JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Novel Image Encryption Algorithm Cross-utilizing 6D Hyperchaotic System and Chebyshev Chaotic Map

[1] Vegesna Girish Varma, [2] K. RamaDevi

[1]*Student, M.Tech (CE & SP), UCEK(A), JNTUK, Kakinada, Andhra Pradesh, India*
[2]*Assistant Professor, Deptartment of ECE, UCEK(A), JNTUK, Kakinada, Andhra Pradesh, India*

Abstract : In the era of information technology, users are transferring millions of images every day. If the information included in these images is susceptible to unlawful use, serious problems could result. There are various techniques for securing images. Digital image encryption is one of the most effective and well-known techniques. The two major steps of encryption algorithms are confusion and diffusion. In this paper, a new algorithm for image encryption that makes use of a hyperchaotic system and a chebyshev chaotic map is proposed. The original image is confused using random numbers generated by 6D hyperchaotic system. Then a key to decode the permuted image is generated via a chebyshev chaotic map. The effectiveness of the proposed method for image encryption is assessed using security analysis and time complexity. Entropy, correlation coefficient, differential attacks, histograms, keyspace, sensitivity, noise and data cut attacks are used to test the security. The bit-stream produced after serializing all intensity values of an encrypted image is also tested using the National Institute of Standards and Technology's Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. The results are also compared with different encryption techniques. With the proposed methodology, a high level of security is attained.

IndexTerms **- Image encryption, hyperchaotic system, chaos, chebyshev map, entropy, attacks, NIST statistical test.**

1. INTRODUCTION

Thousands of digital images are transferred every second during the routine process of digital image transmission over multiple networks. Medical images are sensitive in healthcare networks, where their improper use could result in incorrect diagnosis and poor medical judgement. High-security levels are needed when transmitting military photographs via various networks to avoid unauthorized access. Users of social networks do not want other people to have access to their images. Digital picture owners typically do not want unauthorized access to their images. These factors have made protecting the information in digital images a crucial concern. Image confidentiality is achieved through a variety of security measures, making it impossible for an unauthorized person to view the content of an image.

The three primary categories of image security techniques are data hiding [1], image watermarking [2, 3], and encryption [4, 5]. A secret message is embedded into the cover image using data hiding techniques so that it cannot be detectable. When image watermarking techniques are used, digital data is introduced into the image so that the watermarked and original versions are perceptible. In image encryption methods, the key used to convert the digital input image to a noisy image which cannot be predicted or comprehended. Without the key, users cannot retrieve the encrypted image.

Digital image encryption uses a variety of methods, including those based on the chaos theory, DNA, the quantum approach, and compressive sensing. Techniques for image encryption rely on two key steps. Confusion over which pixel arrangements are altered is the first stage. The second step, diffusion, depends on altering the pixel values. Inherent characteristics of chaotic-based approaches include non-periodicity, random behaviour, and sensitivity to initial conditions and control parameters. These characteristics make it possible to encrypt images successfully using chaotic-based techniques.

Chaotic-based digital image encryption systems can be divided into two groups according to Chai et al. [6]. Low-dimensional systems, such as one-dimensional chaotic maps, fall within the first category. High-dimensional systems, like hyperchaotic systems, are the second type. The simple structures of the low-dimensional chaotic maps make them approachable and useful. These maps have a small keyspace and low security levels despite these inherent qualities. Chen and Hu [7] proposed a medical image encryption method using a logistic-sine map for the confusing process. A coupled hyperchaotic system is used by Liu et al. [8] for pathological image encryption. A novel method for encrypting grey images is developed by Zheng and Liu [9]. First, a brand-new 2D chaotic map system (2D-LSMM), based on both logistic and sine maps is presented. Next, a DNA-based encryption system is used, in which 2D-LSMM chaotic sequences are used to derive the encoding and operation rules for DNA sequences.

There are some limitations on related works, which are as follows:

1. Low sensitivity to the initial conditions and low keyspace.
2. Some encryption techniques are unable to recover the plain image when the encrypted image is subjected to noise and data cuts.
3. Because the histogram of the encrypted image is not flat, some of encryption techniques are vulnerable to statistical attacks.
4. The chaotic map's condition is independent of the plain image, which causes limitations in its ability to resist against differential attacks.

To get beyond the limits of low-dimensional chaotic systems, hyperchaotic techniques are applied. In terms of randomness, unpredictability, nonlinearity, and initial conditions, the hyperchaotic approaches performed better than the low-dimension chaotic methods.

The following is a summary of this paper's contributions:

1. The 6D hyperchaotic system and a chaotic map are integrated to ensure a high level of security.
2. The proposed algorithm offers strong resistance to brute force attacks because to its large keyspace.
3. The proposed algorithm for image encryption is extremely resistant to most attacks.
4. Additionally, the National Institute of Standards and Technology (NIST) Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications is used to test the bit-stream created after serializing all intensity values of an encrypted image.

2. MATHEMATICAL FOUNDATIONS

2.1 Six-Dimensional Hyperchaotic System

In general, chaotic functions are nonlinear with dynamic behaviour, according to mathematical analysis. They are unpredictable in their responses as a result. According to earlier research, the dynamical behaviour of hyperchaotic functions is substantially more sophisticated compared to that of low-dimension chaotic functions. There should be at least four dimensions in a hyperchaotic system. Additionally, compared to hyperchaotic systems, low-dimension chaotic functions only have one positive Lyapunov exponent. The 6D hyperchaotic system is defined by Wang and Yu [10] as follows:

$$\begin{aligned} x_1 &= a\,(x_2 - x_1) + x_4 - x_5 - x_6 \\ x_2 &= cx_1 - x_2 - x_1x_3 \\ x_3 &= -bx_3 + x_1x_2 \\ x_4 &= dx_4 - x_2x_3 \\ x_5 &= ex_6 + x_3x_2 \\ x_6 &= rx_1 \end{aligned} \qquad (1)$$

where a, b, c, d, e, and r are constants; x_1, x_2, x_3, x_4, x_5, and x_6 are state variables of the 6D hyperchaotic system. The constant values chosen for this paper are: a = 10, b = 8, c = 28, d = 1, e = 8, and r = 3. By making this decision, the system is guaranteed to have two positive Lyapunov exponents that satisfy the requirement (sum of all exponents is negative).

2.2 Chebyshev Chaotic Map

The Chebyshev chaotic map is a one dimensional chaotic system with one initial condition y_0 and one control parameter k and can be described as follows [11]:

$$y_{n+1} = \cos\left(k \cos^{-1}(y_n)\right) \qquad (2)$$

where $y_n \in [-1, 1]$ for n = 0, 1, 2, … and $k \in [2, \infty]$.The bifurcation diagram of the Chebyshev map defines that all the (y_0, k) where $y_0 \in [-1, 1]$ and $2 \leq k < \infty$ can be used as secret keys. The Chebyshev map has a positive increasing Lyapunov exponent at k >= 2, and thus, it is always chaotic. In this paper, the parameter k value for generating map is chosen as 2000.

3. THE PROPOSED ALGORITHM

The novel technique used a six-dimensional hyperchaotic system and Chebyshev Chaotic map to encrypt the input image. Due to its complex, high-dynamic behaviours and two positive Lyapunov exponents, the utilization of the 6D hyperchaotic system enhances encryption performance and improves security levels. Chebyshev Chaotic map is swift and able to diffuse the permuted image.

3.1 Encryption

Confusion and diffusion are the two phases that make up the encryption. In each of these procedures, the arrangements and values of the pixels are changed respectively. The confusion step is built on the 6D hyperchaotic system. The system's initial condition, which is based on the plain image, is first calculated. The hyper chaotic system is then iterated to produce a new vector, after which we choose three sequences (x_1, x_3, and x_5). The order of the sorted numbers in this vector is employed to confuse the plain image. The diffusion process is carried out to obtain the encrypted image after confusing the plain image. The diffusion in our approach is based on the key generated by Chebyshev Chaotic map. The key generation steps are defined as follows:

1.Use the following equation to determine the chaotic chebyshev map's initial value, which depends on the plain image P:

$$Y(1) = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} P(i,j)}{MxNx255} \quad (3)$$

2.To create a new sequence S with size MN, iterate the chaotic map (eq. 2) N_0 + MN times, skipping the first N_0 entries.
3.Calculate the key using the following formula:

$$K(i) = mod(floor(S(i) \times 10^{14}), 256) \quad (4)$$

The diffusion process modifies the image's pixel values, which results in the creation of a noisy image. By operating bit-wise exclusively OR operation of the confused image vector with the key K, the encrypted image is produced. In Algorithm section, thorough encryption procedures are described.

3.2 Decryption

In contrast to encryption, decryption involves the exact opposite steps. The steps listed below can be used to extract the plain image from the encrypted image:

1. To obtain the scrambled image, use a bit-wise exclusive OR operation to the encrypted image vector and key K.
2. A vector W is created from the scrambled image (D′) that is obtained in the previous step.
3. The following equation is used to restore each pixel to its initial place using the vector S created during the encrypting step:

$$ER(S_i) = W_i \, , \, i = 1 : MN \quad (5)$$

4. To get the decrypted image, convert the vector ER to a matrix (D).

4. ALGORITHM

The proposed image encryption algorithm is as follows:

1: i = 1
2: Create a P vector by converting the input image matrix.
3: Calculate the hyperchaotic system's initial key as follows:

$$x_1 = \frac{\sum_{i=1}^{MN} P(i) + (MxN)}{2^{23} + (MxN)} \quad (6)$$

$x_i = \text{mod}\,(x_{i-1} \times 10^6, 1)$, i = 2, 3, …, 6
with the initial conditions; $x_1, x_2, \ldots, x_6$.

4: You can create a new sequence L with dimension M x N by iterating the hyperchaotic system in (eq. 1) N_0 + MN/3 times and then discarding the N_0 values. (We choose three sequences from the system in (eq. 1): x_1, x_3, and x_5).
5: Return the positions of L in vector S after sorting L in ascending order.
6: Permit the picture vector P to produce the following newly shuffled sequence

$$R_i = P(S_i), \, i = 1 : MN \quad (7)$$

7: Create the chebyshev map's initial state by utilizing (eq. 3)
8: In order to obtain a new sequence S with size MN, iterate the chebyshev chaotic map (eq. 3) N0 + MN times.
9: Get the sequence K by iterating equation (eq. 4) MN times.
10: Transform the matrix X into the pixel vector of a 1D image, X'.
11: $Enc = R_i' \oplus K$
12: Convert *Enc* into a 2D matrix C.

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

Fig-1: The test grayscale images; (a) Img1 (512x512) [12], (b) Lena (512x512) [13], (c) Boat (512x512) [14], (d) Baboon (512x512) [14], (e) Peppers (512x512) [14]

5. TESTS AND SIMULATION RESULTS

The effectiveness of the proposed method is evaluated using various grayscale images, as shown in Figure 1. The proposed algorithm is also evaluated against other image encryption algorithms. All experiments are carried out on a laptop with an 8 GB RAM and Core i5-1135G7 2.4GH CPU running MATLAB (R2020a).

In eight tests, the proposed encryption method is evaluated using entropy, noise and data cut attacks, correlation coefficients, key sensitivity, differential attack, histograms, keyspace, and the NIST Statistical Test. The parameter used in our algorithm is the iteration number $N_0 = 1000$.

5.1 Entropy

Information entropy calculates the image's randomness. Entropy is described mathematically as follows:

$$H(m) = \sum_{i=1}^{w} P(m_i) log_2 \frac{1}{P(m_i)} \quad (8)$$

where P(m) is the probability of appearance of m. For grayscale images, the maximum value of entropy is 8. The randomness of the image's pixels is higher when the entropy number is close to 8. As part of this experiment, we encrypt the grayscale test images using the proposed algorithm and calculate the entropy values of the encrypted images, which are shown in Table 1. We can see from the observations that every entropy number is close to 8, which indicates that the encrypted images are truly random. Using our algorithm and the other encryption algorithms described in Table 2, the second test image (i.e., Lena) is encrypted. As can be seen, when compared to the various methods in Table 2, our presented method has a greater entropy value. We draw the conclusion from this test that our proposed technique ensures producing encrypted images with great randomness.

Table-1: Encrypted Images entropy

Test image	Entropy
Img1	7.9993
Lena	7.9994
Boat	7.9994
Baboon	7.9993
Peppers	7.9994

Table-2: Entropy value of our algorithm and other algorithms

Method	Entropy
Proposed	7.9994
[15]	7.9973
[16]	7.9993
[17]	7.9971
[18]	7.9971

5.2 Correlation Coefficient

In the input images, the neighbouring pixels frequently show a strong association in the diagonal, horizontal, and vertical axes. This correlation must be reduced for an encryption scheme to be effective. The following formula calculates the correlation coefficient between any two neighbouring pixels, A and B:

$$r_{A,B} = \frac{E\big(\big(A - E(A)\big)\big(B - E(B)\big)\big)}{\sqrt{D(A)D(B)}}$$

$$E(A) = \frac{1}{s} \sum_{i=1}^{s} A_i \quad (9)$$

$$D(A) = \frac{1}{s}\sum_{i=1}^{s}(A_i - E(A))^2$$

where the integer s referring to the total number of adjacent pixels; D(A) and E(A) stand for the variance and expectation of A, respectively.

In the horizontal (H), vertical (V), and diagonal (D) directions of the grey test images and their encrypted versions, Table 3 lists the correlation coefficient values for each. The correlation coefficient values of the test images are all close to one, whereas the correlation coefficient values of the encrypted images are close to zero. Table 4 provides a comparison of Lena image with other methods.

Table-3: Correlation coefficient values

Test Image	Direction	Plain image	Encrypted image
Img1	V	0.9848	0.0016
	H	0.9723	0.0008
	D	0.9649	-0.0037
Boat	V	0.9713	0.0001
	H	0.9381	0.0023
	D	0.9222	-0.0021
Baboon	V	0.7587	-0.0031
	H	0.8665	-0.0036
	D	0.7262	0.0028
Peppers	V	0.9792	0.0013
	H	0.9768	-0.0006
	D	0.9639	-0,0013

Table-4: Comparison of the correlation coefficient values between our algorithm and other algorithms

Method	H	V	D
Proposed	0.0049	0.0004	0.0001
[15]	−0.0053	−0.0012	0.0050
[16]	0.0019	0.0069	0.0200
[17]	-0.0056	0.0006	0.0018
[18]	-0.0059	-0.0064	-0.0003

5.3 Noise and Data Cut Attacks

When images are exchanged over the network, noise or cropping (data cut) attacks are possible. Algorithms for image encryption should be resistant to noise and cropping attacks. The decrypted image quality is assessed using the widely used metric PSNR (peak signal to noise ratio). The PSNR for the original and decrypted images, I_O, and I_D according to mathematics is:

$$PSNR = 10\ x\ log_{10}\left(\frac{255^2}{MSE}\right)\ (db) \qquad (10)$$

where MSE stands for mean square error:

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}|OI(i,j) - EI(i,j)|^2 \qquad (11)$$

High image quality is indicated by a higher PSNR value. Original and decrypted images are indistinguishable for a PSNR > 35. The purpose of this experiment is to evaluate robustness against noise and data cut attacks. In this experiment, the new technique is used to decrypt an encrypted image that has been contaminated with "salt and peppers" noise at two distinct levels, 0.002 and 0.005. Additionally, a data cut of 64 x 64 and 128 x 128 is used to attack the encrypted images before the new approach is used to decrypt them. Table 5 displays the PSNR for the five test images with noise and data cuts.

The PSNR is reduced to 20dB when the encrypted image is attacked with a data cut off size of 128 x 128, which is a reasonably large cut off (i.e., the encrypted image lost 1/8 information). The decrypted image is recognizable despite the lower PSNR values. Figure 2 illustrates the noise and data cut attacks for an encrypted image, showing how the reader can quickly identify the contents of the decrypted images in various scenarios. The new method is therefore robust and resistant to various attacks.

Table-5: Peak signal to noise ratio (PSNR) (dB) values for noise and data cut attacks

Test Images	Data cut with block size:		Salt and Pepper with noise level:	
	64 x 64	128 x 128	0.002	0.005
Img1	23.3337	17.2738	32.9294	28.5912
Lena	27.0084	21.2101	36.2736	31.9938
Boat	27.1581	21.3478	36.3196	32.3013
Baboon	27.9174	21.6533	36.4043	32.8557
Peppers	26.7752	20.7368	35.4221	31.7593

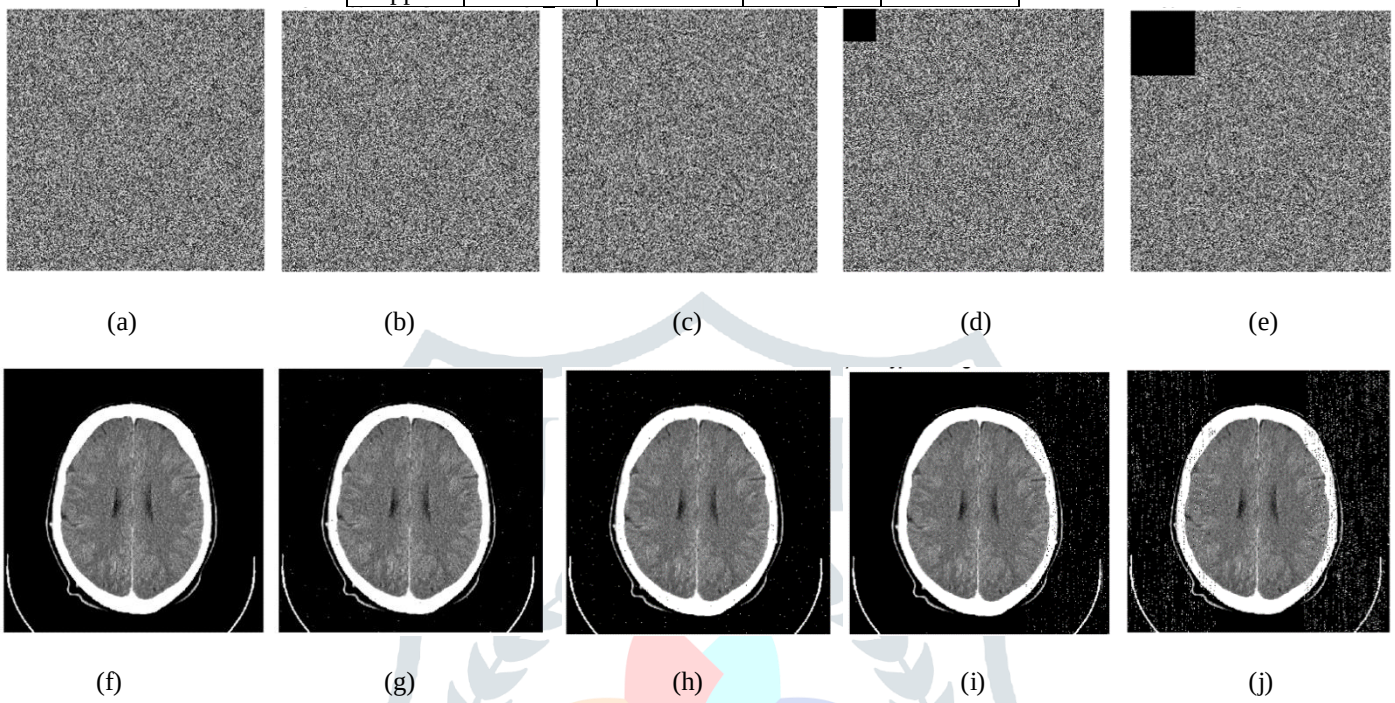


Fig-2: Noise and data cut attacks; (a) The encrypted image of Img2, (b) noisy encrypted image with 0.002, (c) noisy encrypted image with 0.005 and (d) encrypted with 128x128 data cut, (e) encrypted image with 64x64 data cut. (f-j) Images of the decrypted (a-e)

5.4 Differential Attack

By figuring out the relationship between the original and encrypted images, the attacker hopes to decrypt the encrypted images without needing the key in this attack. Small modifications in the original image's pixels have a big impact on the encrypted version, making it more challenging for hackers to decrypt the encrypted version. This attack must be prevented by strong image encryption methods. Robustness to this attack is evaluated based on the Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) values obtained from following equations:

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

$$D(i,j) = \begin{cases} 0 & if\ E_1(i,j) = E_2(i,j) \\ 1 & if\ E_1(i,j) \neq E_2(i,j) \end{cases} \qquad (12)$$

$$UACI = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{|E_1(i,j) - E_2(i,j)|}{255}\ x\ 100\ (\%)$$

symbols E_1 and E_2 refer to two encrypted images i.e., plain image and the modified image (made by changing one pixel in the plain image). The image has a width of M pixels and a height of N.

Here, we compare the NPCR and UACI values between the two encrypted images in Table 6 to examine the effectiveness of our proposed methodology in defending against differential attacks. NPCR should be 99.6094%, and UACI should be 33.4635% with respect to their ideal values. Every value in Table 6 is near to their ideal values. A comparison of our approach and other image encryption algorithms can be seen in Table 7 for the original Lena image. The outcomes demonstrate how well our proposed method can withstand differential attacks.

Table-6: NPCR and UACI performances

Test image	NPCR (%)	UACI (%)
Img1	99.6193	33.4091
Boat	99.5663	33.3844
Baboon	99.6250	33.4544
Peppers	99.6124	33.3804

Table-7: Comparison of NPCR and UACI

Method	NPCR	UACI
Proposed	99.6067	33.4449
[15]	99.6216	33.6642
[16]	99.6174	33.4322
[17]	99.6216	33.5848
[18]	99.6197	33.0443

5.5 Histograms

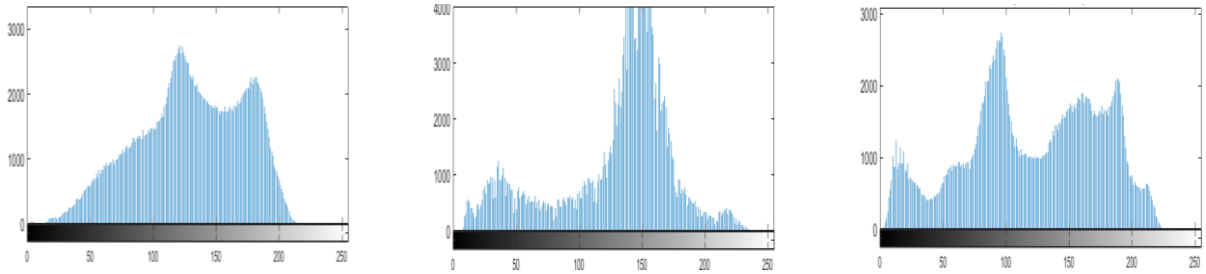
The distribution of pixels in the image is shown by the histogram. The histogram for an encrypted image should be flat to make it impossible for attackers to predict any image data. Additionally, the histograms of the plain image and the encrypted image shouldn't be same. Using the new approach, three standard grayscale images Baboon, Boat, and Peppers were encrypted. The histograms of the encrypted images created using our method are uniform and distinct from those of the equivalent plain image histograms, as can be shown.

To ensure that the histogram of the encrypted image is uniform, a further experiment is conducted. The chi-square test (X^2) used in this experiment is calculated by [19]:

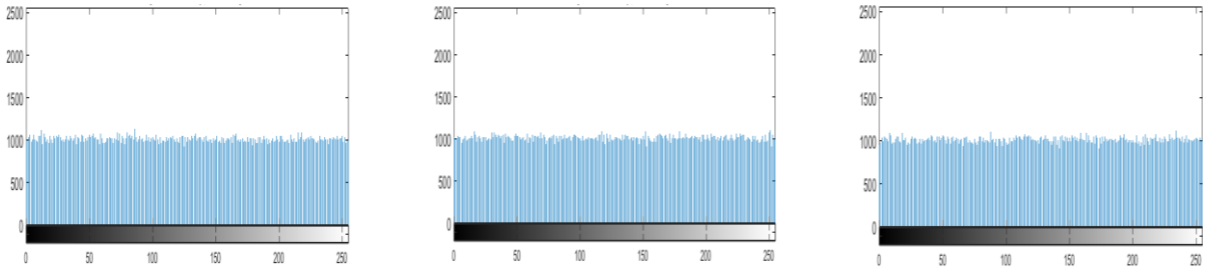
$$\chi^2 = \sum_{i=1}^{256} \frac{(O_i - EV)^2}{EV} \qquad (13)$$

where EV = O/256 is the expected frequency of each grey value and O_i is the rate of occurrence of grey value i. The value of $X^2_{(a,d)}$ is 293.2478, where 0.05 is for significance and d is 255 representing degrees of freedom. Table 8 displays the X^2 values of the encrypted image. As all of the values are below 293, the histogram of the images encrypted using our proposed algorithm is uniform. These outcomes validate the effectiveness of the new algorithm.

(a) (b) (c)



(d) (e) (f)



(g) (h) (i)

Fig-3: Histogram analysis; (a) Baboon, (b) Boat, (c) Peppers, (d) Histogram of (a), (e) Histogram of (b), (f) Histogram of (c), (g) Histogram of encrypted image in (a), (h) Histogram of encrypted image in (b), (i) Histogram of encrypted image in (c)

Table-8: Chi-Square analysis

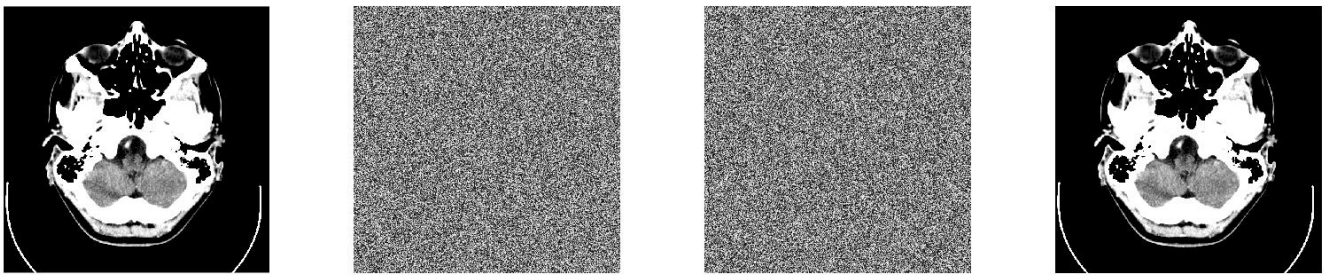
Test image	Encrypted Image
Img1	271.1836
Lena	213.7773
Boat	213.2031
Baboon	260.7637
Peppers	214.5684

5.6 Keyspace

The size of the keyspace is important to the encryption process. If the keyspace size is greater than 2^{100}, the encryption technique is resistant to brute force attacks. Different security keys are included in the proposed encryption algorithm: x_1, x_2, x_3, x_4, x_5, x_6, N_0, a, b, c, d, e, and r. If we assume that the precision of the initial value is equal to 10^{16}, then the total keyspace is greater than N_0 x 10^{96}, indicating robustness to a brute force attack.

5.7 Key Sensitivity

Any little change to the secret key should make a practical algorithm susceptible. Any slight variation in the key used in the decryption step prevents the reconstruction of the plain image because attackers can defeat the encryption process using a key that is identical to it. Here, we alter Y_0 to $Y_0 + 10^{10}$ and generate two secret keys with a minimal change. Figure 4(a) displays the input plain image for which initial key is used to encrypt, and results are displayed in Figure 4(b). The second key is applied to the plain image in the decryption process (Figure 4(c) displays the results). The plain image is not successfully retrieved by the second key, as can be seen. Figure 4(d) illustrates the successful reconstruction of the plain image when the first key is used in the decryption stage.



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

Fig-4: The sensitivity of our algorithm to the key; (a) Original Img1, (b) encrypted image of (a) with the original initial conditions, (c) decrypted image in (a) with the modified key, (d) decrypted image in (a) with the original key

5.8 NIST Statistical Test

An encrypted image produced by a good encryption method ought to be highly random. The NIST statistical test suite includes statistical tests to make sure the encryption algorithm generates a random sequence. For all tests in NIST, the significance level is set to 0.01. In this experiment, a 512 by 512 encrypted peppers image that has been transformed to a binary sequence has p-values computed for it. The results of various statistical tests are then recorded in Table 8. The p-values are greater than 0.01 indicating the randomness of the encrypted image's binary sequence. The outcomes demonstrate that the suggested algorithm's generated sequence passed every test, guaranteeing the unpredictability of the binary sequence.

Table-8: NIST statistical test

Test Name	*p*-Value	Conclusion
Frequency	0.1768	Random
Block-frequency	0.0865	Random
Runs	0.1150	Random
Longest run	0.7077	Random
Discrete Fourier Transform Test	0.7936	Random
Non-overlapping template	0.6308	Random
Cumulative sums (forward)	0.0993	Random
Cumulative sums (reverse)	0.0554	Random

5.9 Computational Complexity

The steps necessary to complete the encryption process are used to evaluate the computational complexity of the algorithm. For a plain image of size M x N, the proposed algorithm's confusion steps have an *O(MxN)* time complexity. The time complexity for the key generation and diffusion stages is *O(MxN)*. As a result, the overall time complexity of our proposed algorithm is *O(MxN)*.

6. Conclusion

This paper introduced a new grey image encryption algorithm. The Chebyshev Chaotic map is combined with a 6D hyperchaotic system in this algorithm. First, we choose three sequences from a 6D hyperchaotic system before changing the pixel positions. The Chebyshev chaotic map is then used to change the pixel values of the shuffled image.

The new technique is sensitive to small variations in the secret key and pixel distribution, producing a completely different encrypted image. As a result, the proposed algorithm successfully defends against the differential attack. When the keyspace size is large enough, the new algorithm can withstand a brute force attack. Additionally, information entropy, correlation coefficients, noise and data cut attack, and histogram are used to evaluate the security performance of the novel algorithm. The results demonstrate that the proposed algorithm performs well when encrypting grayscale images when compared to other recent encryption algorithms. In the future, we will research how well our algorithm works for encrypting colour images and focus on making the algorithm's proposed running speed faster.

References

[1] Abdel-Aziz, M.M., Hosny, K.M. & Lashin, N.A. Improved data hiding method for securing color images. *Multimed Tools Appl* 80, 12641–12670 (2021), doi: 10.1007/s11042-020-10217-9.

[2] K. M. Hosny, M. M. Darwish, K. Li and A. Salah, Parallel Multi-Core CPU and GPU for Fast and Robust Medical Image Watermarking, in *IEEE Access*, vol. 6, pp. 77212-77225, 2018, doi: 10.1109/ACCESS.2018.2879919.

[3] Hosny, K.M., Darwish, M.M. Robust color image watermarking using invariant quaternion Legendre-Fourier moments. *Multimed Tools Appl* 77, 24727–24750 (2018), doi: 10.1007/s11042-018-5670-9.

[4] Dolendro Singh Laiphrakpam, Manglem Singh Khumanthem, Medical image encryption based on improved ElGamal encryption technique, *Optik*, Volume 147, 2017, Pages 88-102, ISSN 0030-4026, doi: 10.1016/j.ijleo.2017.08.028.

[5] Li, Y, Yu, H, Song, B, Chen, J. Image encryption based on a single-round dictionary and chaotic sequences in cloud computing. *Concurrency Computat Pract Exper*. 2021; 33:e5182, doi: 10.1002/cpe.5182.

[6] Chai, X.; Fu, X.; Gan, Z.; Lu, Y.; Chen, Y. A color image cryptosystem based on dynamic DNA encryption and chaos. *Signal Process,* 2019, *155*, 44–62.

[7] Chen, X.; Hu, C.-J. Adaptive medical image encryption algorithm based on multiple chaotic mapping. *Saudi J. Biol. Sci.* 2017, *24*, 1821–1827.

[8] Liu, H.; Kadir, A.; Liu, J. Color pathological image encryption algorithm using arithmetic over Galois field and coupled hyper chaotic system. *Opt. Lasers Eng.* 2019, *122*, 123–133.

[9] Zheng, J.; Liu, L. Novel image encryption by combining dynamic DNA sequence encryption and the improved 2D logistic sine map. *IET Image Process.* 2020, *14*, 2310–2320.

[10] Wang, J.; Yu, W.; Wang, J.; Zhao, Y.; Zhang, J.; Jiang, D. A new six-dimensional hyperchaotic system and its secure communication circuit implementation. *Int. J. Circuit Theory Appl.* 2019, *47*, 702–717.

[11] R Noha, HA HossamEldin, EE Said, and EA Fathi, (2015), Hybrid ciphering system of image based on fractional Fourier transform and two chaotic maps , *International Journal of Computer Applications* (0975 – 8887), 119 (11) 12–17.

[12] *Category: Computed Tomography Images of Mikael Häggström's Brain*, Sept. 2022, [online] Available: https://commons.wikimedia.org/wiki/Category:Computed_tomography_images_of_Mikael_H%C3%A4ggstr%C3%B6m%27s_brain.

[13] File:Lenna (test image).png, Sept. 2022, [online] Available: https://en.wikipedia.org/wiki/File:Lenna_(test_image).png.

[14] *SIPI Image Database*, Sept. 2022, [online] Available: https://sipi.usc.edu/database/database.php?volume=misc.

[15] Tian, P.; Su, R. A Novel Virtual Optical Image Encryption Scheme Created by Combining Chaotic S-Box with Double Random Phase Encoding. *Sensors* 2022, 22, 5325. doi: 10.3390/s22145325.

[16] S. T. Kamal, K. M. Hosny, T. M. Elgindy, M. M. Darwish and M. M. Fouda, "A New Image Encryption Algorithm for Grey and Color Medical Images," in IEEE Access, vol. 9, pp. 37855-37865, 2021, doi: 10.1109/ACCESS. 2021. 3063237.

[17] Q. Lu, C. Zhu and X. Deng, "An Efficient Image Encryption Scheme Based on the LSS Chaotic Map and Single S-Box," in IEEE Access, vol. 8, pp. 25664-25678, 2020, doi: 10.1109/ACCESS.2020.2970806.

[18] W. J. Jun and T. S. Fun, "A New Image Encryption Algorithm Based on Single S-Box and Dynamic Encryption Step," in IEEE Access, vol. 9, pp. 120596-120612, 2021, doi: 10.1109/ACCESS.2021.3108789.

[19] A. Banu S and R. Amirtharajan, ''A robust medical image encryption in dual domain: Chaos-DNA-IWT combined approach,'' *Med. Biol. Eng. Comput.*, vol. 58, no. 7, pp. 1445–1458, Jul. 2020.