

## A FAST AND ROBUST COLOUR IMAGE ENCRYPTION SCHEME USING HUFFMAN COMPRESSION, 5D CHAOTIC MAP AND DNA ENCODING

**Radha Seelaboyina and Dr. Rais Abdul Hamid Khan**

Department of Computer Science & Engineering, Dr. A. P. J. Abdul Kalam University,  
Indore (M.P.) - 452010, India

Corresponding Author Email : [radha.seelaboyina@gmail.com](mailto:radha.seelaboyina@gmail.com)

### **Abstract:**

The greatest issue in the age of information technology is safeguarding the real-time colour digital image. In this work, a colour picture encryption system based on the compression-then-encryption approach is developed to safeguard digital images. The proposed method is a hybrid combination of the Chaos techniques for key generation, Huffman Encoding/compression for compression and avoiding colour decomposition, scrambling for more confusion and Deoxyribonucleic acid (DNA) Encoding for reducing storage size. To enhance security, scrambled data are converted to the DNA sequence, make addition (ADD) operation, and apply the complementary rules to attain the cipher image. Experimental results have been proved as robustness, fast execution time, and high security against attacks, malicious attacks, differential attacks, and statistical attacks. Furthermore, results show that the proposed work as robustness, fast execution time and high security than the existing colour image encryption schemes.

**Keywords:** Deoxyribonucleic acid (DNA); Image Encryption; 5D chaotic map; Huffman Compression; Scrambling; Robustness

### **1. INTRODUCTION**

Information security is a vast field in the IT industry. It would face the greater issue which affects national security, social media, and personal information. An image has some properties such as high redundancy, large data space storage, vividness and high resource allocation compared to text. Due to the larger data space and high redundancy, traditional techniques, especially 3- DES, RSA, and AES, were not well suited to the image because of their high computational complexity, high resource utilization, and better security [1][2] (Swathi, Lahari and Bindu). The various techniques related to images and text were explained to calculate the time consumption of 3- Data Encryption Standard (DES), Rivest Shamir Adleman (RSA), Advanced Encryption Standard(AES). The conclusion was that the high time complexity while using images[4][5]. Another survey paper had explained the various techniques applied to encryption, such as sudoku, Exclusive-OR (XOR), scrambling, permutation, and Chaos Theory [6][7][8]. Hence, the chaos technique would be far superior in security and well-suited to key generation for images. The research focuses primarily on Chaos techniques for key generation, Huffman Encoding for compression and avoiding colour decomposition, DNA Encoding for storage size reduction. A chaotic system is a famous and next-generation approach with a high level of intricacy and sensitivity to initial parameters. It possesses traits like dynamics, ergodicity, replication, and PRP to strengthen the encryption and fend off attackers [9]. It offers a larger, more secure key area. Greater security is provided by larger key spaces. For image

encryption, a chaotic algorithm is developed and employed to encode the data. It generates the PRP based on non-linear dynamics [10]. Client Image data and their keys are transmitted over the cloud directly. CSP could take over the data controller in the cloud. So, clients can face privacy issues due to the loss of control[11]. Thereafter, some techniques were introduced which is passed the partial encrypted image to the cloud. It creates huge complexity while selecting the partial image data . Later, the image is encrypted by the user and stored in the cloud, and minimised the computational complexity to ensure privacy. Image Owner holds the metadata information for privacy assurance. Hence, the proposed privacy-Privacy preserving Chaos-based Symmetric and Efficient Encryption Technique (SEET) is evaluated by using the simple and light-weight method to ensure security and privacy. It ensures secrecy, integrity, and resistance to threats. [13][12]. Xu *et al.* generated a novel method which is encrypted the image by scrambling at the bit level using a chaotic map and ensures high performance compared with other schemes [14].

Brindha*et.al* developed image encryption efficiently and used a compression approach that is based on the widely known Chinese remainder theorem [15]. Yuan *et.al* implemented a new encryption with the movement of pixels by diagonal method to encrypt, and a 5D chaotic map generated the secret keys to achieve the high-level cryptosystem. It has been proven to be efficient encryption/decryption with various security analyses and performance metrics. It has proved that the parallel system as fast and strong[16]. Sun *et.al* implemented the different scrambling and encryption techniques for an image. Further, DNA encoding is applied to achieve robustness against attacks[17]. The 5D hyper-chaotic system generated the secret keys and would be sensitive, reliable, secure, and robust against malicious attacks. It was demonstrated that its time speed was reduced when compared to other schemes. Wu *et.al* presented 1D colour image encryption with numerous upgrades and DNA operations[18]. It was proved to be robust, secure from geometric attacks. Samiullah *et al.* implemented the novel image encryption with the combination of three chaotic systems, chaotic key generator, scrambling technique, and SHA(Secure Hash Algorithm)with DNA sequence[19][20].It ensures multilevel security to improve the confusion and diffusion to achieve high security. Mondal *et al.* implemented image encryption with a simple technique (light-weight method) and better security performance by using the chaos system and DNA operations[21][22]. Ravichandran *et al.* implemented the different chaos-based algorithm which is immutable and applied to images[23][24]. It proved that its algorithm was strong and reliable.

In the remaining part of the paper, Section 2 outlines the contribution, while Section 3 details Huffman compression, the 5D hyperchaotic system, and DNA coding. The suggested encryption system is represented in Section 4. Section 5 summarises the simulation findings and security analysis. The suggested decryption mechanism is shown in Section 6. Section 7 contrasts the proposed encryption/decryption system to the existing approach.

**2. MATERIALS AND METHODS**

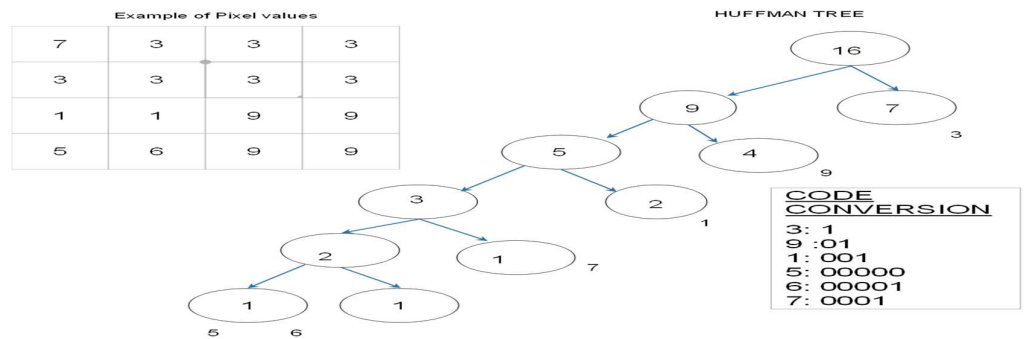
**2.1. CONTRIBUTIONS**

- (i). Normally, an original image was decomposed into three colour formations (RGB) in many research works. But the proposed work is never decomposed.
- (ii). In contrast to current grey-level and colour picture encryption techniques, the suggested encryption method minimizes execution time.
- (iii). The proposed work attains the minimum storage utilization because of using DNA technique.
- (iv). Robustness is proved by using different techniques such as Huffman Encoding, DNA techniques, chaotic key generation, and scrambling.

**2.2. PRELIMINARY WORKS**

**2.2.1. Huffman Encoding**

Huffman Coding is an entropy-based technology which is lossless compression and basically applied to the text and well suited too[24].The sample Huffman tree representation is shown as Fig.1.



**Fig. 1: Example of Huffman tree**

Fig.1 shows the Huffman tree of 4\*4 pixel values and produces the code conversion based on the counting of values of pixel. The plain and original colour image is first applied to the Huffman coding without decomposition of three different colours. Even though, it is retrieved the colour image without any loss. The input image *I* is reduced to ‘n’ number of pixel values based on the image size i.e.4\*4 image size = 16 pixel values.

The following is the likelihood of occurrence of a specific pixel intensity value:

$$prob\_pixel_j = \frac{freq\_pixel_j}{tot\_pixel} \tag{1}$$

$$\sum_{j=1}^m prob\_pixel_j = 1 \tag{2}$$

Where  $j = \{1, 2 \dots m\}$  ‘m’ acts as distinct pixel intensity in an image. Where  $prob\_pixel_j$  acts as a probability of a specific pixel ‘j’ in an image,  $freq\_pixel_j$  represents the number of frequency (tuple) of a specific pixel ‘j’ with a certain intensity value i.e. ‘3’ pixel value occurs ‘7’ ( $freq\_pixel_j$ ) times,  $tot\_pixel$  represents the total number of pixels in an image (16 values). The  $freq\_pixel_j$  act as the leaf nodes to construct the Huffman tree.

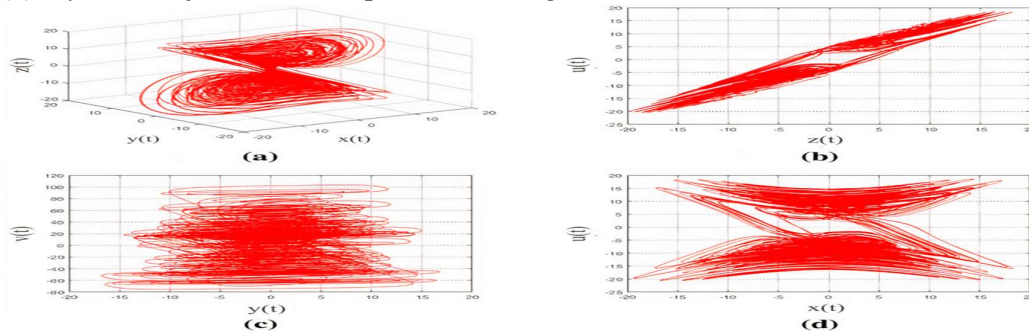
The last two LSB leaf nodes are combined and merged into a new node. Then, sort the nodes based on the new probability nodes in the ‘lookup table’. Continue the processes till it gets the single node with probability 1.0. The final node is known as ‘root’. Thereafter, move to the tree backwards (right to left) and different bits are assigned to the different branches. The binary code is calculated from the  $freq\_pixel_j$  and Huffman tree.

**2.2.2. 5D Chaotic System**

The following equation of 5D Hyperchaotic system[25] could be generated below

$$\begin{cases} x = (a(y - x)) + (yzu) \\ y = (b(x + y)) + v - (xzv) \\ z = (-cy) - (dz) - (eu) + (xyu) \\ u = -fu + xyz \\ v = -g(x + y) \end{cases} \quad (3)$$

where  $a, b, c, d, e, f$  are control parameters[10]. Consider the control parameters value as “ $a=30, b=10, c=15.7, d=5, e=2.5, f=4.45$  and  $g=38.5$ ” to generate chaotic sequences as Equation (3). System Trajectories are represented in Fig.2.



**Fig. 2: System Trajectories of Equation(3) with initial values and control parameters (a) x-y-z phase portrait; (b) z-u phase portrait; (c) y-v phase portrait; (d) x-u phase portrait.**

**2.2.3. DNA Encoding Scheme**

Deoxyribonucleic acid (DNA)[20][19] molecules made up of nucleotides. It has four types such as A, C, G and T. A-Adenine, C- Cytosine, T- Thymine and, G - Guanine. DNA is two twisted stranded around each other. A,C,G and T are indicated as 00,11,10,11. DNA rules, DNA-ADD(Addition)operation, DNA-SUB(SUBtraction) operation and rule of complement, shown in Table 1,2 and 3, used for encoding/decoding. Table 1 shows the first four rules of DNA sequence.

Table 1 DNA Sequence rules

Rules	1	2	3	4
00	A	A	C	G
01	C	G	A	A
10	G	C	T	T
11	T	T	G	C

Table 2 DNA- ADD operation

ADD	A	C	G	T
A	A	C	G	T
C	C	G	T	A
G	G	T	A	C
T	T	A	C	G

Table 3 DNA- SUB operation

SUB	A	C	G	T
A	A	T	G	C
C	C	A	T	G
G	G	C	A	T
T	T	G	C	A

DNA complementary rule must satisfy the following conditions:

$$\left\{ \left\{ x \neq A(x) \neq A(A(x)) \neq A(A(A(x))) \mid x = A(A(A(A(x)))) \right\} \right\} \tag{4}$$

where  $A(x)$  represents as a base pair of  $x$  which differs at least one bit of  $x$ . The value of  $x$  has four times repeat of the  $A(x)$ . So, it represents  $x$  equals to the  $A(A(A(A(x))))$ .

### 2.3. Proposed Image Encryption Scheme

The proposed work with different formats such as JPG(Joint Photographic Graphics), JPEG (Joint Photographic Expert Graphics), GIFF(Graphics Interchange Format), BMP(Bitmap Image File), PNG(Portable Network Graphics), and so on, are evaluated to show the performance of the technique and time complexity. The collection of more than **1000 images** to test with various attacks such as malicious attacks, exhaustive attacks, brute-force attacks to assure the robustness, security analysis to reach the goal.

The proposed scheme is divided into three modules namely Huffman Compression, Scrambling and DNA-based Encryption Module. To obtain the encrypted image, each module has a distinct function. The 5D Hyperchaotic system is generated the chaotic key sequences that is to be applied for the image encryption process. Consider the image ‘P’ size as  $M*N$ . Then, the original image is passed into the three modules and finally produces the encrypted image.

**2.3.1. 5D Hyperchaotic Key generation**

Step 1: Generate the initial values x, y, z, u and v of 5D Hyperchaotic system[25] as follows:

$$\begin{cases} x = \text{mod}(x + y + z + u + v, 1) \\ y = \text{mod}(x + y, 1) \\ z = \text{mod}(y + z, 1) \\ u = \text{mod}(z + u, 1) \\ v = \text{mod}(u + v, 1) \end{cases} \quad (5)$$

Where  $x^0, y^0, z^0, u^0, v^0$  are the initial keys,  $\text{mod}(a,b)$  produces residue of ‘a’ divided by ‘b’.

Step 2: By using the initial keys and control parameters, 5D Hyperchaotic system is iterated MN times to avoid the transient response  $R_0$ .

$$R_0 = 400 + \text{mod}((x^0 + y^0 + z^0 + u^0 + v^0) - \lfloor (x^0 + y^0 + z^0 + u^0 + v^0) \rfloor * 10^{15}, 400)$$

Where  $R_0$  represents the transient response and generate the five chaotic sequences  $K_1, K_2, K_3, K_4, K_5$ . All keys are converted to fixed length float numbers rather than long strings.

$$\begin{aligned} K_1 &= \{x^1, x^2, \dots, x^{MN}\} \\ K_1 &= \text{abs}(x_i) - \lfloor \text{abs}(x_i) \rfloor * 10^{-15}, M - i) \\ K_2 &= \{y^1, y^2, \dots, y^{MN}\} \\ K_2 &= \text{abs}(y_j) - \lfloor \text{abs}(y_j) \rfloor * 10^{-15}, N - j) \\ K_3 &= \{z^1, z^2, \dots, z^{MN}\} \\ K_3 &= \text{abs}(z_i) - \lfloor \text{abs}(z_i) \rfloor * 10^{-15}, M - i) \\ K_4 &= \{u^1, u^2, \dots, u^{MN}\} \\ K_4 &= \text{abs}(u_j) - \lfloor \text{abs}(u_j) \rfloor * 10^{-15}, N - j) \\ K_5 &= \{v^1, v^2, \dots, v^{MN}\} \\ K_5 &= \text{abs}(v_i) - \lfloor \text{abs}(v_i) \rfloor * 10^{-15}, M) \end{aligned} \quad (6)$$

Where the range of ‘i’ and ‘j’ are 1 to M and 1 to N, respectively

Step 3: Combine and sort out the different key formats from the sequences such as  $\{K_1, K_2\}$  for pixel level scrambling,  $\{K_3, K_4\}$  for bit level scrambling and  $\{K_5\}$  for Dna Encoding

**Module 1: Huffman Compression**

Step 4: Consider the equal image (I) size M\*N as 256\*256, 512\*512 and 1028\*1028. A 2-D image is transferred to the 1D array for fast and easy processing. As refer an eqn. [1,2], generate the Huffman binary tree based on the sorted frequencies of distinct intensity values. Then

$$\text{Codeword } W(F) = \{c_1, c_2 \dots c_n\} \quad (7)$$

Where Codeword  $W(F)$  is the tuple of binary code values.  $c_j$  is the code values for  $\text{freq\_pixel}_j, j \in \{1, 2, \dots, m\}$ . Convert the binary code to its decimal sequence  $P$ .

$$P = \{p_1, p_2, \dots, p_{MN}\} \quad (8)$$

Module 2: Scrambling

### 2.3.2. Pixel Distribution

Step 5: The decimal sequences are distributed into different blocks as 10\*10 pixels. The remaining pixels of the distribution are to be extended with the constant pixel value to make the 10\*10 block. Block size B=100. The equation is as follows:

$$\{p_{MN-tot} \dots p_{MN}\} = Const \quad (9)$$

where  $tot = n(P)/100$ , Const is the positive integer,  $Const \leq M$ .  $M$  is the height of an image

### 2.3.3. Pixel-Level Scrambling

Step 6: Combine  $\{p_1, p_2, \dots, p_{MN-tot-1}\}\{p_{MN-tot} \dots p_{MN}\}$ . Suppose the two keys  $(K_1, K_2)$  are merged and applied for scrambling[26] by the following equation as

$$Scr(i) = mod(P(i) + \{K_1, K_2\}, M) \quad (10)$$

Where  $Scr(i)$  act as the scrambling based on the pixels,  $P$  acts as the pixel values  $P(i) = \{p_1, p_2, \dots, p_{MN}\}$  including remaining pixels,  $i=1,2,\dots,MN$ .

### 2.3.4. Bit-level Scrambling

Step 7: Transform the decimal sequence  $Scr, K_3, K_4$  into corresponding binary sequences. The two keys are merged and applied to the following equation

$$Bit\_Scr(i) = circshift[P(i), LSB\{K_3, K_4\}, \{K_3, K_4\}] \quad (11)$$

In this case,  $circshift[a, b, c]$  refers to a c-bit circular shift on the binary sequences  $a$ , Left/right circular shift are to be decided based on the b value as  $b=0$  or  $1$ ,  $LSB(m)$  means *the least significant bit of m*.

Module 3: DNA Encoding

DNA Encoding [20] is an encoding process which converts into DNA sequence and operation handled on it based on the DNA-ADD, DNA-SUB lookup table. DNA Encoding is divided into five different sections: DNA Sequence, DNA-ADD operation, DNA Rotation, DNA Complement and DNA combination.

### 2.3.5. DNA Sequence

Step 8: The binary codes  $Bit\_Scr$  are translated to DNA sequences based on lookup table, the DNA rules and decimal form of Key  $K_5$  converted to the binary form and DNA sequences.

$$BKey: Bin\_Key \leftarrow Bin\_code(K_5) \quad (12)$$

$$Key: BKey \rightarrow lookup(DNA\_rules) \quad (13)$$

$$C: Bit\_Scr \rightarrow Bit\_lookup(DNA\_rules) \quad (14)$$

Where  $Bin\_code(K_5)$  denotes the binary code of key  $K_5$   $lookup(DNA\_rules)$  converts the sequences of DNA (as A, T, G or C) as per the bit represented in  $Bit\_Scr$  ('00', '01', '10', '00').  $BKey$  act as the binary code of  $K_5$ .  $Key$  act as the DNA code key generated after look-up table of DNA rules.

**2.3.6. DNA-ADD Operation:**

Step 9: After the collectively sequence of DNA(as A,T,G or C) , perform the DNA-ADD as ref in Table 2, to get the DNA sequence F with the combination of C(i) in eqn.(14) and Key in eqn (13).

$$F(i) = C(i) + Key(i) \quad (15)$$

Where the range of 'i' from 1...MN, C(i) acts as a bit scrambling code and Key(i) acts as the Binary form of K<sub>s</sub>.

**2.3.7. DNA Complement**

Step 10: Consider BKey (12) as the key to complement F(i) as shown below.

$$F'(i) = E_{BKey(i)}(F(i)) = \begin{cases} F(i) & \text{if } Bkey(i) = 0 \\ E(F(i)) & \text{if } Bkey(i) = 1 \\ E(E(F(i))) & \text{if } Bkey(i) = 2 \\ E(E(E(F(i)))) & \text{if } Bkey(i) = 3 \end{cases} \quad (16)$$

Here E(x) denotes the base pair of, the range of 'i' from 1 to MN, F'(i) acts as a complement of F(i).

**2.3.8. DNA Rotation**

Step 11: Consider every 10\*10 blocks to be rotated by 90° degree. Rotation made by two times to get 180°. In Decryption, the same process is to be repeated to attain the original blocks.

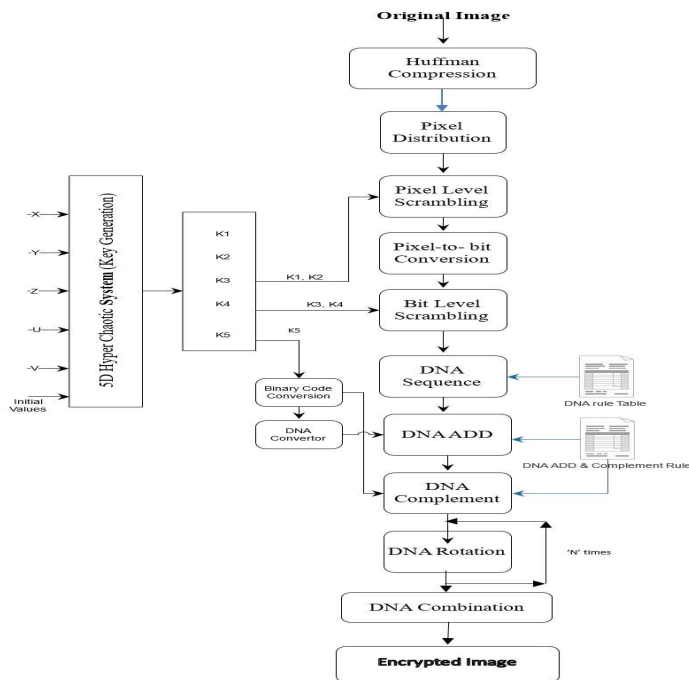
$$DNA\_rot = rot(F'(i), 90) \quad (17)$$

Where DNA\_rot denotes the DNA rotation, rot(F'(i), 90) denotes to rotate twice(90°) the 10\*10 blocks.

**2.3.9. DNA Combination**

Step 12: Combine all DNA blocks together to achieve the DNA Sequences. Finally, Decode F' converted to the binary form, convert to decimal sequence H. Fig. 3 displays the suggested colour Image Encryption Scheme.





**Fig. 3: The Proposed Encryption Scheme**

### 3. RESULTS AND DISCUSSION

Experimental simulations are evaluated and established on the 5D hyper-chaotic image encryption scheme by Python 2.7 on a personal computer with YOGA 520 system, 8GB RAM, and Intel core i3 processor. The description of Python as “Python is a high level, scripting language and easy to execute the code. It is simpler code, quick, understandable and open source”. Some of the results were executed in Matlab R2016a. The color plain images such as Lena, baboon, and fruit of different sizes are taken for the proposed scheme as shown in Fig. 3.

#### 3.1. Key Space

Key space is the major factor to determine the strength of the proposed image encryption scheme. If it is greater than  $2^{100}$ , it is resistant to exhaustive attacks. The suggested work's key calculation is based on the initial values. The initial values are 1.2356, 2.8905, 0.89648, 3.45797 and 0.45723. Rotation key represents any single character respectively. Thus, the total key space as  $(10^{-15})^6 = 10^{90}$  which is approximately equal to  $2^{298}$ . If it is more than  $2^{100}$ , it is proven to resist the exhaustive attacks and brute force attacks.

Table 4 Key Space Analysis


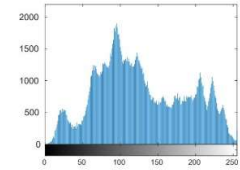

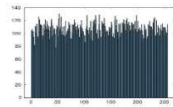

Crypto System	Ours	Ref[[27]	Ref[26]	Ref[[28]
Keys Space	$10^{90} (\approx 2^{298})$	$10^{56}$	$10^{60}$	$2^{128}$

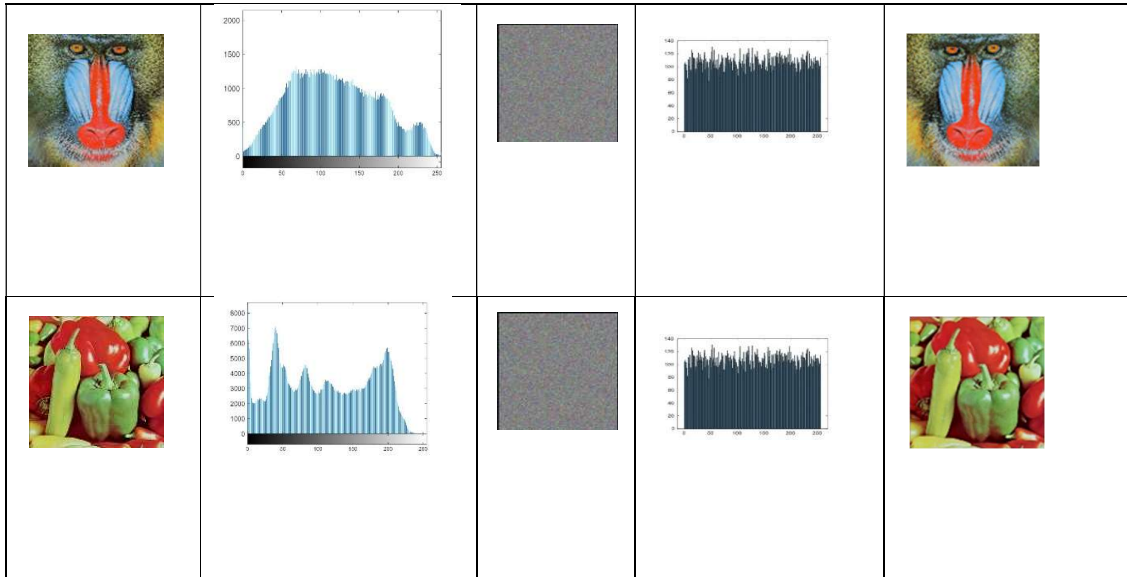
The key space of existing image encryption scheme compared with the proposed image encryption scheme is in Table 4. It has shown the proposed cryptosystem is larger when compared to Ref [28][27][26] and has chaotic properties such as 6 control parameters and key sensitivity. So, it proves to be robust while tested with exhaustive attacks and brute-force attacks.

### 3.2. The Histogram Analysis

Histograms are represented that the data are distributed with pixel/data values. Histogram(plain images) are non-uniformly distributed with pixel values. Histogram (cipher) are uniformly distributed and exhibits exhaustive attacks. Table 5 displays the plain images (256\*256) along with their histograms, the cipher image and its histogram, and the decrypted image with the correct key. It could be observed that the encrypted image is uniformly distributed and flat. Thus, it could withstand several attacks.

Table 5 The original image, its Histogram, Encrypted Image and its histogram, Reconstructed Image.

Image	Histogram of Plain Image	Cipher Image	Histogram of Cipher Image	Decrypted Image
				



### 3.3.Key Sensitivity Analysis

#### Key changes to attain encryption

Consider the only 5 initial keys with slight changes of one initial secret key to attain the 0.99 % of NPCR in encryption. As observed from the calculation below, encrypted images can cause greater differences because of slight changes in encryption keys. It is shown in Table 6.

**Table 6 NPCR score comparing the encrypted image with the correct key and the modified key**

Encryption Keys					NPCR(%)
X	Y	Z	U	v	
$x^0$	$y^0$	$z^0$	$u^0$	$v^0$	-
$x^0+10^{-15}$	$y^0$	$z^0$	$u^0$	$v^0$	99.55
$x^0$	$y^0+10^{-15}$	$z^0$	$u^0$	$v^0$	99.63
$x^0$	$y^0$	$z^0+10^{-15}$	$u^0$	$v^0$	99.57
$x^0$	$y^0$	$z^0$	$u^0+10^{-15}$	$v^0$	99.12
$x^0$	$y^0$	$z^0$	$u^0$	$v^0+10^{-15}$	99.14

#### Key changes to attain decryption

Consider the only 5 initial keys with slight changes of one initial secret key to attain the 0.99 % of NPCR in decryption. As observed from the calculation below, encrypted images can cause greater differences because of slight changes in decryption keys. It is shown in Table 7.

**Table 7 NPCR score comparing the decrypted image with the correct key and the modified key**

Decryption Keys					NPCR(%)
x	Y	Z	U	v	
$x^0$	$y^0$	$z^0$	$u^0$	$v^0$	-
$x^0+10^{-15}$	$y^0$	$z^0$	$u^0$	$v^0$	0.9941
$x^0$	$y^0+10^{-15}$	$z^0$	$u^0$	$v^0$	0.9931
$x^0$	$y^0$	$z^0+10^{-15}$	$u^0$	$v^0$	0.9934
$x^0$	$y^0$	$z^0$	$u^0+10^{-15}$	$v^0$	0.9963
$x^0$	$y^0$	$z^0$	$u^0$	$v^0+10^{-15}$	0.9912

**3.4. Correlation Coefficient**

The correlation coefficient is measured by the linear relationship between two variables [19]. A secure cryptosystem will reduce the high correlation between two neighbouring pixels.

The coefficient of correlation is measured as

$$r_{xy} = \frac{|Cov(x,y)|}{\sqrt{D(x)}\sqrt{D(y)}} \tag{18}$$

Where  $Cov(x,y) = \frac{1}{N} \sum_{i=1}^N ((x_i - E(x))(y_i - E(y)))$ ,  $E(x) = \frac{1}{N} \sum_{i=1}^N x_i$ ,  $D(x) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))^2$

**Table 8 Correlation Coefficient of two neighbouring pixels in different images**

Image Name	Image Size	Plain Image			Cipher Image		
		Horizonta l	Vertica l	Diagona l	Horizonta l	Vertica l	Diagona l
Lena	256*256	0.9595	0.9810	0.9456	-0.0011	-0.0009	0.0016
Baboon	256*256	0.9976	0.951	0.9610	-0.0009	-0.0014	-0.0015
Lena1	512*512	0.9596	0.9710	0.9656	-0.0011	-0.0010	-0.0015

Baboon 1	512*512	0.9755	0.9110	0.9556	-0.0015	-0.0007	0.0014
Lena2	1024*1024	0.9796	0.969	0.9710	-0.0009	-0.0011	-0.0010
Baboon 2	1024*1024	0.9467	0.951	0.9556	-0.0010	-0.0015	-0.0012

From these results, correlation coefficient of plain images with horizontal, vertical and diagonal are 0.9 and that of cipher images are negative values shown in table 8. So, it was concluded that the correlation of original images and cipher images are different. It was very hard to malicious, exhaustive attacks.

### 3.5. Information Entropy

It is designed to quantify the image quantity which is evaluated the uncertainty with a random variable. The proposed work is used the entropy-based technique as Huffman coding for the initial step of encryption. The entropy of the image has a positive correlation with its randomness.

The information entropy ‘s’ is defined as

$$H(s) = \sum_{i=0}^{2^N-1} p(s_i) \log_2 p(s_i) \tag{19}$$

where  $p(s_i)$  denotes the probability at the symbol  $s_i$ ,  $2^N$  represents to count the total number of possible symbols. In an ideal scenario, the information entropy corresponds to 8 bits for a random image. Information entropies values of encrypted images are calculated and shown in Table 9. The results in the proposed scheme are closer to 8 bits. So, it proves to resist entropy attacks.

**Table 9 Information Entropies**

Image Name	Image Size	Cipher Image
Lena	256*256	7.9973
Baboon	256*256	7.9154
Lena1	512*512	7.9311
Baboon1	512*512	7.9876
Lena2	1024*1024	7.9453
Baboon2	1024*1024	7.9898

### 3.6. Differential Attack

Attacks on the current image cryptosystem are what measure the changes in pixel rate and intensity. The popular parameters for measuring the plaintext sensitivity are NPCR (Number of Pixel Changing Rate) and UACI (Unified Average Changing Intensity) (Valander, M.Y et al). It is the ability to test against differential attacks. The cipher image is subject to differential attacks when plain images are slightly altered. The two parameters are defined as

$$NPCR = \frac{\sum_{i=1}^M \sum_{j=1}^N D(i,j)}{M \times N} \times 100\% \tag{20}$$

$$UACI = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N \frac{|C_1(i,j) - C_2(i,j)|}{255} \times 100\% \tag{21}$$

$$D(i,j) = \begin{cases} 1, & \text{if } C_1(i,j) \neq C_2(i,j) \\ 0, & \text{otherwise} \end{cases}$$

here ‘N’ denotes the image height and ‘M’ denotes the image width, and  $C_1$  and  $C_2$  are the ciphered versions of the original plain image and its changed image. The expected NPCR and UACI values are 99.6122% and 33.4636%. Table 10 shows the values of NPCR and UACI between two encrypted images. The experimental results concluded that even without any other modifications in the plain text, a tiny alteration in the plain text can cause the cipher image to reflect differently. It shows sensitivity to the plain text, which resists to plain attack and differential attack.

**Table 10 NPCR and UACI scores for the two encrypted image**

Image Name	Image Size	NPCR(%)	UACI(%)
Lena	256*256	99.61	33.46
Baboon	256*256	99.57	33.51
Lena1	512*512	99.25	33.67
Baboon1	512*512	99.45	33.85
Lena2	1024*1024	99.78	33.24
Baboon2	1024*1024	99.12	33.56

### 3.7.DECRYPTION

The decimal Sequence H converted to binary form and further converted to the DNA Sequence. After the key generation K1,K2,K3,K4 and K5, use the complement rules with K5 to attain intermediate encrypted image. Then, proceed with the DNA- SUB Operation. And continue with the DNA sequence, reverse process of scrambling and Huffman coding as ref in equation[1-17]. Finally, original image is retrieved without lossless.

$$\left\{ \begin{array}{l} H = F'(i) \\ F(i): DNA_{rot} = rot(F'(i), 90) \\ C(i) = F(i) - Key(i) \\ C(i) \rightarrow Bit\_scr(i) \\ Bit_{scr(i)} \rightarrow Scr(i) \\ Scr(i) \rightarrow P(i) \\ P(i) \rightarrow W(F) \\ W(F) \rightarrow I \end{array} \right. \quad (22)$$

[29]

Where  $i=1 \dots MN$ , H is a final decimal form. ‘I’ acts as the decrypted Image.

### 3.8. Time Analysis of Proposed Encryption/Decryption algorithm vs Existing algorithm

The current encryption and decryption algorithms were evaluated based on the chaos based key generation and it measures the encryption time. It could be used in a lot of techniques to enhance the security and effective encryption. But it produced higher time consumption when compared to the proposed encryption and decryption algorithm. Consider the Lena image with the size of 512\*512 with the recent encryption algorithm for implementation and testing.

The existing algorithms such as PRRABPM - Plaintext-related random access bit-permutation mechanism, LSCM-IEA - 2D-LSCM( Logistic-Sine-coupling map)-based image encryption algorithm, SDC Encryption- SHA DNA Chaotic Encryption are compared with the proposed algorithms[14][19][29][30].The detailed time analysis of existing and proposed techniques is shown in Table 11.

TABLE 11 Time Analysis of Proposed Algorithm Vs Existing Algorithm

Algorithm/Time Measurement	Average Encryption Time (ms)	Average Decryption Time (ms)	Total Time (ms)
Proposed	3071	3065	6136
PRRABPM Technique	4600	4300	8900

(Cai <i>et al.</i> , 2018)			
Image Encryption based on Permutation-Diffusion[29] Architecture (Cheng <i>et al.</i> , 2019)	7200	6940	14100
LSCM-IEA Technique	94100	101400	195500
SDC Encryption (Samiullah <i>et al.</i> , 2020)	22400	23120	45520

The proposed encryption time is 3371 milliseconds and decryption time is 3475 milliseconds. It produced a lower time consumption when compared to the proposed algorithm. Hence, the proposed encryption and decryption algorithm proved as the high efficiency and lower time consumption.

### 3.9. COMPUTATIONAL COMPLEXITY ANALYSIS

THE ANALYSIS OF COMPUTATIONAL COMPLEXITY MAINLY FOCUSES THE AMOUNT OF RESOURCES USED FOR THE ENCRYPTION AND DECRYPTION ALGORITHM. THERE ARE TWO BASIC REQUIREMENTS FOR COMPUTATIONAL COMPLEXITY SUCH AS TIME AND SPACE COMPLEXITY [31]. TABLE 11 SHOWS THE REDUCTION OF EXECUTION TIME WHILE COMPARED WITH THE EXISTING ALGORITHMS. AVERAGE TIME CALCULATION IS IMPLEMENTED WITH VARIOUS SYSTEMS AND DIFFERENT OPERATING SYSTEMS SPACE COMPLEXITY IS THE MEMORY SPACE REQUIRED TO COMPLETE THE PROPOSED ENCRYPTION ALGORITHM. THE MINIMUM MEMORY SPACE UTILIZATION WAS TO BE PROVEN BY VARIOUS FACTORS.

## 4. CONCLUSION

A fast colour image encryption scheme was introduced by the compression-then-encryption concept to improve speed and secure against attacks. The proposed scheme is the hybrid combination of Chaos techniques for key generation, Huffman Encoding for compression and avoiding colour decomposition, scrambling for more confusion, and DNA Encoding for reducing storage size. Security analyses demonstrated to prove the greater NPCR, UACI, robust against attacks and improved the speed performance of the proposed work. For future direction, the proposed work was extended with the various images to prove the high security, robustness and reliability.



## REFERENCES

- [1] S. Swathi, P. Lahari and Bindu, "Encryption Algorithms: A Survey," *International journal of Advanced Research in computer science & technology*, vol. 4, no. 2, 2016.
- [2] M. M. Ahamad and M. I. Abdullah, "Comparison of Encryption Algorithms for Multimedia," *Rajshahi University Journal of Science & Engineering*, vol. 44, pp. 131-139, 2016.
- [3] G. Singh and Supriya, "A Study of Encryption Algorithms (RSA, DES, 3DES and AES) for Information Security," *International Journal of Computer Applications*, vol. 67, no. 19, pp. 33-38, 2013.
- [4] P. M. Modak and D. V. Pawar, "A Comprehensive Survey on Image Scrambling Techniques," *International Journal of Science and Research (IJSR)*, vol. 4, 2015.
- [5] M. Bahrami and M. Singhal, "A Light Weight Permutation Based Method for Data Privacy in Mobile CloudComputing," in *3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud).*, 2015.
- [6] M. Bahrami and M. Singhal, " cloudPDB:A light-weight data privacy schema for cloud-based databases," *International Conference on Computing, Networking and ommunications, Cloud Computing and Big Data.*, 2016.
- [7] N. Holt, "Chaotic Cryptography:Application of Choas Theory to cryptography," 2017.
- [8] S. Shuliang, "A novel hyperchaotic image encryption scheme based on DNA encoding, pixel-level scrambling and bit-level scrambling," *IEEE Photonics Journal*, vol. 10, no. 2, pp. 1-14, April 2018.
- [9] M. Sankari and P. Ranjana, "Energy Efficient Symmetric image protection over cloud storage," *International Journal of Advanced Science and Technology*, vol. 29, no. 9s, pp. 4427-4432, 2020.
- [10] M. Sankari, P. Ranjana and D. Venkata Subramanian, "Iprivacy-Performance Measurement of Encrypted Image Over Mobile Cloud," *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 8, no. 4, pp. 2919-23, Nov 2019.
- [11] M. Sankari, Y. Kalaivani and R. P. , "Anomaly Detection in Distributed Denial of Service Attack using Map Reduce Improvised Counter Based Algorithm in Hadoop," *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 8, no. 4, pp. 4668-71, nov 2019.
- [12] A. Muhammad Baqer Mollah, A. Md. Abul Kalam Azad and A. Vasilakos, "Security and privacy challenges in mobile cloud computing: Survey and way ahead," *Journal of Network and Computer Applications*, p. 38–54, 2017.
- [13] L. Xu, Z. Li, J. Li and W. Hua, "A novel bit-level image encryption algorithm based on chaotic maps," *Opt. Lasers Eng.*, vol. 78, no. 21, pp. 17–25,, 2016.

- [14] M. Brindhaa and N. Ammasai, "A chaos based image encryption and lossless compression algorithm using hash table and Chinese Remainder Theorem," *Applied soft computing*, p. 379–390, 2016.
- [15] S. sun, "A Novel Hyperchaotic Image Encryption scheme based on DNA Encoding , pixel level scrambling and bit level scrambling," *IEEE photonics*, vol. 10, no. 2, April 2018.
- [16] X. Wu, H. Kan and J. Kurths, "A new color image encryption scheme based on DNA sequences and multiple improved 1D chaotic maps," *Applied Soft computing*, vol. 37, pp. 24-39, 2015.
- [17] S. Muhammad, W. Aslam, and N. Hira, "An Image Encryption Scheme Based on DNA Computing and Multiple Chaotic Systems," *IEEE Access*, vol. 8, pp. 25650-25663, 2020.
- [18] M. Mondal and K. S.Ray, "Review on DNA Cryptography," Cornell University, 2019.
- [19] B. Mondal and T. Mandal, "A light weight secure image encryption scheme based on chaos & DNA computing," *Journal of King Saud University – Computer and Information Sciences*, p. 499–504, 2017.
- [20] Alsarhan, Ayoub & Almalkawi, Islam & Halloush, Rami & Al-Karaki, JN & Al-Dubai, Ahmed., "A lightweight and efficient digital image encryption using hybrid chaotic systems for wireless network applications," *Journal of Information Security and Applications*, vol. 49, pp. 2214-2126, 2019.
- [21] R. D and . P. P, "Chaos based crossover and mutation for securing DICOM image," *Computers in biology and medicine*, pp. 170-184, 2016.
- [22] Vikas Kumar, "Compression Techniques vs Huffman Coding," *International Journal of Informatics and Communication Technology*, vol. 4, no. 1, pp. 29-37, 2015.
- [23] Yuan, L. T. tan, T. Hu and Hong, "A new parallel image cryptosystem based on 5D hyper-chaotic system," *image communication*, vol. 52, no. C, pp. 87-96, 2017.
- [24] Wu, Jiang et al, "A Novel Image Encryption Approach Based on a Hyperchaotic System, Pixel-Level Filtering with Variable Kernels, and DNA-Level Diffusion," *Entropy*, vol. 22, no. 1, pp. 1-19, 2020.
- [25] W. Xingyuan , L. Teng and Q. Xue , "A novel colour image encryption algorithm based on chaos," *signal Processing*, pp. 1101-08, 2011.
- [26] X. Chai, . Z. Gan, Y. Lu, . Y. Chen and D. , "A novel image encryption algorithm based on the chaotic system," *International Journal of Modern Physics C*, vol. 28, no. 4, p. 1750069 (24 pages), 2017.
- [27] M. Y. Valandar, M. J. Barani and Peyma, "A fast color image encryption technique based on three dimensional chaotic map," *Optik- International Journal for Light and Electron Optics*, vol. 183, pp. 1-17, 2019.
- [28] S. Sun, "Chaotic Image Encryption Scheme using two-by-two DNA complementary rules," *Optical Engineering*, vol. 56, no. 11, pp. 116117-1-9, 2017.

- [29] M. Kaur and S. D, “Multiobjective evolutionary optimization techniques based hyperchaotic map and their applications in image encryption.,” *Multidim Syst Sign Process*, vol. 32, p. 281–301, 2021.
- [30] Q. Liu , G. Wang and J. Wub, “Time-based proxy re-encryption scheme for secure data sharing in a cloud environment,” *Information Sciences*, vol. 258, p. 355–370, 2014.