iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** IV **Month of publication:** April 2025

DOI: https://doi.org/10.22214/ijraset.2025.69071

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

Enhanced Image Encryption using Chaotic Substitution, DNA Encoding, and Random Permutation

T Pavan Kumar[1], Odugu Srinivasa Rao[2], K Saraswathi[3]
[1]*M.Tech,* [2]*Professor,* [3]*Assistant Professor, CSE department, UCEK, JNTUKakinada, Andhra Pradesh, India*

Abstract: ***With the increasing challenges of securing multimedia transmission over the internet, image encryption has emerged as a critical research area. This paper presents a novel image encryption scheme that integrates chaotic substitution, DNA encoding, and random permutation techniques to enhance security. The proposed method exploits the unpredictability of chaotic sequences, the computational benefits of DNA encoding, and the randomness of permutation to strengthen encryption.In this approach, pixel values are initially converted into DNA sequences using a predefined encoding scheme. A high-dimensional Lorenz chaotic map generates chaotic sequences, which are then employed for pixel substitution, effectively scrambling the DNA-encoded pixel values. Furthermore, a random permutation technique known as the Fisher-Yates shuffle is applied to further disrupt the substituted pixel values, increasing the overall complexity of the encryption process.By integrating these techniques, the proposed scheme achieves robust security, making it resistant to various cryptographic attacks. Experimental results validate the effectiveness of this approach, demonstrating high encryption strength while maintaining computational efficiency.***

Keywords: ***Chaotic Substitution, DNA Encoding, Random Permutation, Lorenz Chaotic Map, Fisher-Yates Shuffle, Cryptographic Attacks, Computational Efficiency***

I. INTRODUCTION

Image encryption plays a crucial role in information security, aiming to protect the confidentiality and integrity of digital images. As digital imagery is widely used across various domains, including photography, multimedia, medical imaging, and satellite imaging, ensuring the secure transmission, storage, and sharing of images has become increasingly important [1],[2].The fundamental objective of image encryption is to convert visual data into an unreadable format, preventing unauthorized access. By securing images through encryption, sensitive information is safeguarded against unauthorized viewing, modification, or interception [3]. Encryption methods employ mathematical algorithms and cryptographic techniques to alter pixel values or distort an image's visual structure. These algorithms transform the original image into an unintelligible encrypted version, which can only be decrypted using the appropriate key [4],[5].

Even if an encrypted image is intercepted, an attacker cannot retrieve the original content without the decryption key. Image encryption techniques generally fall into two main categories: symmetric key encryption and public key encryption.

1) Symmetric Key Encryption: This method relies on a single secret key for both encryption and decryption. The key must be securely shared between the sender and receiver to maintain confidentiality.
2) Public Key Encryption: This technique utilizes a pair of keys—a public key for encryption and a private key for decryption. Since the private key remains confidential, this approach enhances security by eliminating the need for key sharing [9],[10].

By implementing robust encryption mechanisms, digital images can be securely protected, ensuring confidentiality and preventing unauthorized access in various applications.

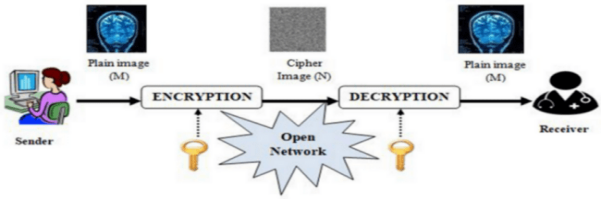


FIG 1: IMAGE ENCRYPTION

A. Chaotic Maps:

Chaotic maps are mathematical functions that generate a very complex and unpredictable behaviour, making them useful in applications such as encryption and secure communications. This unpredictability poses certain problems, particularly in cryptographic applications where generating sequences that are difficult to reverse-engineer is essential for ensuring data security. Additionally, the behaviour of chaotic maps can change drastically with slight variations in parameters, complicating their analysis and application.

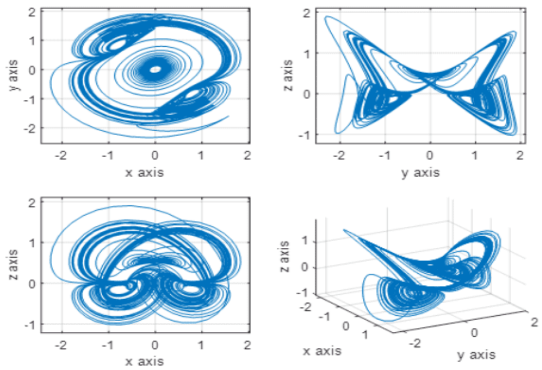


FIG 2: CHAOTIC MAPS

The applications of chaotic maps are diverse and significant. In the realm of cryptography, they are utilized to create encryption algorithms due to their inherent unpredictability and extreme sensitivity, which help in generating secure keys and encoding data. In signal processing, chaotic maps find use in applications such as image scrambling and watermarking, where chaotic behaviour assists in obscuring original data. Moreover, they are employed in physics and biology to model complex systems in nature, such as population dynamics and fluid turbulence.

B. DNA encoding:

DNA encoding refers to the process of representing data using the sequences of nucleotides found in deoxyribonucleic acid (DNA). This approach takes advantage of DNA's natural storage ability of vast amounts of information in a highly compact form. It consists of four nucleotides: adenine (A), cytosine (C), guanine (G), and thymine (T) can be used to encode binary data by mapping binary digits (0s and 1s) to nucleotide sequences. For instance, one could represent 00 as A, 01 as C, 10 as G, and 11 as T. This encoding method allows for the storage of digital information in a biochemical medium, leveraging DNA's high density and stability as a storage format.

A major advantage of DNA-encoding is its exceptional ability for high-density information storage. Theoretically, a single gram of DNA can store approximately 215 petabytes (215 million gigabytes) of data, far surpassing traditional storage mediums like hard drives and SSDs. Moreover, DNA is remarkably durable; under optimal conditions, it can last for thousands of years without degradation, making it an attractive option for archiving critical data. The process of DNA synthesis, which involves creating custom DNA sequences, has become increasingly accessible, allowing researchers and companies to encode digital data into synthetic DNA.

In addition to its storage capabilities, DNA encoding has applications in various fields, including computer science, biology, and medicine. In computer science, researchers are exploring DNA computing, where computations are performed using biochemical reactions instead of traditional silicon-based methods. This has the potential to revolutionize how complex problems are solved, especially in parallel processing. Additionally, DNA data storage can be utilized for long-term archiving of critical information, such as historical records or scientific data.

Despite its promising advantages, DNA encoding also faces several challenges. The process of synthesizing and reading DNA sequences can be costly and time-consuming, limiting its current practical applications. Additionally, error rates in DNA synthesis and sequencing can lead to data corruption, necessitating the development of robust error-correction methods to ensure data integrity. Furthermore, the environmental impact of large-scale DNA synthesis processes is an area of concern, prompting research into more sustainable practices.

C. Random permutation:

Random permutations refer to the rearrangement of elements in a set in a random order. A permutation of a set of n distinct objects refers to any possible arrangement of those objects. The total number of permutations of a set containing n elements is given by n! (n factorial), which is the product of all positive integers from 1 to n. Random permutations are crucial in various fields, particularly in mathematics, computer science, and statistics, where they are used to ensure unbiased sampling and to study properties of algorithms.

One of the essential properties of random permutations is their uniform distribution. When selecting a random permutation from a set of n elements, each permutation has an equal probability of being chosen, which is 1/n!. This uniformity is vital in applications such as Monte Carlo methods, where random permutations can help estimate the properties of complex systems by sampling. Additionally, random permutations are closely related to combinatorial structures and play a significant role in probability theory and statistical analysis, particularly in problems involving order statistics and random sampling.

Random permutations have several practical applications across various domains. In computer science, they are employed in algorithms for data shuffling, sorting, and cryptographic schemes, where randomization helps ensure security and efficiency. For example, random permutations can be used in randomized algorithms like QuickSort, where the pivot is chosen randomly to improve performance on average. In the field of cryptography, random permutations are critical in designing secure encryption algorithms, ensuring that data is transformed in an unpredictable manner. Moreover, random permutations are utilized in statistical sampling techniques, such as randomization in clinical trials, where the order of treatment assignments is randomized to eliminate bias.

Despite their usefulness, generating random permutations efficiently can be challenging. The naive approach of generating all permutations and then selecting one at random is computationally expensive for large sets due to the factorial growth of permutations. Instead, more efficient algorithms, such as the Fisher-Yates shuffle, allow for the generation of a random permutation in linear time O(n). This algorithm iterates through the array, swapping each element with a randomly selected element that follows it, ensuring that all permutations are equally probable.

II. RELATED WORKS

In modern cryptographic research, chaotic maps and DNA encoding have emerged as powerful tools for secure image encryption [1][2]. Chaotic systems exhibit sensitivity to initial conditions and generate pseudo-random sequences, making them well-suited for encryption applications. DNA encoding, inspired by biological DNA sequences, provides an additional layer of complexity and non-linearity. The integration of chaotic maps with DNA encoding enhances security by increasing key space, improving diffusion, and strengthening resistance against statistical and differential attacks.

Various chaotic maps have been used in image encryption to achieve efficient pixel permutation and diffusion. Low-dimensional maps such as the Logistic Map and Sine Map are commonly used due to their simplicity and ease of implementation [1][2]. However, their relatively low complexity can make them easily compromised to certain cryptographic attacks. To overcome these limitations, researchers have explored high-dimensional chaotic systems, such as the Rössler system [3], Chen's chaotic system [4], and Lorenz system [5]. These higher-dimensional maps exhibit more complex chaotic behaviour, improving encryption strength and key sensitivity.

Additionally, hyperchaotic systems that extend traditional chaotic maps by introducing more control parameters have been widely adopted. The Four-Dimensional Hyperchaotic System [6] and Seven-Dimensional Hyperchaotic System [7] significantly increase randomness, making them highly effective in preventing brute-force attacks. The Baker Map [1] [8] is commonly used for permutation due to its ability to shuffle image pixels in a highly unpredictable manner, further enhancing security.

DNA computing has been integrated into encryption schemes to increase complexity. DNA encoding represents pixel values as nucleotide sequences (A, T, C, G) and applies DNA rules such as addition, subtraction, and XOR operations to enhance diffusion [1][2][8][7]. Various encoding rules, such as complementary pairings and logical operations based on Watson-Crick complementarity, have been explored to introduce non-linearity into encryption processes [9].

DNA encoding is often combined with chaotic sequences to create a dynamic encoding process, where the encoding rule is selected based on a chaotic sequence, ensuring unpredictable transformations [10]. This approach significantly improves security by preventing attackers from exploiting fixed encoding patterns. Additionally, DNA sequence-based key generation using chaotic maps ensures that the decryption process is highly sensitive to initial conditions, making brute-force attacks infeasible [11].

Recent studies have combined chaotic maps and DNA encoding to develop hybrid encryption methods. A common approach involves using chaotic maps for pixel permutation followed by DNA encoding for diffusion [12].

For example, researchers have proposed using the Baker Map for pixel scrambling, followed by DNA encoding operations based on chaotic key sequences to apply non-linear transformations [13].

Other hybrid techniques utilize multiple chaotic maps in combination with DNA encoding. In one study, a two-dimensional Sine-Logistic chaotic system was combined with DNA sequence operations to improve security and resistance against statistical attacks [14]. Another approach employed the hyperchaotic Lorenz system with DNA XOR encoding to enhance the encryption's diffusion properties, ensuring high resistance against differential attacks [15].

III. METHODS AND METHODOLOGY

A. *Chaotic Key Generation Using Lorenz System:*

A high-dimensional chaotic map, the Lorenz system is used to generate two chaotic sequences. The Lorenz system is generated by the following differential equations:

$$\frac{dx}{dt} = \sigma(y - x)$$
$$\frac{dy}{dt} = x(\rho - z) - y$$
$$\frac{dz}{dt} = xy - \beta z$$

Where σ, ρ, and β are system parameters that dictate chaotic behaviour. The initial values (x_0,y_0,z_0) are perturbed using a key that is derived from the SHA-256 hash of the user-provided password, ensuring unique encryption keys per user. A discrete-time iterative approximation with a step size dt =0.01 generates the chaotic sequence. The normalized sequence (modulo 1 operation) is then used for pixel shuffling and DNA encoding modifications.

The Lorenz system is notable for its chaotic solutions and sensitivity to initial conditions, famously illustrated by the "butterfly effect." The trajectories of the Lorenz system in phase space exhibit a strange attractor, characterized by its distinctive shape resembling a butterfly or figure-eight. Check FIG 2.

B. *Pixel Scrambling Using Fisher-Yates Shuffle:*

Pixel scrambling is performed using the Fisher-Yates shuffle, a well-known algorithm for generating a random permutation. The chaotic sequence generated from the Lorenz system determines the permutation indices. Given a flattened image array III, the shuffle index array SSS is computed as:

$$\mathbf{S} = \text{argsort (chaotic sequence)}$$

This ensures the image pixels are permuted according to the unpredictable nature of the Lorenz sequence. The inverse shuffle is used during decryption to restore the original pixel order.

C. *DNA Encoding and Modification:*

To increase diffusion and introduce non-linearity, each pixel is transformed into a DNA sequence. Each 8-bit pixel value is mapped to a 4-base DNA sequence based on the following encoding scheme:

Binary	DNA
00	A
01	C
10	G
11	T

For each pixel, the chaotic sequence dictates a DNA modification operation. The modification shifts the DNA base cyclically using:

$$\text{modified\_DNA} = \text{base}[(\text{index}+\text{modifier}) \bmod 4]$$

Where the modifier is derived from the chaotic sequence. The inverse transformation is applied during decryption.

D. *Chaotic Diffusion Using XOR Operation:*

After the DNA encoding step, a chaotic diffusion step is performed using an XOR operation with a second Lorenz-generated chaotic sequence:

$$I' = I \square \text{ (chaotic sequence} \times 255)$$

This guarantees that even a minor change in the original image spreads across all pixels, significantly enhancing security. The same XOR operation is applied during decryption to reverse the diffusion.

E. Encryption Process:

1) The encryption consists of four rounds to maximize security:
2) First Fisher-Yates Shuffle – The image pixels are shuffled based on the first chaotic sequence.
3) First DNA Encoding and Modification – Each pixel undergoes DNA encoding and chaotic modification.
4) Second Fisher-Yates Shuffle – The image undergoes a second pixel permutation based on a second chaotic sequence.
5) Second DNA Encoding and Modification – The modified DNA sequence is again altered using a chaotic sequence.
6) Finally, an XOR-based chaotic diffusion step is performed, leading to the final encrypted image.

F. Decryption Process:

1) The decryption process here is the exact reverse of the encryption process and follows these steps:
2) Reverse Chaotic Diffusion – The XOR operation is applied again to restore the pre-diffusion state.
3) Reverse DNA Decoding (Second Round) – DNA modifications are reversed using negative shifts.
4) Reverse Fisher-Yates Unshuffle (Second Round) – The pixel shuffling is undone using the inverse permutation.
5) Reverse DNA Decoding (First Round) – The first round of DNA modifications is reversed.
6) Reverse Fisher-Yates Unshuffle (First Round) – The final unshuffling restores the original pixel order.

Each step is executed using the same chaotic sequences derived from the user-provided password, ensuring a successful decryption only if the correct key is used.

And the below is the flowcharts representing the flow of the encryption and decryption processes followed to encrypt/ decrypt the image properly.

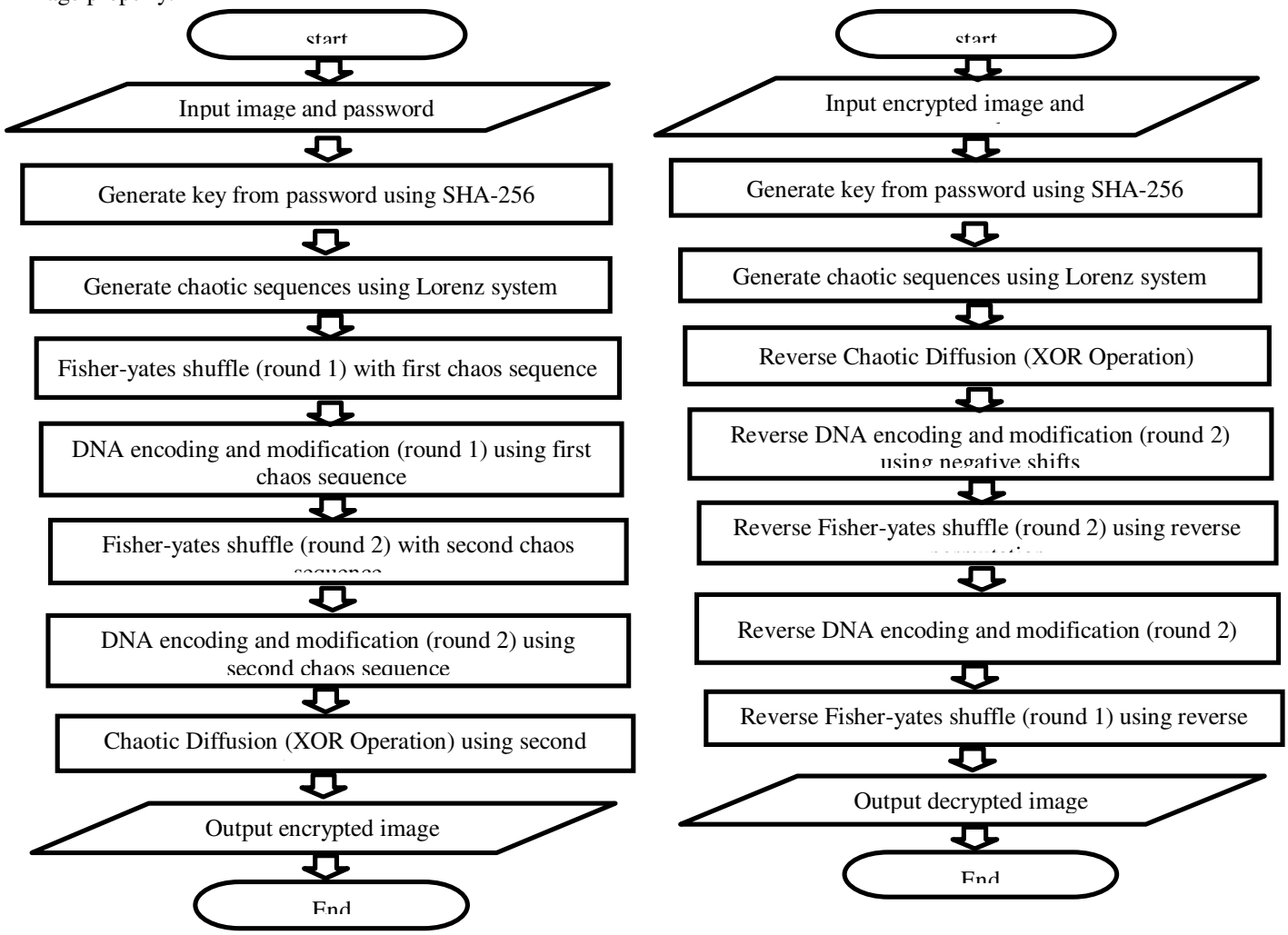


FIG 3: FLOW CHARTS OF ENCRYPTION AND DECRYPTION RESPECTIVELY

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

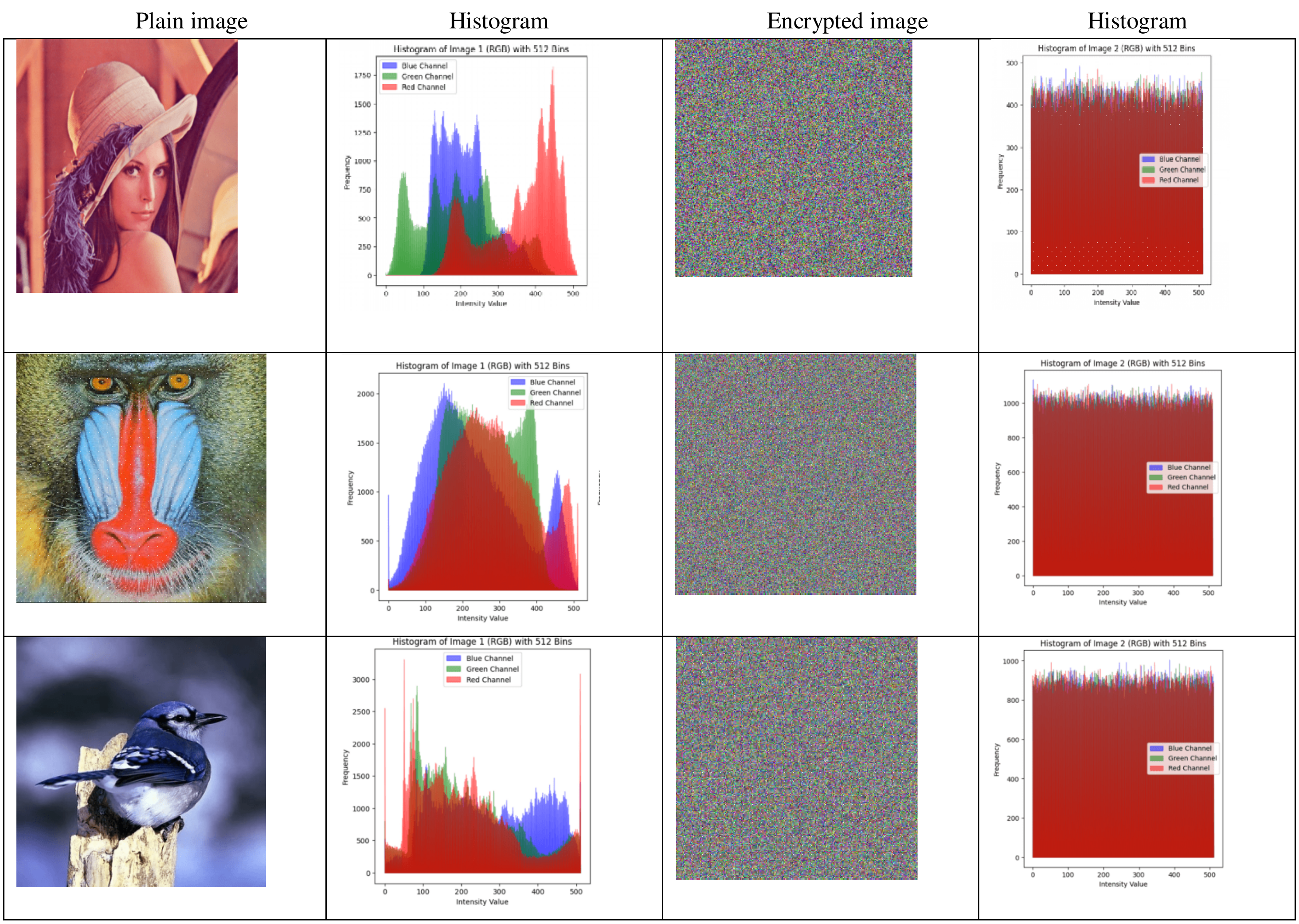
IV. RESULTS AND PERFORMANCE ANALYSIS

A. *Histogram Analysis:*

In image encryption, histogram analysis assesses the uniformity of pixel intensity distribution in an encrypted image. The histogram depicts the frequency of pixel values (ranging from 0 to 255 for 8-bit grayscale images). An effective encryption algorithm should generate a uniform histogram, ensuring an even distribution of pixel intensities. This uniformity prevents attackers from detecting patterns that could expose information about the original image.

For an unencrypted (plain) image, the histogram typically exhibits noticeable peaks and variations, reflecting the structure and redundancy of the image. However, after encryption, the histogram should be flat and random-like, indicating that pixel values are well-distributed across all intensity levels. The similarity of histograms can be evaluated using the Chi-Square test to determine whether the encrypted image follows a uniform distribution. The significance of histogram analysis lies in its ability to detect weaknesses in encryption algorithms. If an encrypted image retains noticeable peaks in its histogram, it suggests that the encryption is not fully disrupting the pixel structure, resulting in vulnerable to statistical attacks.

In the fig 4, the histogram analysis between the plain image and its respective encrypted image using the proposed methodology has been presented. These histograms shows that there are no noticeable peaks in it for an attacker to get to assume the pattern of the original image providing a high encryption security.



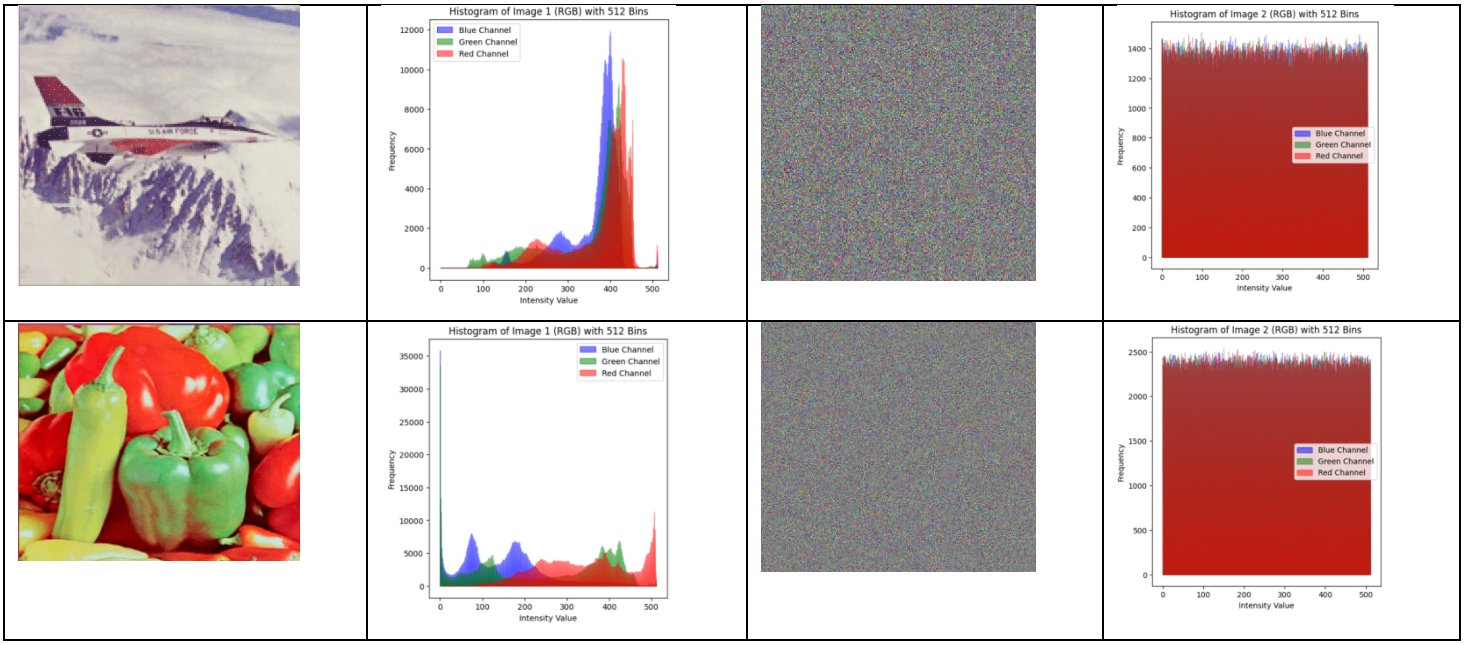


Fig 4: Histogram Analysis of Source and Encrypted images

B. Entropy Analysis:

Entropy analysis is a fundamental metric for evaluating the randomness and unpredictability of pixel values in an encrypted image. A well-encrypted image should exhibit high entropy, approaching the theoretical maximum, to ensure the strong resistance and uniform pixel distribution to statistical attacks. Shannon entropy is the most commonly used measure for this purpose and is calculated as:

$$\mathbf{H(X) = -\sum_{i=1}^{n} P(x_i)\ log_2 P(x_i)}$$

Where $P(x_i)$ is the occurrence probability of pixel value x_i. The maximum entropy for any 8-bit grayscale image is 8 bits, indicating a completely random distribution. If the entropy value is significantly lower, it suggests patterns in the encrypted image, making it vulnerable to cryptanalysis.

The significance of entropy analysis lies in its ability to verify the security strength of encryption algorithms. A high entropy value (close to 8 bits) ensures that encrypted images exhibit near-random characteristics, making them resistant to information leakage. If the entropy is low, the encryption method may be producing predictable patterns, which could be exploited by attackers.

Table 1 below is the entropy analysis of the source image and the encrypted image generated by using the proposed methodology. And the values are very optimal as the entropy of encrypted image is almost equal to 8, thus the image generated is highly random.

Sample Images	Color channels	Source image	Encrypted image
Lena	Red	7.250595888	7.998417682
	Green	7.589723689	7.998325404
	Blue	6.938571598	7.998356614
	Avg	7.259630392	7.998366567
Baboon	Red	7.752819021	7.999227897
	Green	7.465408947	7.999334076
	Blue	7.766575378	7.999293192
	Avg	7.661601115	7.999285055
Bird	Red	7.732382885	7.999203369
	Green	7.765874523	7.999169286

	Blue	7.878771695	7.99912128
	Avg	7.792343034	7.999164645
Plane	Red	6.712918677	7.999536705
	Green	6.916311436	7.999433823
	Blue	6.442145026	7.999468618
	Avg	6.690458379	7.999479715
Peppers	Red	7.530244343	7.999728772
	Green	7.51139617	7.999739514
	Blue	7.238032607	7.999699671
	Avg	7.426557707	7.999722652

Table 1: Entropy analysis of plain and encrypted image

C. Correlation Analysis:

Correlation analysis is a crucial metric for assessing the randomness and security of encrypted images. It quantifies the relationship between adjacent pixels in an image before and after encryption. In plaintext images, adjacent pixels typically exhibit high correlation, meaning their values are closely related. However, an effective encryption algorithm should disrupt this correlation, making the encrypted image appear random. The correlation coefficient (r) between two adjacent pixels (horizontal, vertical, or diagonal) is computed as:

$$\mathbf{r} = \frac{\sum_{i=1}^{n}(x_i-\mu_x)(y_i-\mu_y)}{\sqrt{\sum_{i=1}^{n}(x_i-\mu_x)^2}\sqrt{\sum_{i=1}^{n}(y_i-\mu_y)^2}}$$

Where x_i and y_i are pixel values of adjacent pixels, μ_x and μ_y are their mean values, and N is the number of pixel pairs. In an unencrypted image, r is close to 1, while in a well-encrypted image, it should be close to 0, indicating no correlation.

The significance of correlation analysis lies in its ability to verify encryption strength. A high correlation in the original image confirms redundancy, while a low correlation in the encrypted image ensures that encryption effectively disperses pixel values, preventing statistical attacks.

The table 2 below presents the correlation coefficients of source and encrypted images, and the values obtained are very much optimal proving that this methodology is reliable.

Sample Images	Color channels	Source image			Encrypted image		
		H	V	D	H	V	D
Lena	Red	0.9656	0.9824	0.9471	0.000746485	-0.000643474	-0.000114036
	Green	0.9508	0.9741	0.9300	-0.001570239	0.001077985	0.00234068
	Blue	0.9279	0.9550	0.9042	-0.003483641	0.003562376	-0.001742043
	Avg	0.9481	0.9705	0.9271	-0.001435798	0.001332296	0.000161534
Baboon	Red	0.9285	0.8648	0.8423	0.001600000	-0.001200000	0.002400000
	Green	0.8937	0.8014	0.7662	0.002500000	-0.001500000	0.000900000
	Blue	0.9366	0.8839	0.8621	-0.00300000	0.001700000	0.001700000
	Avg	0.9196	0.8500	0.8235	0.000366667	-0.000333333	0.001666667
	Red	0.9622	0.9629	0.9348	-0.001300000	-0.001900000	-0.000500000

Bird	Green	0.9587	0.9594	0.9287	-0.001700000	0.003000000	-0.000800000
	Blue	0.9641	0.9643	0.9380	0.000800000	0.000500000	0.001100000
	Avg	0.9616	0.9622	0.93383	-0.000733333	0.000533333	-0.000066700
Plane	Red	0.9855	0.9890	0.9747	-0.000500000	-0.002300000	-0.003300000
	Green	0.9867	0.991	0.9781	0.002800000	-0.000200000	-0.000700000
	Blue	0.9803	0.9861	0.9679	-0.003400000	0.001300000	-0.001600000
	Avg	0.9842	0.9887	0.9735	-0.000366667	-0.000400000	-0.001866667
Peppers	Red	0.9974	0.9961	0.9933	0.000200000	-0.000300000	0.000900000
	Green	0.9977	0.9968	0.9941	-0.000800000	0.000600000	0.001400000
	Blue	0.9965	0.9952	0.9914	-0.001900000	-0.002200000	0.000100000
	Avg	0.9972	0.9960	0.9929	-0.000833333	-0.000633333	0.000800000

Table 2: Correlation coefficients analysis of plain and encrypted images

D. Chi-Square Analysis:

Chi-Square (X^2) analysis is a crucial statistical test used to evaluate the randomness and uniformity of pixel distributions in an encrypted image. A well-encrypted image must demonstrate a nearly uniform pixel intensity distribution, ensuring that no patterns can be exploited by attackers. The Chi-Square test compares the observed pixel frequency with the expected frequency to determine whether deviations are statistically significant. It is calculated as:

$$\mathbf{X^2 = \sum \frac{(O_i - E_i)}{2E_i}}$$

Where O_i represents the observed frequency of pixel values, and E_i is the expected frequency (assuming uniform distribution). A higher chi-square value suggests deviations from randomness, whereas a lower chi-square value (closer to the theoretical threshold) implies that the encryption algorithm has effectively randomized all the pixel values.

The significance of chi-square analysis lies in its ability to assess the statistical uniformity of encrypted images. A well-designed encryption algorithm should produce a chi-square value close to the critical value at a chosen significance level (e.g., 0.05), confirming that the pixel distribution follows a uniform pattern. If the chi-square test reveals non-uniformity, it suggests potential weaknesses in the encryption process, making the cipher vulnerable to statistical attacks.

In the table 3 below, the chi-square values of source image and the encrypted image are evaluated separately. Chi-square values of encrypted images are very low approximately equal to 250, this indicates a large difference between the adjacent pixels. Whereas the source image's values are very high, which means the difference between adjacent pixels is very low. The outcomes of the proposed scheme proves that this methodology generates a encrypted image whose distribution of pixels is very random, thus providing robustness and a high encryption security.

Sample Images	Color channels	Source image	Encrypted image
Lena	Red	106739.8816	238.4230670
	Green	48665.9595	252.4290174
	Blue	149283.1511	248.6113499
	Avg	101562.9974	246.4878114
	Red	73376.8594	281.0820000

Baboon	Green	146030.4277	241.7539000
	Blue	77251.0469	257.0820000
	Avg	98886.1113	259.9726333
Bird	Red	86877.6222	255.2333000
	Green	78160.9644	265.2000000
	Blue	31865.7933	281.0000000
	Avg	65634.7933	267.1444333
Plane	Red	887906.6143	227.8860000
	Green	741711.6782	278.3429000
	Blue	1167032.5610	261.7339000
	Avg	932216.9511	255.9876000
Peppers	Red	339689.5477	231.0999000
	Green	714793.9194	221.4014000
	Blue	975719.8458	255.5411000
	Avg	676734.4376	236.0141333

Table 3: Chi Square analysis of plain and encrypted images

E. PSNR and MSE Analysis:

Peak Signal-to-Noise Ratio (PSNR) and Mean Squared Error (MSE) are essential metrics for evaluating encryption strength and decryption accuracy. MSE calculates the average squared difference between the source and encrypted images, ensuring that encryption introduces significant distortion. Calculated by:

$$\mathbf{MSE} = \frac{1}{\mathbf{m \times n}} \sum_{\mathbf{i=1}}^{\mathbf{m}} \sum_{\mathbf{j=1}}^{\mathbf{n}} (\mathbf{I(i,j)} - \mathbf{K(i,j)})^2$$

Where I(i,j) and K(i,j) are the pixel values of the original and encrypted/decrypted images, respectively. A high MSE enc confirms strong encryption, making the image unrecognizable, while a low MSE dec indicates successful recovery. PSNR, derived from MSE, measures reconstruction quality and is calculated as:

$$\mathbf{PSNR = 10\ log10\ (\frac{M^2}{MSE})}$$

Where M is the maximum pixel intensity value (Generally 255 for 8-bit images). A low PSNR for the encrypted image (PSNR enc) (e.g., below 10 dB) confirms strong security, whereas a high PSNR for the decrypted image (PSNR dec) (e.g., above 40 dB) ensures accurate decryption.

In the table 4 below, the PSNR and MSE values of both encryption and decryption processes are tabulated. And the outcomes are quite satisfying i.e., the encryption has low PSNR and high MSE, and the decryption has high PSNR and low MSE. Which means the encryption and reconstruction (decryption) quality is very high using this proposed scheme.

Sample Images	Color channels	Encryption process		Decryption process	
		MSE	PSNR (dB)	MSE	PSNR (dB)
Lena	Red	10546.7453	7.90	0	inf
	Green	9037.0734	8.57	0	inf

	Blue	7071.4796	9.64	0	inf
	Avg	8885.0994	8.64	0	inf
Baboon	Red	8713.3698	8.73	0	inf
	Green	7578.5698	9.33	0	inf
	Blue	9300.2956	8.45	0	inf
	Avg	8530.7451	8.82	0	inf
Bird	Red	10225.2288	8.03	0	inf
	Green	9760.6545	8.24	0	inf
	Blue	9964.0042	8.15	0	inf
	Avg	9983.2958	8.14	0	inf
Plane	Red	10548.0653	7.90	0	inf
	Green	10139.2868	8.07	0	inf
	Blue	9181.8598	8.50	0	inf
	Avg	9956.4014	8.15	0	inf
Peppers	Red	10138.6655	8.07	0	inf
	Green	11443.4197	7.55	0	inf
	Blue	11057.6178	7.69	0	inf
	Avg	10879.9010	7.76	0	inf

Table 4: MSE and PSNR analysis of encrypted and decrypted images

F. NPCR and UACI Analysis:

Number of Pixel Change Rate (NPCR) and Unified Average Changing Intensity (UACI) are essential metrics for assessing an encryption algorithm's sensitivity to minor alterations in the plaintext image. These metrics measure how significantly an encrypted image changes when a single pixel in the original image is modified, ensuring that the encryption scheme exhibits strong avalanche effects—a critical property for security.

NPCR calculates the percentage of pixels that differ between two encrypted images when a single pixel in the plaintext image is altered, indicating the algorithm's effectiveness in diffusing changes throughout the encrypted output. It is calculated as:

$$\mathbf{NPCR}=\frac{\sum_{i=1}^{m}\sum_{j=1}^{n}D(i,j)}{m\times n}\times 100\%$$

Where, D(i,j)=1 if C1(i,j)≠C2(i,j), otherwise D(i,j)=0. Here, C1(i,j) and C2(i,j) represents the pixel values of two images corresponding to the original and encrypted plain text images. A high NPCR value (typically above 99%) implies that the encryption algorithm is highly sensitive to a very small changes in the input, ensuring strong diffusion

UACI quantifies the average intensity difference between two encrypted images after modifying a single pixel in the plain text. It is given by:

$$\mathbf{UACI}=\frac{1}{m\times n}\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|C1(i,j)-C2(i,j)|}{255}\times 100\%$$

Where 255 is the maximum pixel value possible. A high UACI value (typically between 33% and 35%) indicates a strong encryption algorithm that effectively distributes intensity variations, preventing patterns from being detected by attackers.

NPCR and UACI are crucial metrics for evaluating the robustness of an encryption scheme against differential attacks. A high NPCR and UACI indicate that even minor changes in the original image result in significant, unpredictable alterations in the encrypted output, making it extremely challenging for attackers to extract meaningful information.

In the table 5 above, the NPCR and the UACI values between the source (plain) and encrypted image are noted. And the values are nearer to the optimum values i.e., NPCR values are closer to 100% and the UACI values are closer to the 33% which means that the proposed scheme is highly resistant to different cryptanalysis and has a very good diffusion resulting in strong encryption strength.

Sample Images	Color channels	UACI	NPCR
Lena	Red	32.87944496	99.64646465
	Green	30.56665957	99.5959596
	Blue	27.5359571	99.64187328
	Avg	30.32735387	99.62809917
Baboon	Red	30.09584844	99.6383667
	Green	28.33591402	99.61471558
	Blue	30.99984527	99.61929321
	Avg	29.81053591	99.62412516
Bird	Red	32.39893913	99.609375
	Green	31.68116808	99.60199653
	Blue	31.97694421	99.60503472
	Avg	32.01901714	99.60546875
Plane	Red	32.89903998	99.60502794
	Green	32.26397038	99.60164971
	Blue	30.78719079	99.61037681
	Avg	31.98340038	99.60568482
Peppers	Red	32.26810098	99.61381783
	Green	34.26275551	99.62082451
	Blue	33.68905187	99.61886915
	Avg	33.40663612	99.61783716

Table 5: NPCR and UACI analysis between plain and encrypted images

V. CONCLUSION

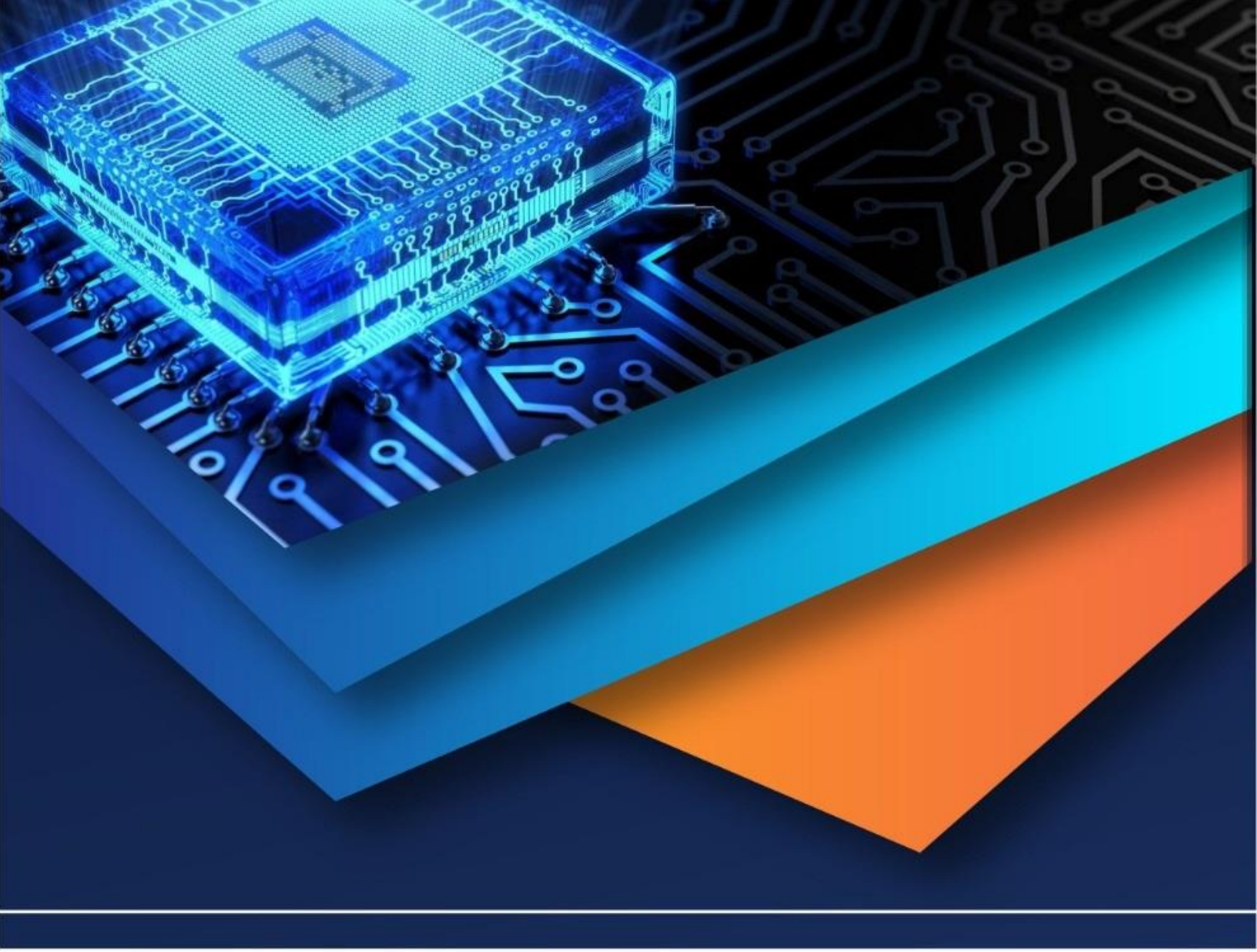
In this work, we propose a robust image encryption and decryption framework that integrates high-dimensional chaotic sequences, DNA encoding, and the Fisher-Yates shuffle to enhance security. The encryption process utilizes the Lorenz chaotic system to generate pseudo-random sequences, which drive pixel shuffling and DNA-based transformations. Multiple rounds of Fisher-Yates shuffling and chaotic diffusion further strengthen security by ensuring high resistance against statistical and differential attacks.

The decryption process accurately reverses the encryption steps using the same chaotic sequences derived from the password, ensuring precise image reconstruction. DNA encoding introduces an additional layer of complexity, while XOR-based chaotic diffusion increases randomness in pixel values, further enhancing security.

The proposed encryption scheme demonstrates strong security properties, including high key sensitivity, robustness making it resistant to brute-force attacks, and statistical attacks. This conclusion is supported by performance evaluations using key metrics such as entropy, correlation coefficient, chi-square test, histogram analysis, NPCR, and UACI analysis. In all these metrics, the proposed scheme achieved optimal values, outperforming many existing encryption methods.

REFERENCES

[1] Es-Sabry, Mohammed, Nabil El Akkad, Mostafa Merras, Khalid Satori, Walid El-Shafai, Torki Altameem, and Mostafa M. Fouda. "Securing images using high dimensional chaotic maps and DNA encoding techniques." IEEE Access (2023).
[2] Allawi, Salah Taha, Yasmin Makki Mohialden, and Nadia Mahmood Hussien. "Protection of the Image Data by Using Chaotic Maps and DNA Sequence." International Journal of Intelligent Engineering & Systems 17.4 (2024).
[3] Zhu, Jianjun, and Zhao Jian'E. "A novel four-dimensional hyperchaotic system and DNA encoding method for image encryption." Int. J. Netw. Secur. 26.1 (2024): 43-50.
[4] Peng, Jun, et al. "Research on a novel image encryption algorithm based on the hybrid of chaotic maps and DNA encoding." 2013 IEEE 12th International Conference on Cognitive Informatics and Cognitive Computing. IEEE, 2013.
[5] Amani, Hamid Reza, and Mahdi Yaghoobi. "A new approach in adaptive encryption algorithm for color images based on DNA sequence operation and hyper-chaotic system." Multimedia Tools and Applications 78 (2019): 21537-21556.
[6] Zhang, Chaoxia, et al. "Double image encryption algorithm based on parallel compressed sensing and chaotic system." IEEE Access (2024).
[7] He, Qiji, Peiya Li, and Yanyixiao Wang. "A Color Image Encryption Algorithm Based on Compressive Sensing and Block-based DNA Coding." IEEE Access (2024).
[8] Zhu, Shenli, Xiaoheng Deng, Wendong Zhang, and Congxu Zhu. "Image encryption scheme based on newly designed chaotic map and parallel DNA coding." Mathematics 11, 1 (2023): 231.
[9] Bencherqui, Ahmed, Mohamed Amine Tahiri, Hicham Karmouni, Mohammed Alfidi, Saad Motahhir, Mohamed Abouhawwash, S. S. Askar, Shuhuan Wen, Hassan Qjidaa, and Mhamed Sayyouri. "Optimal algorithm for color medical encryption and compression images based on DNA coding and a hyperchaotic system in the moments." Engineering Science and Technology, an International Journal 50 (2024): 101612.
[10] Cao, Guanghui, et al. "Image Encryption Based on a Coined Chaotic System and High-intensity Encryption Primitives." IEEE Access (2024).
[11] Alexan, Wassim, Dina El-Damak, and Mohamed Gabr. "Image encryption based on fourier-DNA coding for hyperchaotic chen system, chen-based binary quantization S-box, and variable-base modulo operation." IEEE Access (2024).
[12] Girdhar, Ashish, Himani Kapur, and Vijay Kumar. "A novel grayscale image encryption approach based on chaotic maps and image blocks." Applied Physics B 127 (2021): 1-12.
[13] Zang, Hongyan, Mengdan Tai, and Xinyuan Wei. "Image encryption schemes based on a class of uniformly distributed chaotic systems." Mathematics 10.7 (2022): 1027.
[14] Zhu, Congxu. "A novel image encryption scheme based on improved hyperchaotic sequences." Optics communications 285.1 (2012): 29-37.
[15] Yang, Na, et al. "Medical image encryption based on josephus traversing and hyperchaotic lorenz system." Journal of Shanghai Jiaotong University (Science) 29.1 (2024): 91-108.
[16] Li, Chengqing, Shujun Li, Guanrong Chen, and Wolfgang A. Halang. "Cryptanalysis of an image encryption scheme based on a compound chaotic sequence." Image and Vision Computing 27, 8 (2009): 1035-1039.
[17] Zhao, Jingbo, Tian Zhang, Jianwei Jiang, Tong Fang, and Hongyang Ma. "Color image encryption scheme based on alternate quantum walk and controlled Rubik's Cube." Scientific Reports 12, 1 (2022): 14253.
[18] Mfungo, Dani Elias, Xianping Fu, Xingyuan Wang, and Yongjin Xian. "Enhancing image encryption with the Kronecker Xor product, the Hill Cipher, and the Sigmoid Logistic Map." Applied Sciences 13, 6 (2023): 4034.
[19] Elazzaby, Fouzia, and Nabil EL Akkad. "Advanced encryption of image based on S-box and chaos 2D (LSMCL)." 2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET). IEEE, 2020.
[20] Es-Sabry, Mohammed, Nabil El Akkad, Mostafa Merras, Abderrahim Saaidi, and Khalid Satori. "A new color image encryption algorithm using multiple chaotic maps with the intersecting planes method." Scientific African 16 (2022): e01217.
[21] Es-Sabry, Mohammed, Nabil El Akkad, Mostafa Merras, Abderrahim Saaidi, and Khalid Satori. "A new image encryption algorithm using random numbers generation of two matrices and bit-shift operators." Soft Computing 24 (2020): 3829-3848.
[22]]Neamah, Ammar Ali. "An image encryption scheme based on a seven-dimensional hyperchaotic system and Pascal's matrix." Journal of King Saud University-Computer and Information Sciences 35, no. 3 (2023): 238-248.
[23] Yan, Shaohui, Lin Li, Binxian Gu, Yu Cui, Jianjian Wang, and Jincai Song. "Design of hyperchaotic system based on multi-scroll and its encryption algorithm in color image." Integration 88 (2023)



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY