Palestine Journal of Mathematics

Vol.15(3), (2026) , 677–696 © Palestine Polytechnic University-PPU 2026

A NOVEL AUTHENTICATION BASED ENCRYPTION SCHEME USING ELGAMAL CRYPTOSYSTEM AND CHAOTIC MAPS

A. Paul, R. Singh, M. Rao, M. K. Patel and D. Singh

Communicated by Manoj Kumar Patel

MSC 2010 Classifications: Primary 68P25; Secondary 94A08.

Keywords and phrases: Encryption, Decryption, Henon map, ElGamal cryptosystem, Arnold cat map, Digital Signature.

The authors would like to thank the reviewers and editor for their constructive comments and valuable suggestions that improved the quality of our paper.

Corresponding Author: D. Singh

Abstract In this study, a color image is encrypted by utilizing the Henon map, Arnold cat map, and ElGamal digital signature. Firstly, the original image is partially encrypted by applying the XOR operation with the first key sequence produced from the Henon map, to obtain the partially encrypted image (PEI). Then, the digital signature is acquired from the PEI with the help of SHA256 and the ElGamal public cryptosystem. Further, PEI is re-encrypted by applying XOR operation with the second key sequence produced by utilizing the Henon map, and subsequently, a digital signature is appended to it. Finally, the image containing the digital signature is encrypted by employing Arnold cat map. The digital signature is utilized to confirm the legitimacy of the image data, and encryption provides the safety to the image information from illegal access. The experimental findings of this study shows the effectiveness and dependability of the recommended approach concerning noise and occlusion attacks. Also, the encryption scheme's resilience to statistical attacks is suggested by the analysis of the histogram, information entropy and correlation coefficient. The encrypted data achieves a high entropy of 7.99, indicating a near-ideal value of 8 of randomness, and a low correlation coefficient value close to zero. Further, average NPCR value of 99.8%, and average UACI value of 33.3%, reflecting its strong capability to resist the differential attacks.

1 Introduction

Technology has its advantages and pitfalls in modern times. The major disadvantage involves the unethical incursion over the open channels to sneak the valuable information transmitted by the sender. Various encryption schemes [16, 14, 30, 2, 20, 19, 21, 17, 22, 9, 10, 18] have been presented to minimize the leakage of information over open channels. The safety of information has increasingly become critical with technological advancement, since the communication of digital information over the internet occurs frequently. In particular, the security of images is a major concern because of their significant application in different fields, including defense, education, and administration. Therefore, cryptographic algorithms are needed to conquer these problems.

Cryptographic algorithms can be divided into two types, namely private and public. The private cryptographic algorithms operate the same key for the originator and receiver involved in communication, while the public cryptographic algorithm employs different keys at both ends. The organizations are leading towards electronic files to lessen the paperwork. To protect electronic files from potential hackers, there is a need for strong cryptographic algorithms, which can protect the information. So, the purpose of designing and implementing image encryption and decryption algorithms is to preserve confidentiality and safeguard the communication of digital

information.
In chaos theory [16, 17, 24, 34, 36], confusion and diffusion [11, 3, 12, 8, 31, 28, 37] are two customary phases that any cryptographic algorithm must pass to improve its appearance. Pixel confusion or scrambling is achieved by arbitrarily shifting pixel locations without altering pixel intensity. However, it is deficient in offering utter reliability for image data. On this ground, the diffusion state is quite essential. In this step, a variety of operations or substitutions are assembled to modify the values of the pixels.

The transforms, such as wavelet, cosine, Hartley, and Fourier, are used to create intricate diffusion within image pixels. While the maps, such as skew maps, logistic map, tent map, and Henon map, could be utilized in either way to attain confusion and diffusion. The algorithms established on the chaos theory are unpredictable, sensitive to initial values and parameters, and pseudo-random. Within chaotic maps, even a modest manipulation of the parameters or preliminary conditions could result in unpredictable sequences. Therefore, if an illegal individual attempts to enter and is unaware of the exact key, he is unable to determine the chaos sequence. Thus, it is better to use chaos-based encryption.

Encryption can safeguard data privacy, but there is still a need to protect the authenticity and reliability of data. To test the authenticity of the data on public channels, digital signatures are used in combination with the chaotic maps. To ensure data authenticity over public channels, [1] and [25] proposed encryption schemes that incorporate a digital signature. It has many benefits like, relative ease of laying down the signature's authenticity, the difficulty of modifying the signature, the difficulty of creating a signature, and the non-repudiation, which ensures that the signer cannot subsequently reject the signature. To prevent rejection and counterfeiting, the signature needs to use certain data that is specific to the originator.

Many authors have proposed various encryption methods for image security using public and private key cryptography. [25] developed an encryption algorithm for digital images by using a digital signature. They evaluated the hash of the original image by employing a hash function that generated a message digest of a fixed length of 128 bits. Then, this message digest is encrypted with the help of the RSA (Rivest-Shamir-Adleman) public crypto-system to obtain a digital signature. Moreover, they added the digital signature of the input image to the coded image, which is encoded by an error control code like the Bose-Chaudhuri-Hochquenghem (BCH) code. This approach is employed to verify the authenticity and integrity of the decoded image. Also, [1] formulated an encryption methodology for image authentication with the help of a digital signature. They had taken the hash of the original image by using SHA-1 and then encrypted it by employing the RSA public crypto-system. The digital signature thus obtained is concealed in this image and is sent along with the image. [4] designed an algorithm that is grounded on pixel confusion and pixel diffusion. In the pixel confusion step, they randomized pixels by utilizing the Gingerbreadman chaotic map, while in the diffusion step, they used bit-wise XOR operation by employing the key produced by using RC4A stream cipher. [5] introduced a method that was built on the Diffie-Hellman key exchange method, the fractional discrete cosine transform, and the Arnold transform. In this scheme, they changed the pixel values and pixel positions with the help of a combination of these three maps. Further, [21] outlined an encryption algorithm by adopting two chaotic maps, the Arnold cat map and the Zaslavskii map. In this method, the intensity of the image pixels is modified by operating XOR between the image pixels and pseudo-random numbers, produced by using the Zaslavskii map. [29] has proposed an encryption scheme by incorporating the RSA cryptosystem and a modified random matrix affine cipher. They have used Chen's hyperchaotic system and Arnold cat map for creating a distraction between the original and ciphered image. Recently, [23] has developed a new image encryption scheme that used Vigenère cihper, Henon map, Arnold cat map, and discretized Baker map for the protection of the image data. But, these techniques [29, 23] can only encrypt the image data.

In the proposed scheme, SHA-256 and the Elgamal public key cryptosystem are employed to generate a digital signature, ensuring the authenticity of the received image. Firstly, we take the original image and scramble it by using the matrix key produced from the Henon map. Subsequently, the SHA256 hash function is employed to generate a hash value for the scrambled

image. Further, the hash value is encrypted by utilizing the ElGamal public cryptosystem to produce the digital signature. Then, the modified image is further encrypted by employing the second matrix key to get the PEI. By repeating the 32 bytes of digital signature, we generate a digital signature and append it to the PEI image in row form. To keep the image square, attach one column vector of random numbers to this image. Finally, to distribute the digital signature across the entire image, we will apply the Arnold cat map on the image containing the digital signature. The experimental findings of this study show that the recommended strategy is reliable against noise and cropping attacks. Further, information entropy, correlation coefficient, NPCR, histogram analysis, and UACI values demonstrate how resilient the encryption system is to statistical assaults.

In this paper, Section 2 gives basic theories used in this research. Section 3 explains the mechanism of encryption, Section 5 contains the security analysis, and Section 6 concludes about the proposed encryption technique.

2 Background theories

2.1 2D Henon Map

Pseudo-random sequences are produced by means of chaotic maps. Because they are very sensitive to the initial conditions, even small alterations to the starting values can lead to significant variations in the output, making these functions unpredictable. One of the popular chaotic functions in scientific literature is the Henon map [33, 35], which produces two pseudo-random value sequences on each run. The standard form of the 2D Henon map is described by the equations given below

$$\begin{aligned} U(n+1) &= 1 + V(n) - a(U^2(n)) \\ V(n+1) &= b(U^2(n)), \end{aligned} \tag{2.1}$$

by taking a = 1.4, b = 0.3, $U_o = 0.1$, and $V_o = 0.1$, the graph for 256 iterations of the Henon map is displayed in Fig. 1.

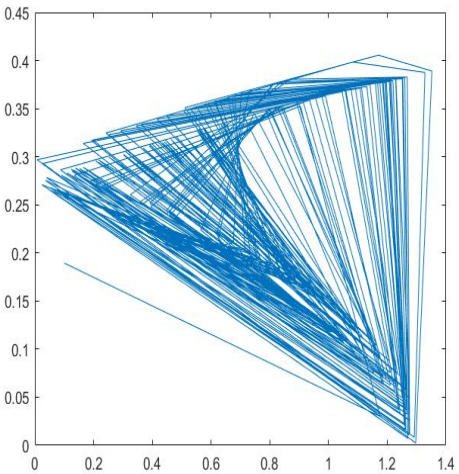


Figure 1: Bifurcation diagram for Henon map

2.2 2D Arnold Cat Map

Arnold (1960) created a two-dimensional map using a picture of a cat [19, 21, 15, 26, 27, 13]. This method offers the basic idea of iteratively moving the locations of the pixels over time. Nevertheless, it was discovered that the initial state would reappear after several iterations. It

is known as the Arnold period, and it changes according to the size of the image. This technique illustrates the chaotic map's propensity for mixing. The general expression for the two-dimensional Arnold's cat map is given by

$$x_{i+1} \equiv mod(x_i + uy_i, M), \tag{2.2}$$

$$y_{i+1} \equiv mod(vx_i + (uv + 1)y_i, N), \tag{2.3}$$

where the numbers of rows and columns are given by M and N, respectively, and u and v are the control variables. In each repetition, this expression uses the coordinate (x_i, y_i) of the image pixel to move it to a new position $(x_i + 1, y_i + 1)$.

2.3 Secure Hash Algorithm (SHA)

SHA is a function that takes any document D as input and gives a short bit string H as the output, like SHA256, which returns the output as 256 bits for any document [6]. There are some properties that make hash functions more secure for information encoding. First property is linear time, that is, the computation of hash for any document is very fast and easy, second is inversion free, that is, the inverse of the hash function does not exist, and third is collision resistant, that is, there does not exist two different documents whose hash is the same. There are different hash functions which can achieve various levels of security, such as SHA-1, SHA-2 family (for example, SHA-512).

2.4 ElGamal Public Cryptosystem

ElGamal public cryptosystem [7], which produces distinct encrypted messages for the identical plain-text on consecutive operations, is considered to be among the most reliable unpredictable algorithms. The foundation of it is the Diffie-Hellman key exchange procedure [5], but its safety depends primarily on the solution of the discrete logarithmic problem. The general structure of the ElGamal public cryptosystem is described below:

Choose a prime P, and its primitive root g. Consider a number a as private key and compute public key A by using following

$$A \equiv mod(g^a, P). \tag{2.4}$$

Now to encrypt plain-text, firstly convert it into an integer between 2 and P and choose a random ephemeral key k. The encryption is done by using the following

$$c_1 \equiv mod(g^k, P), \tag{2.5}$$

and

$$c_2 \equiv mod(mA^k, P), \tag{2.6}$$

where m is the plain data, and it is encrypted in the form of a pair (c_1, c_2). Further, the decryption of the pair (c_1, c_2) is done by using

$$x \equiv mod(c_1^a, P), \tag{2.7}$$

and computing m by multiplying x^{-1} with c_2

$$m \equiv mod(x^{-1}c_2, P). \tag{2.8}$$

3 Methodology

(i) Consider a colored image of size $(n-1) \times (n-1) \times 3$.

(ii) Segregate the image into three gray-scale images namely, red, green, and blue of sizes $(n-1) \times (n-1)$ each.

(iii) Choose the initial parameters for the Henon map and produce two sequences of random numbers x and y of size $1 \times (n-1)$ and $(n-1) \times 1$ respectively.

(iv) Multiple column vector y by row vector x to produce matrix key A_1 of size $(n-1)\times(n-1)$ and then multiply transpose of vector x with the transpose of vector y to generate matrix key A_2 of size $(n-1)\times(n-1)$.

(v) Take the red plane of the original image and XOR it with the matrix key A_1 to generate a partially encrypted red plane.

(vi) Then generate a hash of n-bits for this partially encrypted red plane by employing SHA256.

(vii) To generate a digital signature, encrypt the hash by using the ElGamal public cryptosystem.

(viii) Operate XOR between matrix key A_2 and the encrypted image in step 5.

(ix) Add digital signature to the coded image in step 8 as 256th row and to make the size $n\times n$ add a column vector of $(n-1)$ random numbers to this.

(x) To randomize the digital signature in the whole image, apply the Arnold cat map, and we get the final encrypted red plane.

(xi) Apply the whole mechanism on the green and blue plane and then concatenate the three planes to get the final encrypted colored image.

The encryption process is also represented in the form of a flowchart in Fig. 2. Also, the encrypted image after every stage can be seen in the Fig. 3. The decryption process of the image is exactly opposite to that of the encryption process.

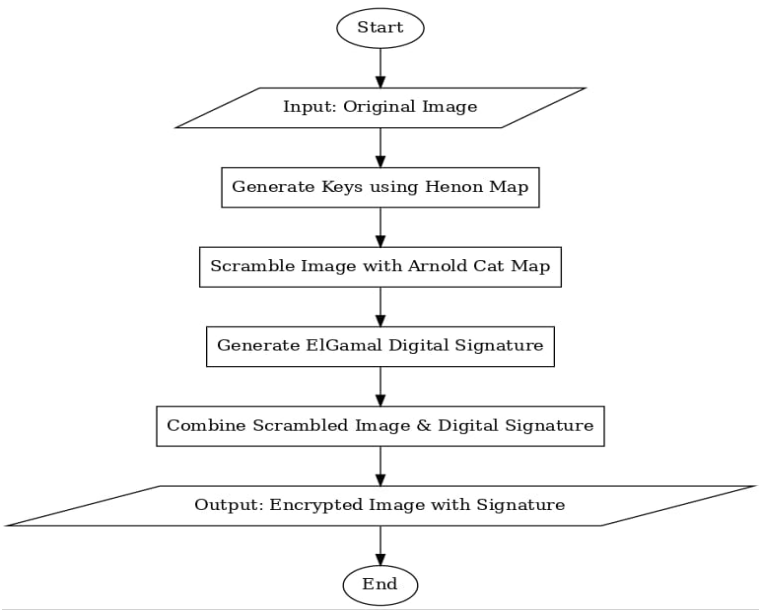


Figure 2: Encryption process in the form of flowchart.

4 Simulation Results

The section presents the simulation outputs used to evaluate the effectiveness and strength of the proposed encryption method. All the experimental results were obtained using MATLAB 2021b on a computing platform equipped with an 11th Gen Intel(R) Core(TM) i5-1135G7 processor operating at 2.40GHz (2.42 GHz) and 8GB of RAM. Five colored test images, Lena, Baboon, Home, Parrot, and Vegetables, taken from the USC-SIPI image database, were employed to validate the proposed method, each with dimension $255\times255\times3$. Fig. 4 illustrates the plain and encrypted images using the proposed scheme. It demonstrates that the proposed scheme can encrypt the colored image strongly and efficiently.

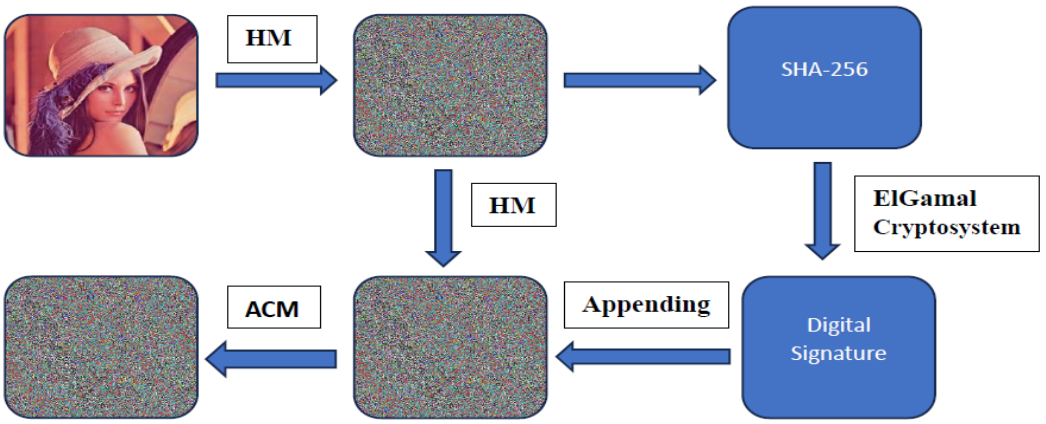


Figure 3: Encrypted image at every stage of encryption process.

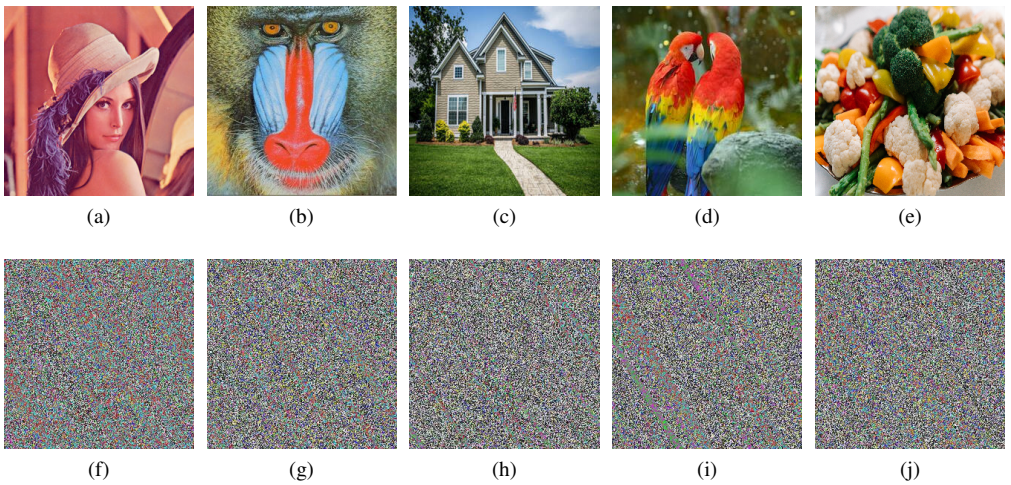


Figure 4: (a)-(e) represents the original images and (f)-(j) their corresponding encrypted images.

5 Security Analysis

This section provides a thorough analysis of this encryption technique. A robust encryption technique shields the image from all known attacks, including key search attacks, known plain text attacks, statistical attacks, and so on.

5.1 Key Space

The level of safety that the method of encryption provides for the image information is affected by key space [16, 29, 23]. The strength of the encryption method against key search attacks depends on the key space size. In this algorithm, two matrices of size $(n-1) \times (n-1)$ are used as a key, so the key set contains $2^{8*(2*(n-1)^2)}$ elements. Also, the number of iterations of the Arnold cat map acts as the key, and it contributes to the key space. So with this key space, the recommended encryption algorithm cannot be broken by a traditional key search attack.

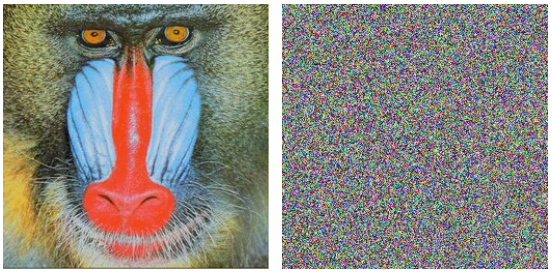
5.2 Key Sensitivity

A strong encryption technique depends on how sensitive the key is [16, 29, 23]. It ascertains that a minor alteration in the key will yield distinct outcomes. High key sensitivity is necessary for a robust encryption method so that even a small alteration in the key values results in an



(a) (b)

Figure 5: (a) Original Home image and (b) decrypted Home image with minor modification in Henon map parameters.



(a) (b)

Figure 6: (a) Original Baboon image and (b) decrypted Baboon image by changing the parameters of the Arnold cat map.

incorrectly decoded image. Fig. 5 and Fig. 6 show that when a minor adjustment is made in the key, the decrypted image differs greatly from the original. By making small changes in the initial parameters of the Henon map and Arnold cat map, the decrypted images are entirely different from the original images.

5.3 Entropy Analysis

Image entropy is a technique for determining how unpredictable the image data is. It is also referred to as Shannon entropy. An image with a uniform distribution pattern of pixel values has a higher degree of entropy. As a result, the entropy of the ciphered image must be mathematically greater than that of the plaintext image. An effective encryption method aims for the entropy value of the coded image to be near 8 [16, 29, 23]. This value is determined from the general expression of entropy given below

$$H = \sum_{j=0}^{2^n-1} p(m_i) log_2(\frac{1}{p(m_i)}), \tag{5.1}$$

where H is entropy, n denotes the number bits, and $p(m_i)$ denotes the probability of the pixel. Table 1 depicts the entropy of different encrypted images.

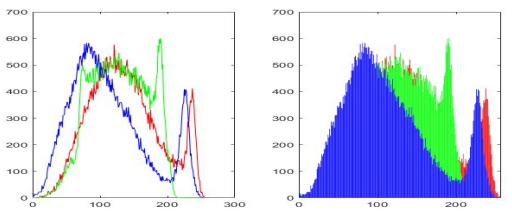
5.4 Histogram Analysis

A graph that shows the frequency distribution of a variable is called a histogram [16, 29, 23]. Histogram analysis offers a visual depiction of the distribution pattern of pixel intensity levels. It tallies the sum of image pixels at a particular intensity level. From this graph, one can know about the frequency of the image pixels at a specific intensity level. So, to protect the image from various statistical assaults, the encryption algorithm should be of the type that makes the histogram uniform. Fig. 8 depicts the the histogram of different plain-text images and histogram

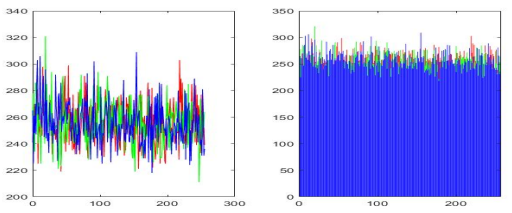
Table 1: Entropy of the encrypted images

Image	R	G	B	Average
Lena	7.9973	7.9973	7.9972	7.9973
Baboon	7.9974	7.9972	7.9972	7.9972
Home	7.9970	7.9975	7.9973	7.9972
Parrot	7.9972	7.9970	7.9973	7.9971
Vegetables	7.9973	7.9975	7.9971	7.9973
[16]	-	-	-	7.5884
[11]	7.9978	7.9979	7.9978	7.9978
[22]	7.9976	7.9976	7.9975	7.9975
[26]	7.9969	7.9973	7.9964	
[27]	-	-	-	7.9974
[5]	7.9935	7.9934	7.9936	7.9966
[12]	-	-	-	7.9972
[3]	-	-	-	7.9541

of the encoded images. It shows that histogram of the plain-text images are not uniform, but those of the ciphered images are uniform. From this observation, it is concluded that the presented encryption technique can withstand the different statistical attacks. Fig. 7 depicts the histogram of the initial colored image and the encoded colored image.



(a)



(b)

Figure 7: (a) Histogram of plain colored image and (b) encrypted colored image.

5.5 Chi-Square Test

The histogram of the image provides a graphical illustration of the patterns of pixel intensity distribution, and the chi-square test [16, 29, 23] uses numerical analysis to determine the uniformity of the secret image's pixel distribution. It is determined using the general formula shown below

$$\chi^2 = \sum_{i=0}^{2^n-1} \frac{(e0_i - e1_i)^2}{f1_i}, \tag{5.2}$$

where $e0_i$ and $e1_I$ denote the observed frequency and expected frequency, respectively. Given an image of dimension $M \times N$, then $e1_i = \frac{mn}{256}$. The critical chi-square values H_o with the degree of

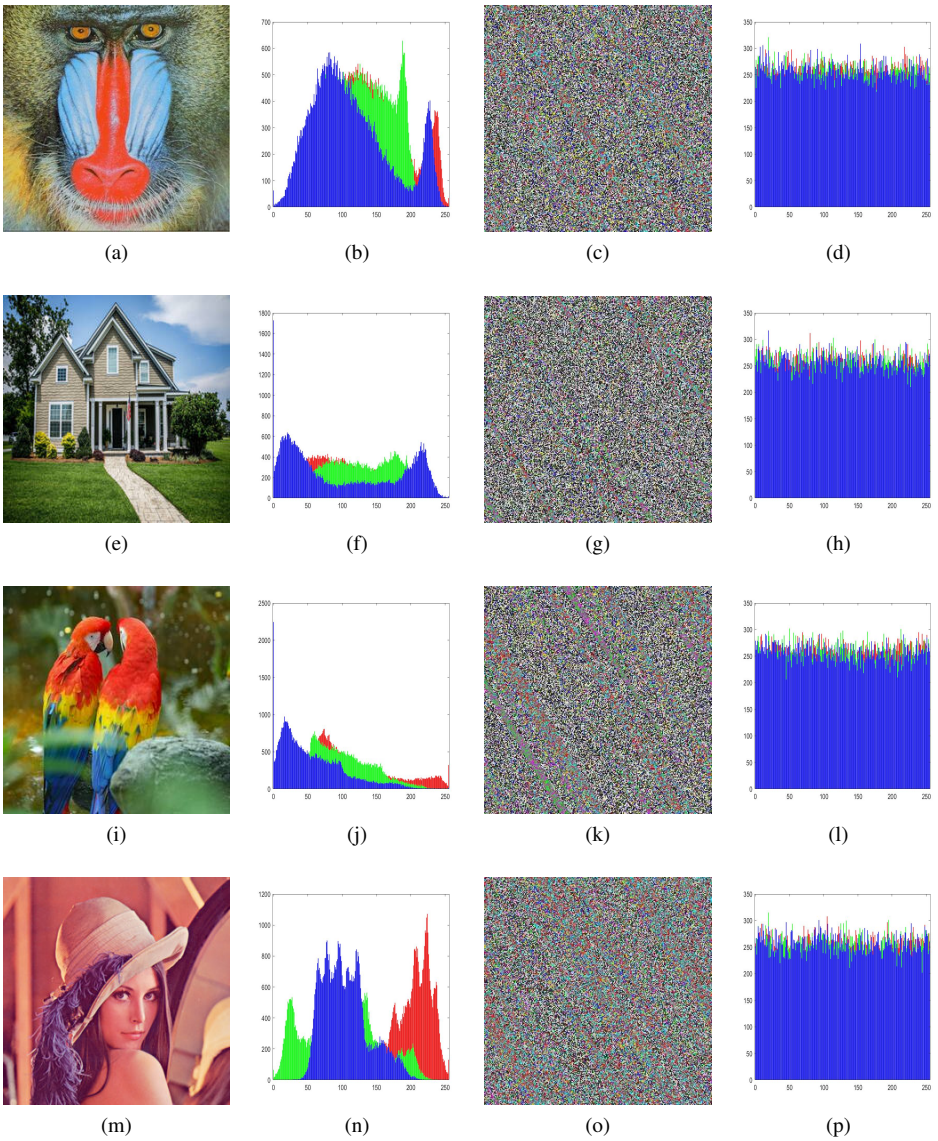


Figure 8: (a)-(d) show histograms of the original and encrypted Baboon image, (e)-(h) show histograms of the original and encrypted Home image, (i)-(l) show histograms of the original and encrypted Parrot image, and (m)-(p) show histograms of the original and encrypted Lena image.

freedom 255 and significance levels $s = 1\%$ and $s = 2\%$ are, respectively, $\chi^2_{(0.01,255)} = 310.4574$ and $\chi^2_{(0.05,255)} = 293.2478$. The chi-squared values shown in the Table 2 are below the critical values, so the hypothesis of uniform distribution of pixels is accepted, indicating the scheme's resilience to various statistical attacks.

5.6 Pixel Correlation Analysis

There exists a close relationship among each pixel within the image. In any image, there is enough correlation between every pair of adjacent pixels [16, 29, 23]. Due to this inter-pixel correlation, by knowing the intensity of one pixel, one can get the idea of their neighboring pixels. Consequently, a successful encryption method must minimize the correlation among neighboring pixels. For any given image, the relation in adjacent pixels can be chosen in three distinct orientations: diagonally (D), vertically (V), and horizontally (H). This relationship between the

Table 2: Chi-square values of the encrypted images

Image	R	G	B	Ave.	H_o
Lena	243.5703	245.2812	306.2812	265.0442	pass
Baboon	238.3290	252.7266	256.7969	249.2841	pass
Home	278.6641	285.7500	279.4609	281.2916	pass
Parrot	258.2500	304.1641	247.8281	270.0807	pass
Vegetables	266.4375	261.8516	268.1250	265.4713	pass
[11]	284.2314	218.6172	232.1876	245.0120	pass
[22]	221.1328	218.6172	226.3672	222.0390	pass

pixels in all these directions can be computed with the help of the correlation coefficient. The correlation coefficient is determined by applying the established formula given below

$$c_{xy} = \frac{CV(x,y)}{\sqrt{V(x) \times V(y)}}, \tag{5.3}$$

where

$$CV(x,y) = \frac{1}{M}\sum_{i=1}^{M}[x_i - E(x)][y_i - E(y)],$$

$$V(x) = \frac{1}{M}\sum_{i=1}^{M}[x_i - E(x)]^2,$$

and

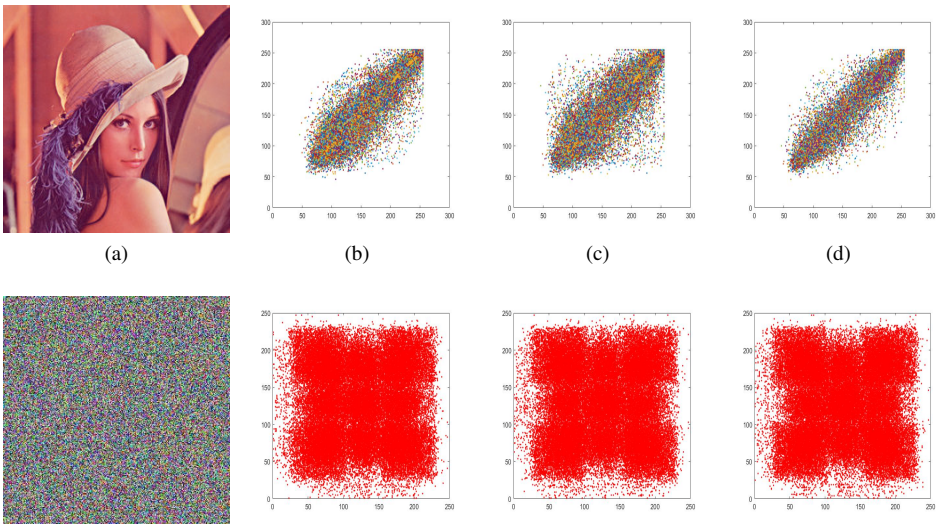
$$E(x) = \frac{1}{M}\sum_{i=0}^{M} x_i.$$

In the above expressions, M is the number of pixels, c_{xy} represents the correlation coefficient, $V(x)$ denotes the variance, $CV(x,y)$ is the covariance, and $E(x)$ denotes the expectation. Table 3 and Table 4 depict the pixel correlation among the pixels in the plain-text images and encoded images, respectively. It is evident that after encryption, the inter-pixel correlation reduced significantly.

Table 3: Correlation coefficient values of original images

Image	HC	VC	DC
Lena	0.9405	0.9694	0.9149
Baboon	0.8334	0.7934	0.7493
Home	0.8967	0.9163	0.9163
Parrot	0.9518	0.9697	0.9255
[11]	0.9231	0.9568	0.9425
[2]	0.9231	0.9568	0.9425
[22]	0.9458	0.9133	0.8673
[26]	0.9548	0.9816	0.9351
[27]	0.9356	0.9543	0.9060
[5]	0.9086	0.8313	0.9053
[12]	0.9520	0.9612	0.9643
[3]	0.9215	0.9145	-

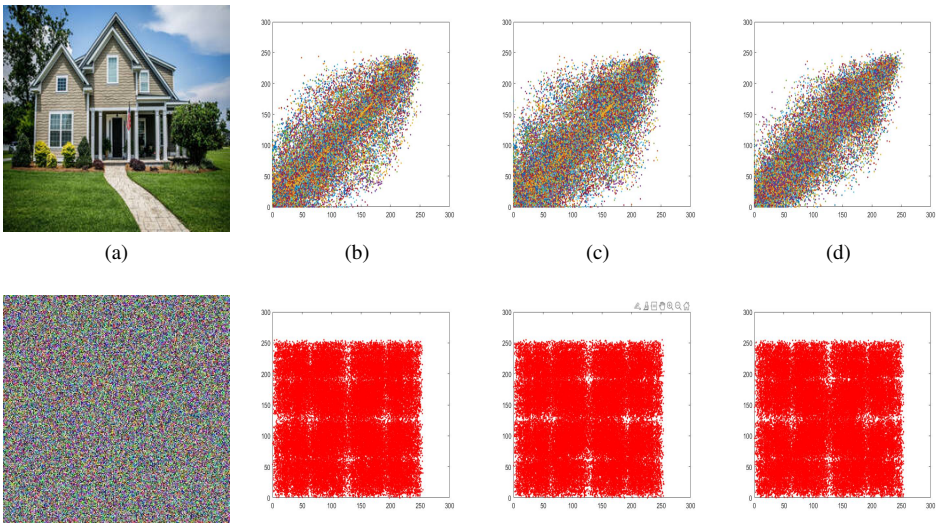
The intensity values of the image have been plotted in Figs. 9, 10, and 11. Since the intensity values of the original image are near one another. So, there is little space between the dots in the graph. On the other hand, the encrypted image's graph shows that the dots are scattered over a large region rather than concentrated.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 9: (a)-(d) show the inter-pixel correlation plot of the original Lena image in horizontal, diagonal, and vertical directions, and (e)-(h) show the inter-pixel correlation plot of the encrypted Lena image.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 10: (a)-(d) show the inter-pixel correlation plot of the original Home image in horizontal, diagonal, and vertical directions, and (e)-(h) show the inter-pixel correlation plot of the encrypted Home image.

Table 4: Correlation coefficient values of encrypted images

Image	HC	VC	DC
Lena	-0.0034	-0.0024	-0.0009
Baboon	-0.0026	-0.0011	0.0007
Home	0.0015	0.0011	-0.0016
Parrot	-0.0029	0.0015	-0.0013
Vegetables	-0.0019	-0.0001	-0.0004
[16]	-	-	Average 0.0037
[11]	0.0002	-0.0024	0.0026
[2]	0.0002	-0.0024	0.0026
[22]	0.0016	-0.0020	0.0047
[26]	0.0011	0.0010	-0.0008
[27]	0.00058	0.00061	-0.00030
[5]	0.0005	-0.0047	0.1313
[12]	0.0102	0.0180	0.0126
[3]	-0.0008	-0.0102	-

5.7 Differential Analysis

Attackers can decrypt an image through a process called differential cryptanalysis, which compares two ciphered images: first generated by directly encrypting a plain-text image, and the other one by encrypting the same image following minor modifications. With differential analysis, an attacker can find the connection between the initial and ciphered images. For the evaluation of the effectiveness of the suggested scheme against differential attacks, we employed differential analysis. Therefore, in order for a cryptographic technique to be considered safe against differential attacks, it must produce an encrypted image that is unrelated to the input image. To evaluate an encryption system's resistance to these kinds of attacks, the two parameters are employed: unified average changing intensity (UACI) and number of pixels change rate (NPCR) [32]. Assume that E_1 is the initial encoded image and that E_2 is the encoded image following a one pixel intensity adjustment. Next, the standard NPCR and UACI calculation formula is provided by

$$NPCR = \left(\frac{\sum_{x,y} F(x,y)}{mn} \right) \times 100\ \%, \tag{5.4}$$

if $I_1(x,y) = I_2(x,y)$, then F(x, y) = 0 , otherwise F(x, y) = 1, and

$$UACI = \frac{1}{mn} \sum_{x,y} \left(\frac{I_1(x,y) - I_2(x,y)}{255} \right) \times 100\ \%, \tag{5.5}$$

respectively. In the above expressions, (x,y) denotes the position of the image elements. If the UACI and NPCR values for gray-scale images are near 33.4635 and 99.6094, respectively, we can state that the encryption scheme is robust to withstand differential attacks. The computed NPCR and UACI values are displayed in the Table 5 and Table 6, respectively. This demonstrates that the encryption algorithm can nullify the effect of differential attacks.

5.8 Quality Measure

Standard similarity index (SSIM), mean square error (MSE), and peak signal to noise ratio (PSNR) are a few parameters employed to assess the quality of encryption.

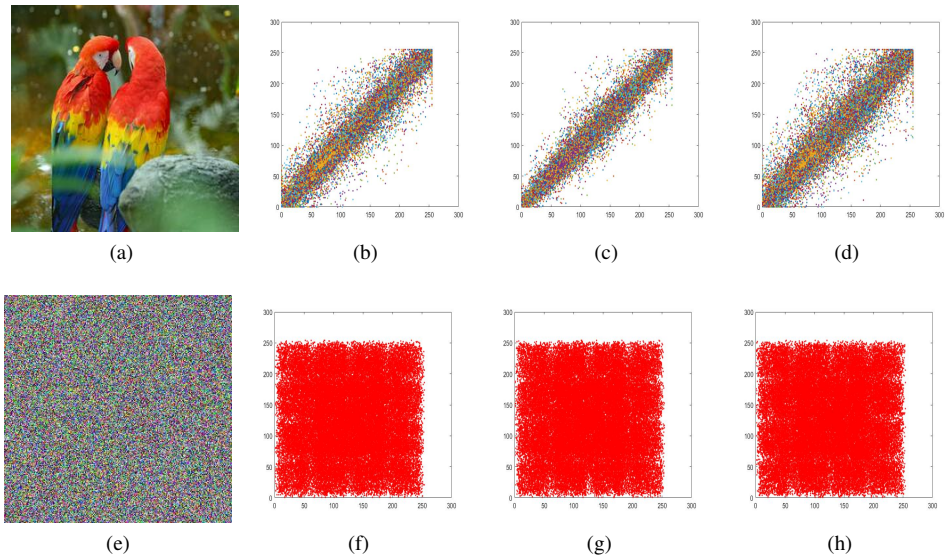


Figure 11: (a)-(d) show the inter-pixel correlation plot of the original Parrot image in horizontal, diagonal, and vertical direction, and (e)-(h) show the inter-pixel correlation plot of the encrypted Parrot image.

Table 5: NPCR values for encrypted images

Image	R	G	B	Average
Lena	99.8113	99.8220	99.8174	99.8169
Baboon	99.8144	99.8159	99.8190	99.8164
Home	99.8205	99.8205	99.8235	99.8205
Parrot	99.8174	99.8159	99.8220	99.8184
Vegetables	99.8190	99.8159	99.8174	99.8174
[16]	-	-	-	99.61
[11]	99.5832	99.6321	99.6154	99.6102
[22]	99.5941	99.6048	99.6002	99.5997
[26]	99.3057	99.2081	99.4568	99.3235
[27]	-	-	-	99.60
[5]	-	-	-	99.61
[3]	-	-	-	99.637

Mean Square Error (MSE)

To determine the total error between any two images, the MSE is used. Using the reference image as the starting image, we compute the mean square error (MSE) [16, 29, 23]. It must be greater among the plain image and the ciphered image, and roughly zero among the decrypted image and the initial image in order for an encryption scheme to be both secure and effective. For an image of dimension $M \times N$, MSE is determined with the help of the general formula stated below

$$MSE = \frac{1}{NM}\left(\sum_{i}^{N}\sum_{j}^{M}(f_i(i,j) - f_o(i,j))\right), \tag{5.6}$$

where f_i and f_o are the input and output images, respectively. Table 7 depicts the MSE values among the different initial and ciphered images, which demonstrate that the encryption algorithm is strong enough.

Table 6: UACI values for ciphered images

Image	R	G	B	Average
Lena	33.1858	33.2942	33.3643	33.2814
Baboon	33.3621	33.3264	33.3937	33.3607
Home	33.2153	33.2142	33.0640	33.1645
Parrot	33.3638	33.3010	33.2882	33.3176
Vegetables	33.3006	33.2876	33.2975	33.2952
[16]	-	-	-	32.41
[11]	33.4425	33.3901	33.3956	33.4049
[22]	33.2184	33.3138	33.3131	33.2817
[26]	33.2686	33.3941	33.4193	-
[27]	-	-	-	0.3349
[5]	-	-	-	33.12
[3]	-	-	-	34.359

Table 7: MSE values for different encrypted images

Image	R	G	B
Lena	3.4874×10^4	1.2552×10^4	1.2234×10^4
Baboon	2.1459×10^4	1.8297×10^4	1.5901×10^4
Home	1.4442×10^4	1.7620×10^4	1.6971×10^4
Parrot	1.5220×10^4	1.1542×10^4	5.4149×10^3
Vegetables	1.8983×10^4	7.0307×10^3	5.0879×10^3
[11]	4.192×10^4	7.180×10^4	8.993×10^3
[22]	-	-	Ave. 8.9804×10^3
[26]	8.1869×10^3	7.2852×10^3	9.0537×10^3

Peak Signal to Noise Ratio (PSNR)

This allows you to assess how much the signal from two images dominates the noise. This variable is also used to monitor the encryption reliability. It is computed using the original image as a guide for the encoded image. The level of encryption quality will be higher with the lower PSNR value [16, 29, 23], which should not exceed 10 dB. The lower PSNR for an encoded image and the increased value observed in the decrypted image are solid proof of an efficient encryption technique. The PSNR values are computed using the general formula described below

$$PSNR = 10\left(log_{10}(\frac{2^n - 1}{MSE})\right), \tag{5.7}$$

where n is the total number of bits used in the pixel value. PSNR values for various encrypted images are provided in the Table 8, which are all lower than 10db.

Structural Similarity Index (SSIM)

The SSIM approach works because there are structural similarities between two identical images rather than two different images [16, 29, 23]. It gauges how well the encryption procedure is working. It is employed to estimate the observed difference between two similar images, and its value falls within the range [-1,1]. An SSIM value near zero signifies high-quality encryption.

Table 8: PSNR values for various encrypted images

Image	R	G	B
Lena	7.8479	8.5867	9.6321
Baboon	8.9531	9.5080	8.6336
Home	8.2015	8.5235	7.2165
Parrot	8.3555	8.9523	7.1653
Vegetables	7.2440	8.0189	6.9206
[11]	8.145	7.547	7.128
[22]	-	-	Ave. 8.5979
[26]	8.9996	9.5064	8.5625

The standard formula utilized to obtain the SSIM values is as follows

$$SSIM = \frac{(2a_o a_c + c_1)(2b_{oc} + c_2)}{(a_o^2 + a_c^2 + c_1)(b_o^2 + b_c^2 + c_2)}, \tag{5.8}$$

where c_1 and c_2 are variables, a_o, b_o, b_c, and a_c represent mean, covariance, variance, and standard deviation, respectively. Table 9 gives the SSIM values among different plain and ciphered images, which are all in the ideal range.

Table 9: SSIM values for encrypted images

Image	R	G	B
Lena	0.0114	0.0085	0.0083
Baboon	0.0090	0.0114	0.0107
Home	0.0119	0.0129	0.0095
Parrot	0.0098	0.0091	0.0076
Vegetables	0.0075	0.0082	0.0089
[11]	0.002	0.005	0.005
[22]	-	-	Ave. 0.0020
[26]	0.0010	0.0010	0.0068

5.9 Randomness Test

Randomness is essential for the image's encrypted data set. To protect the image data from hackers, the encrypted image's data must be sufficiently random [16, 29, 23]. A run test is used to confirm whether the encrypted image data is unpredictable or not. The data is random if this test returns a result of 0; if it returns a result of 1, the data is not random. This method has been used to apply this test to the encrypted image. As the Table 10 illustrates, the result of this is zero for each colour component. Thus, there is enough unpredictability in the suggested encryption algorithm to fend off image attacks.

5.10 Noise Attack Analysis

Noise can obstruct the cipher images during transmission, either accidentally or on purpose from an unauthorized party [16, 29, 23]. When an image is sent over an open internet channel, its surroundings can affect the information it contains. An effective method can be resilient against noise attacks if, after decryption, it can show the important visual representation of image data affected by the noise. In this analysis, we apply noise to the image and examine the result to see how well the encryption method performs with noise. There are a few methods, such as

Table 10: Results for randomness test for encrypted images

Image	Plane	Result	Randomness
Lena	R	0	pass
	G	0	pass
	B	0	pass
Baboon	R	0	pass
	G	0	pass
	B	0	pass
Home	R	0	pass
	G	0	pass
	B	0	pass
Parrot	R	0	pass
	G	0	pass
	B	0	pass
Vegetables	R	0	pass
	G	0	pass
	B	0	pass

Gaussian noise and the Salt and Pepper approach, to confirm the noise influence on the image data. As seen in Fig. 12 and Fig. 13, the decrypted images, which were obtained after inserting noise into the encrypted images, are visible.



(a) (b) (c) (d)

Figure 12: (a)-(d) show images decrypted by applying Gaussian noise.



(a) (b) (c) (d)

Figure 13: (a)-(d) show images decrypted by applying salt and pepper noise.

The Table 11 displays the PSNR and SSIM values for noise-attacked decrypted images. PSNR values are greater than 10db and SSIM values are approaching 0, which states that the recommended encryption strategy works well against a range of noise attacks.

Table 11: Standard calculation of decrypted image attacked with noise

Attacks	Image	PSNR	SSIM
Salt and pepper			
I=0.01	Lena	28.5457	0.9680
I=0.02	-	25.5004	0.9391
I=0.10	-	18.6450	0.7585
I=0.20	-	15.5757	0.5939
Gaussian			
V=0.01	Lena	14.6407	0.5418
V=0.02	-	14.5325	0.5341
V=0.10	-	13.2776	0.4442
V=0.20	-	11.3180	0.2887
Occlusion			
cropping 10%	Baboon	16.3276	0.7392
cropping 20%	-	13.4974	0.5774
cropping 25%	-	10.9627	0.3165
cropping 50%	-	10.2306	0.2161

5.11 Occlusion Attack Analysis

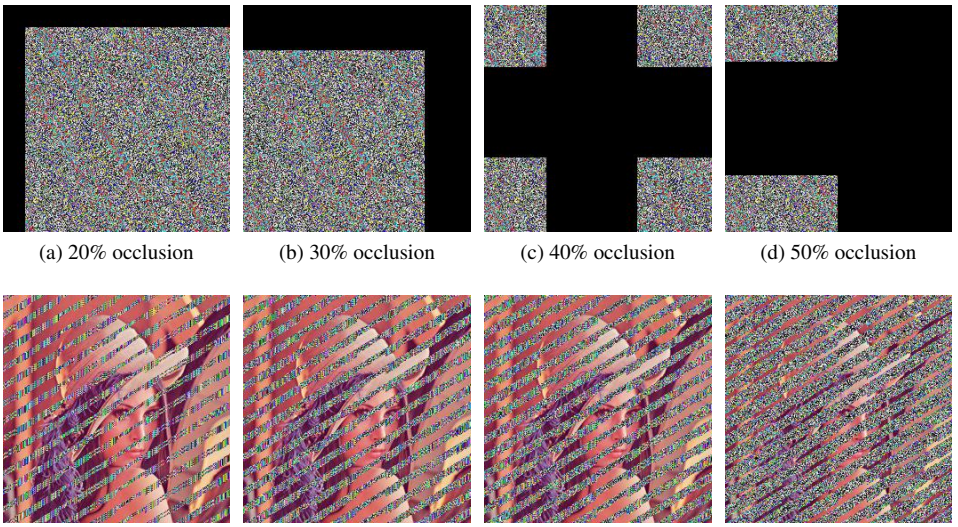
Due to malicious effects, information loss may happen as the image is being communicated on the open channel [16, 29, 23]. To test the strength of the image encryption scheme in response to this type of attack, we do an occlusion attack analysis. It is a technique to evaluate the degree to which encrypted images can be restored to their original form if some data is lost during transfer. The occlusion attack analysis for the Lena image is displayed in Fig. 14. The illustration amply illustrates how the recommended algorithm is both effective and resilient against the cropping attack, even in the event that the encrypted image is cropped or information is lost during the exchange, yielding a suitable and understandable image. Table 11 shows the SSIM and PSNR values for the decrypted images with acknowledgment to the plain images, which are in accordance with their ideal values. This concludes that the presented encryption scheme is strong against the information occlusion attacks.

5.12 Complexity Analysis

A cryptosystem that works well should have a low computational overhead and be able to process data quickly enough [16, 29, 23]. The speed at which data is encoded and decoded depends on the degree of complexity of the encryption method, the hardware, and the software being used. The suggested algorithm's encryption times for a range of images are compared to those of some other algorithms in the Table 12, which demonstrates that the presented encryption method is fast in contrast to some other established methods.

6 Conclusion

This article presents a new and effective method for the encryption of color images utilizing chaotic maps with a digital signature. The two arrays generated by the chaotic map as keys greatly enhance the safety of the image data. The randomness introduced by the Arnold cat map amplifies the protection of the image information by scattering the digital signature in the complete image. The salient feature of this encryption scheme is that the digital signature is assessed using the partially encrypted image rather than being extracted straight from the original image. The algorithm known as the ElGamal public cryptosystem is employed in the assessment of digital signatures and guarantees key security for both the sender and the recipient. Moreover,



(a) 20% occlusion (b) 30% occlusion (c) 40% occlusion (d) 50% occlusion

(e) Decryption of 20% occluded image (f) Decryption of 30% occluded image (g) Decryption of 40% occluded image (h) Decryption of 50% occluded image

Figure 14: Images (a)-(d) show occluded images with varying percentage and (e)-(f) show their corresponding decrypted images.

Table 12: Complexity analysis of the proposed algorithm

Encryption algorithm	Image	size	speed(s)
Proposed	Lena	255×255	0.3799
	Home	255×255	0.4201
	Vegetables	255×255	0.4196
	Parrot	255×255	0.4359
[13]		256×256	8.2
[28]		256×256	36.0663
[9]		256×256	5.7754

the signature is not hidden in the image from which it is generated; instead, the array key is employed to further encrypt the image, and then the signature is added to this image. The Arnold cat map encrypts the image containing the digital signature. By doing this, the digital signature throughout the image will be randomly scattered, improving the level of protection of the image information. The statistical analysis of the system is supplied in the form of key set analysis, sensitivity of key, MSE, PSNR, SSIM, differential analysis, entropy analysis, etc., to meet the needs required to guarantee the system's dependability. Furthermore, the safety of the system and reliability against known common attacks are reinforced by the efficacy analysis of the proposed technique. In the future, the suggested scheme can be extended to handle real-time and large-scale multimedia data like video. Further, optimization can be explored to improve computational efficiency for practical applications. Moreover, employing adaptive key management algorithms may enhance the strength of the scheme against advanced attacks.

References

[1] S. Alam, A. Jamil, A. Saldhi, and M. Ahmad, *Digital image authentication and encryption using digital signature*, International Conference on Advances in Computer Engineering and Applications (ICACEA), IEEE, 332–336, (2015).

[2] J. Chen, Y. Wu, Y. Sun, and C. Yang, *Image encryption algorithm using 2-order bit compass coding and chotic mapping*, Symmetry, **14**, 1–21, (2022).

[3] Fu. Chong, W. Li, Z. Meng, T. Wang, and X. Li, *A symmetric image encryption using chaotic Baker map and Lorenz system*, Ninth International Conference on Computational Intelligence and Security, IEEE, 724–728, (2013).

[4] A. Gaffar, A.B. Joshi and D. Kumar, *Image encryption using Gingerbreadman map and RC4A stream cipher*, Application and Applied Mathematics, **15**, 1230–1244, (2020).

[5] V. Guleria, and D.C. Mishra, *A new multi-layer RGB image encryption algorithm based on Diffie-Hellman cryptography associated with FrDCT and Arnold transform*, Multimedia Tools and Applications, **79**, 33119–33160, (2020).

[6] J. Hoffestein, J. Pipher, and J.H. Silverman, *An Introduction to Mathemaical Cryptography* (1st ed.), Springer, (2008).

[7] H.I. Hussein and R.J. Mstafa, *An enhanced ElGamal cryptosystem for image encryption and decryption*, International Conference on Computer Science and Software Engineering (CSASE), IEEE, 337–342, (2022).

[8] B.Y. Irani, P. Ayubi, F.A. Jabalkandi, M.Y. Valandar, and M.J. Barani, *Digital image scrambling based on a new one dimensional coupled sine map*, Nonlinear Dynamics, **97**, 2693–2721, (2019).

[9] N. Iqbal, M. Hanif, and S. Abbas, *DNA strands level scrambling based color image encryption scheme*, IEEE Access, **8**, 178167–178182, (2020).

[10] A.B. Joshi, D. Kumar, D.C. Mishra, *Security of digital images based on 3D Arnold cat map and elliptic curve*, In. J. of Image and Graphics, **21**, 10–28, (2020).

[11] A.B. Joshi, D. Kumar, D.C. Mishra, and V. Guleria, *Colour image encryption based on 2D discrete wavelet transform and 3D logistic chaotic map*, J. of Modern optics, **67**, 933–949, (2020).

[12] A.B. Joshi, and A. Gaffar, *Digital image encryption based on chaotic map and circulant matrix*, GANITA, **68**, 103–115, (2018).

[13] K.C. Jithin, and S. Sankar, *Colour image encryption algorithm combining, Arnold map*, DNA sequence operation, and amandelbrot set, J. of Information Security and Application, **50**, (2020).

[14] N. Koul,, N. Kumar, A. Sayeed,, C. Verma, and M.S. Raboaca, *Data exchange techniques for Internet of Robotic Things: Recent Developments*, IEEE Access, **10**, 102087–102106, (2022).

[15] D. Kumar, A.B. Joshi, and A. Gaffar, *Double color-image encryption based on 2D fractional discrete Fourier transform and Arnold cat map*, GANITA, **70**, 129–141, (2020).

[16] P.N. Lone, D. Singh and U.H. Mir, *A novel color image encryption using random matrix affine cipher and the chaotic maps*, Journal of Modern Optics, **68**, 510–522, (2021).

[17] P.N. Lone, D. Singh, and U.H. Mir, *Image encryption using DNA coding and three-dimensional chaotic system*, Multimedia Tools and Applications (Springer), **81**,5669–5693, (2021).

[18] Y. Luo, J. Yu, and L. Lai, *A novel chaotic image encryption algorithm based on improved baker map and logistic map*, Multimedia Tools and Applications, **78**, 22023–22043, (2019).

[19] F. Masood,, W. Boulila,, A. Alsaeedi,, J.S. Khan,, J. Ahmad,, M.A. Khan, and S.U. Rehman, *A novel image encryption scheme based on Arnold cat map*, Newton-Leipnik system and Logistic Gaussian map, Multimedia Tools and Applications, **81**, 30931–0959, (2022).

[20] A.E. Naeem,, A.B. Joshi,, D. Kumar, and F.E. Abd El-Samie, *Few-detail encryption algorithm based on diffusion and confusion using Henon and Baker chaotic map*, **28**, 2851–2861, (2022).

[21] R.R. Parida, B.K. Singh, and C. Pradhan, *A novel approach for image encryption using Zaslavskii map and Arnold's cat map*, Advances in Intelligent System and Computing, **1**, 269–282, (2021).

[22] S. Patel, and A. Vaish, (2021) *A novel image coding through the chaos theory and compressed sensing*, Proceedings of International Conference on Data Science and Applications, **2**, 615–623.

[23] A. Paul, R. Singh, and D. Singh, *A multilayer encryption scheme using Vigenère cipher and chaotic maps*, SN Computer Science, **6**, 1-17, (2025).

[24] R. Rezaei, H. Ghanbari, and R. Enayatifar, *An image encryption approach using tuned Henon chaotic map and evolutionary algorithm*, Nonlinear Dynamics, **111**, 9629–9647, (2023).

[25] A. Sinha, and K. Singh, *A technique for image encryption using digital signature*, Optics Communications, **218**, 229–234, (2003).

[26] P. Sankhe, S. Pimple, S. Singh, and A. Lahane, *An image cryptography using Henon map and Arnold map, In. Research J. of Engineering and Technology*, **5**, 2395–0072, (2018).

[27] R.D. Syah, and R.J. Shatril, *Digital image cryptography using combination of Arnold cat map and Bernoulli map based on chaos theory*, Int. Research J. of Advanced Engineering and Science, **4**, 258–262, (2019).

[28] S.J. Sheela, S.V. Suresh, and D. Tandur, *Image encryption based on modified Henon map using hybrid chaotic shift transform*, Multimedia Tools and Application, **77**, 25223–25251, (2018).

[29] D. Singh, and S. Singh, *A multiphase encryption scheme using RSA, modified RMAC and Chen's hyperchaotic map*, Multimedia Tools and Application, **83**, 57059–57088, (2023).

[30] A. Vaish, and S. Patel, *Securing color images using DNA coding and cosine stockwell transformation in wavelet domain*, Optik, **266**, (2022).

[31] X.Y. Wang, P. Li, Y.Q. Zhang, L.Y. Liu, H. Zhang, and X. Wang, *A novel color image encryption scheme using DNA permutation based on the Lorenz system*, Multimedia Tools and Application, **77**, 6243–6265, (2018).

[32] Y. Wu, J.P. Noonan, and S. Agaian, *NPCR and UACI randomness tests for image encryption*, Cyber Journals: Multidisciplinary Journals in Science and Technology, J. of Selected Areas in Telecom (JSAT), **1**, 31–38, (2011).

[33] A.E. Naeem, A.B. Joshi, D. Kumar, F.E. Abd El-Samie, *Low detail encryption algorithm based on diffusion and confusion using Henon and Baker chaotic map*, Research square, (2022).

[34] D. Singh, M.K. Patel, Mukesh, *Color image encryption scheme based on RC4A and chaotic maps*, Palestine Journal of Mathematics, **14**, 530–553, (2025).

[35] S.E. Kaddouhi, Y. Qobbi, A. Abid, M. Jarjar, H. Zaaraoui, A. Jarjar, *A new image encryption approach that uses an improved Hill-Vigenère method and chaotic maps*, Multimedia Tools and Applications, **84**, 17279–17307, (2025).

[36] D. singh, S. Kumar, *Amultiphase encryption scheme using RSA, modified RMAC and Chen's hyperchaotic map*, Multimedia Tools and Applications, **83**, 57059–57088, (2024).

[37] W. Alexan, M. Elkandoz, M. Mashaly, E. Azab, A. Aboshousha, *Color image encryption through chaos and KAA map*, IEEE Access, **11**, 11541–11554, (2023).

Author information

A. Paul, R. Singh, Department of Mathematics, Guru Nanak Dev University, Amritsar, Punjab, India.
E-mail: amitpaulcuj@gmail.com; rajwindermaths221@gmail.com

M. Rao, Department of Mathematics and Statistics, Central Univeristy of Punjab, Bathinda, Punjab, India.
E-mail: mukesh01259@gmail.com

M. K. Patel, Department of Mathematics, National Institute of Technology, Nagaland, India.
E-mail: mkpitb@gmail.com

D. Singh, School of Undergraduate Studies, Dr. B. R. Ambedkar University Delhi, Delhi-6
Department of Mathematics and Statistics, Central Univeristy of Punjab, Bathinda, Punjab, India.
E-mail: deepsinghspn@gmail.com

Received: 2025-09-02
Accepted: 2026-01-08