on deep learning and block embedding. *Security and Communication Networks*, *2022*(2), 1–14. https://doi.org/10.1155/2022/6047349

Nilsback, M.-E., & Zisserman, A. (n.d). 17 Category flower dataset. https://www.robots.ox.ac.uk/ vgg/data/flowers/17/

Ping, P., Xu, F., Mao, Y., & Wang, Z. (2018). Designing permutation–substitution image encryption networks with Henon map. *Neurocomputing*, *283*, 53–63. https://doi.org/10.1016/j.neucom.2017.12.048.

Qian, X., Yang, Q., Li, Q., Liu, Q., Wu, Y., & Wang, W. (2021). A novel color image encryption algorithm based on three-dimensional chaotic maps and reconstruction techniques. *IEEE Access*, *9*, 61334–61345. https://doi.org/10.1109/ACCESS.2021.3073514

Rahman, Z.-A. S., Jasim, B. H., Al-Yasir, Y. I., & Abd-Alhameed, R. A. (2022). Efficient colour image encryption algorithm using a new fractional-order memcapacitive hyperchaotic system. *Electronics (Switzerland)*, *11*(9). https://doi.org/10.3390/electronics11091505

Rahmawati, W. M., & Liantoni, F. (2018, September 29). *Image compression and encryption using DCT and Gaussian map*. IOP Conference Series: Materials Science and Engineering, Surabaya, Indonesia.

Ravichandran, D., Praveenkumar, P., Rayappan, J. B. B., & Amirtharajan, R. (2016). Chaos based crossover and mutation for securing DICOM image. *Computers in Biology and Medicine*, *72*, 170–184. https://doi.org/10.1016/j.compbiomed.2016.03.020.

Shahna, K. U., & Mohamed, A. (2021). Novel hyper chaotic color image encryption based on pixel and Bit level scrambling with diffusion. *Signal Processing: Image Communication*, *99*, 116495. https://doi.org/10.1016/j.image.2021.116495.

"SIPI Image Database". (n.d). http://sipi.usc.edu/database/database.php

Song, W., Fu, C., Tie, M., Sham, C.-W., Liu, J., & Ma, H.-f. (2022a). A fast parallel batch image encryption algorithm using intrinsic properties of chaos. *Signal Processing: Image Communication*, *102*. https://doi.org/10.1016/j.image.2021.116628.

Song, W., Fu, C., Zheng, Y., Cao, L., Tie, M., & Sham, C.-W. (2022b). Protection of image ROI using chaos-based encryption and DCNN-based object detection. *Neural Computing and Applications*, *34*(7), 5743–5756. https://doi.org/10.1007/s00521-021-06725-w.

Song, W., Fu, C., Zheng, Y., Tie, M., Liu, J., & Chen, J. (2023). A parallel image encryption algorithm using intra bitplane scrambling. *Mathematics and Computers in Simulation*, *204*, 71–88. https://doi.org/10.1016/j.matcom.2022.07.029

Song, W., Zheng, Y., Fu, C., & Shan, P. (2020). A novel batch image encryption algorithm using parallel computing. *Information Sciences*, *518*, 211–224. https://doi.org/10.1016/j.ins.2020.01.009.

Su, Y., & Wang, X. (2022). Characteristic analysis of new four-dimensional autonomous power system and Its application in color image encryption. *Multimedia Systems*, *28*, 553–571. https://doi.org/10.1007/s00530-021-00861-y.

Tang, M., Zeng, G., Yang, Y., & Chen, J. (2022). A hyperchaotic image encryption scheme based on the triple dislocation of the Liu and Lorenz system. *Optik*, *261*(April), 169133. https://doi.org/10.1016/j.ijleo.2022.169133

ul Haq, T., & Shah, T. (2021). 4D mixed chaotic system and Its application to RGB image encryption using substitution-diffusion. *Journal of Information Security and Applications*, *61*, 102931. https://doi.org/10.1016/j.jisa.2021.102931.

Veena, G., & Ramakrishna, M. (2021). A survey on image encryption using chaos-based techniques. *International Journal of Advanced Computer Science and Applications*, *12*(1). https://doi.org/10.14569/IJACSA.2021.0120145

Wang, S., Peng, Q., & Du, B. (2022). Chaotic color image encryption based on 4D chaotic maps and DNA sequence. *Optics and Laser Technology*, *148*. https://doi.org/10.1016/j.optlastec.2021.107753.

Wang, X., Feng, L., & Zhao, H. (2019). Fast image encryption algorithm based on parallel computing system. *Information Sciences*, *486*, 340–358. https://doi.org/10.1016/j.ins.2019.02.049.

Wang, X., & Gao, S. (2020). Image encryption algorithm for synchronously updating boolean networks based on matrix semi-tensor product theory. *Information Sciences*, *507*, 16–36. https://doi.org/10.1016/j.ins.2019.08.041.

Wang, X., & Yang, J. (2021). Spatiotemporal chaos in multiple coupled mapping lattices with multi-dynamic coupling coefficient and Its application in color image encryption. *Chaos, Solitons and Fractals*, *147*, 110970. https://doi.org/10.1016/j.chaos.2021.110970.

Wu, X., Wang, K., Wang, X., Kan, H., & Kurths, J. (2018). Color image DNA encryption using NCA map-based CML and one-time keys. *Signal Processing*, *148*, 272–287. https://doi.org/10.1016/j.sigpro.2018.02.028.

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

Xian, Y., Wang, X., Yan, X., Li, Q., & Wang, X. (2020). Image encryption based on chaotic Sub-block scrambling and chaotic digit selection diffusion. *Optics and Lasers in Engineering*, *134*, 106202. https://doi.org/10.1016/j.optlaseng.2020.106202.

Xu, D., Li, G., Xu, W., & Wei, C. (2022). Design of artificial intelligence image encryption algorithm based on hyperchaos. *Ain Shams Engineering Journal*, *14*(3), 101891. https://doi.org/10.1016/j.asej.2022.101891.

Yan, S., Li, L., Gu, B., Cui, Y., Wang, J., & Song, J. (2023). Design of hyperchaotic system based on multi-scroll and its encryption algorithm in color image. *Integration (Tokyo, Japan)*, *88*(October 2022), 203–221. https://doi.org/10.1016/j.vlsi.2022.10.002

Yang, F., Mou, J., Sun, K., & Chu, R. (2020). Lossless image compression-encryption algorithm based on BP neural network and chaotic system. *Multimedia Tools and Applications*, *79*(27-28), 19963–19992. https://doi.org/10.1007/s11042-020-08821-w.

Yavuz, E. (2019). A novel chaotic image encryption algorithm based on content-sensitive dynamic function switching scheme. *Optics and Laser Technology*, *114*, 224–239. https://doi.org/10.1016/j.optlastec.2019.01.043.

Zefreh, E. Z. (2020). An image encryption scheme based on a hybrid model of DNA computing, chaotic systems and hash functions. *Multimedia Tools and Applications*, *79*(33-34), 24993–25022. https://doi.org/10.1007/s11042-020-09111-1.

Zhang, Y. Q., He, Y., Li, P., & Wang, X. Y. (2020). A new color image encryption scheme based on 2DNLCML system and genetic operations. *Optics and Lasers in Engineering*, *128*. https://doi.org/10.1016/j.optlaseng.2020.106040.

Zheng, Y., Tian, H., Du, M., & Fu, C. (2022). *Encrypted video search: Scalable, modular, a nd content-similar*. Proceedings of the 13th ACM Multimedia systems Conference, 14-17 June 2022, 177–190.