International Scientific Journal of Engineering and Management ISSN: 2583-6129
Volume: 03 Issue: 05 | May – 2024 DOI: 10.55041/ISJEM01815
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

A New Approach to Secure Image Information using Modified DNA Sequence and Chaos-Based Image Cryptosystem Architecture

FSanjit Mazumder[1], Sadananda Banerjee[2]

[1]Computer Science & Engineering, Seacom Engineering College, Howrah
[2]S^{2}Keross R & D, Junior Software Engineer, Kolkata, India,

---***---

Abstract - Image encryption stands as a vital safeguard for securing sensitive visual information in the modern digital landscape. In the current digital era, image encryption is a crucial precaution for protecting sensitive visual data. In order to increase security and effectiveness, this research presents a novel image encryption method that combines chaotic maps and DNA encoding. The image is first converted into a binary format and then encoded using DNA principles in the suggested method, adding complexity and security. A chaotic key-stream is produced by using chaotic maps, which are renowned for their sensitivity to initial conditions and capacity to produce pseudo-random sequences. The visual data is then effectively scrambled by applying this key-stream to permutation and diffusion operations at the pixel level. The combination of chaotic maps and DNA encoding enables strong defense against multiple attackers while retaining computing efficiency.

.

***Key Words*: :** Encryption, Decryption, Image encryption, DNA cryptography, Chaos-Based Cryptography, 2-D logistic map

1. INTRODUCTION

This The rapid advancement and widespread adoption of digital communication and network technologies have revolutionized data storage and exchange over the Internet. However, as information flows more freely, safeguarding the confidentiality and integrity of data has become paramount. Network security and data integrity have emerged as critical concerns in this digital age. To address these concerns, scientists have taken proactive steps to enhance security and prevent vulnerabilities.

In the context of this digital landscape, much of the multimedia content shared and stored online consists of images. As a result, ensuring the confidentiality and authenticity of digital images has become a primary focus. To achieve this, researchers and experts have developed image encryption techniques, which play a crucial role in protecting sensitive visual information during transmission and storage. Image encryption methods are designed to secure digital images, ensuring that only authorized individuals can access and decipher them.

In a world where visual data is being exchanged and transferred electronically, picture encryption is essential for maintaining the secrecy, integrity, and authenticity of digital images. Protecting sensitive data and reducing potential risks related to illegal access or data breaches are core security measures. Many industries, including commerce, telemedicine, medicine, biometric authentication, and military communication, use image encryption.

To address these security concerns, many picture encryption techniques have been developed, such as digital watermarking [1], image scrambling [2], image steganography [3], and image cryptography [4]. By altering pixel placements as well as their contents, picture encryption [5][6] can be accomplished, entirely modifying the image's original information. The image can be encrypted using a variety of methods, including cryptography [7], steganography, compression, and digital watermarking. Due to its essential trait of sensitivity to initial conditions, which results in data sets that are deterministic yet appear random, chaos has recently attracted a lot of attention in the field of cryptography. With regard to speed, cost, processing power, computational overhead, complexity, vulnerability, and other factors, chaos-based cryptographic models have been applied to create unique techniques for designing effective image encryption systems.

The paper is organized as follows: Section 1 is the introduction of image encryption and Steganography, Section 1.1 contain the brief introduction of chaos based cryptography, Section 1.2 contain chaotic System, Section 1.3 is Chaos's architecture, Section 1.4 is brief description of DNA cryptography, Section 2 is the methodology, Section 2.1 is the algorithm for encryption, Decryption algorithm is given in Section 2.2, Section 3 is result analysis and Section 4 is conclusion and Section 5 is reference.

1.1 Brief Introduction To Chaos-Based Cryptography

In order to provide safe image encryption and communication in the presence of an attacker, researchers have been carefully examining the intimate correspondence between chaotic systems and cryptography in recent years [8]. As a result, chaotic cryptography is described as a balanced fusion of the science of encryption and chaos theory. The primary distinction between the two is that while cryptosystems are mapped on an infinite set of integers, chaotic systems are defined on real numbers. Traditional ciphers like AES and DES work well for text encryption but are ineffective for image encryption because of the repeating data that pixels in related images display. This problem is fixed by chaos-based encryption methods, which generate uniformly dispersed random keys that encrypt the image data in the cipher images [9].

Although closely related, these two scientific ideas offer a good combination of enhanced performance, high levels of security, and a number of practical applications, including secure communications [10], image and video encryption [11], and the generation of pseudo-random numbers for stream and block ciphers [12]. Time can either be discrete or continuous, and dynamical equations that change with time are thought of as chaotic systems [13]. As these characteristics are comparable to the confusion and diffusion aspects of an effective cryptosystem, the distinctive characteristics of chaotic systems, such as determinacy, ergodicity, and sensitivity to initial conditions, make it a suitable choice for constructing cryptosystems. Consideringthe ongoing advancements in cryptanalysis and image encryption.

When creating a chaos-based algorithm, choosing the chaotic maps is the first and possibly most difficult stage [14]. When creating an effective cryptographic algorithm, one of the most crucial requirements is that the cryptosystem be quick and reliable. Researchers started the design process by employing simpler chaotic maps, like the tent map and the logistic map, which have a limited key space and provide insufficient protection. Later, as encryption algorithms were developed further, chaotic maps with bigger dimensions were used, which made the cryptosystem faster, more secure, and of higher quality. The Henon map [15], Tinkerbell map [16], Logistic map 1D [17], Logistic map 2D [18], Tent map [19], and a 5D Hyper-chaotic map [20] are only a few chaotic maps that have been researched for picture encryption.

1.2 Chaotic Systems

To better assist readers in understanding the next material, we first introduce chaotic systems, their traits, and certain indicators in this section.

1.2.1 Overview of Chaotic Systems

According to how they change over time, chaotic systems can be divided into two categories: continuous chaotic systems and discrete chaotic maps [21]. The dynamical equations of a continuous chaotic system are often a set of differential equations, and a continuous chaotic system is one in which the state of the system evolves continuously across time. The dynamical equation of a discrete chaotic map, on the other hand, is typically an iterative equation and describes a system whose state changes discretely over time. Discrete chaotic map: An evolving dynamic system with discrete state variables is a discrete chaotic map. Nonlinearity, sensitive dependency on beginning circumstances and parameter values, period multiplication, and other characteristics are some of the characteristics that define the discrete chaotic map [22]. Below are some examples of discrete chaotic maps along with their mathematical explanations to better understand:

Logistic map:

$$X_{t+1} = f_l\,(X_n) = \mu X_t\,(1 - X_t),$$

where $X_t \in (0, 1)$ is the state variable at time step t, μ ∈ [0, 4] is a control parameter, and $(1 - X_t)$ is the factor that limits the growth of the system. When the parameter is in the range (3.57, 4), the logistic map displays complex dynamical behaviour that is entirely nonperiodic, nonconvergent, and sensitive to beginning value. The chaos-based image cryptosystem architecture consists of two phases: confusion and diffusion. The confusion phase involves rearranging pixel positions while keeping values unchanged, converting the image into an unidentifiable form. The diffusion phase, executed using a chaotic map, changes pixel values consecutively, with multiple iterations until satisfactory security is achieved. The architecture is shown using block diagram in figure-1.

1.3 Chaos's Characteristics

Chaos has many crucial characteristics, which makes it perfect for creating a safe cryptosystem. It has been linked to logical starting points, nonlinearity, determinism, ergodicity, nonperiodicity, and unexpected behaviour [23, 24, 25]. Chaotic maps like Arnold's cat, logistic, sine, Henon, and tent are commonly used for image encryption confusion and diffusion. They can be adapted to provide larger chaotic ranges by adding nonlinear terms, increasing dimensions, or controlling settings.

TABLE 1

Provides a comprehensive list of chaotic maps and their key characteristics.

Chaos Maps	Mathematical form	Range of chaos
Logistic map [26]	$x_{n+1} = rx_n\,(1 - xn\,)$	r ∈ [3.54, 4]
Sine map [27, 28, 29]	$f_r\,(x\,)= rsin(\pi x)$	r ∈ [0.87,1]
Generalized logistic map [30, 31]	$f(\,x\,)= 4\mu\,/(1 + 4\,\mu\,2 - 1\,x(1 - x))$	$\mu\in[-0.6795, 0.4324]$
Squared sine logistic map [32]	$f(x) = rx(1 - x) + \mu sin\,2\,(2x)$	r ∈ [3.5, 6.5]
Enhanced - sine map [29]	$x_{n+1} = sin(\pi\mu sin(\pi x_n\,))$	$\mu\in (0, +\infty)$
Enhanced - logistic map [29]	$x_{n+1} = sin(\pi a x_n\,(1 - x_n\,))$	$a\in (0, +\infty)$
Enhanced tent map [29]	$x_{n+1} = \;sin\pi r x_i x_i< 0.5$ $sin(\pi r\,(1 - x_i\,))\;x_i \geq 0.5$	$r\in (0, +\infty)$
Enhanced Henon map [29]	$x_{i+1} = sin\pi(1 - ax_i^2 + y_i)$ $y_{i+1}sin(\pi bx_i\,)$	a = 912 b = 39
Modified logistic chaotic map [33]	$x_{n+1} = 2\beta - xi^2\,/\beta$	cryptanalyzed an infinite key space.
Tent map [34]	$X_{n+1} = rx_n;\;\; x_n< ½$ $= rx_n(1\text{-}x_n);\;\; x_n \geq 1/2$	r ∈ [0,2]

(1D)- sine-powered chaotic map (1DSP) [35]	$x_{n+1} = (x_n\,(\alpha + 1))$ $sin(\beta\pi + xn)$	$\alpha \in [2.5,4]$

1.4 DNA Cryptography

Adenine (A), guanine (G), cytosine (C), and thymine (T) are the four compound bases that make up the code that protects the information in DNA. More than 99 percent of the 3 billion bases that make up human DNA are the same in every person. Similar to how letters of the alphabet together appear in a specific order to form words and phrases, the invitation, or gathering, of those bases determines the data available for creating and maintaining a living creature. The information security plan might not work as planned. The nitrogen bases An (Adenine), T (Thymine), and C (Cytosine) are shown to mix with G (Guanine) and match with each other. Another area of cryptography that has grown with the study of DNA handling is DNA cryptography [37], where DNA is employed as an information carrier and advanced natural development is used as a usage device. DNA particles naturally exhibit enormous parallelism and extraordinary information thickness, which are investigated for a wide range of cryptographic applications, including encryption, confirmation, mark, etc.

A multi-level picture encryption calculation uses a combination of bedlam and DNA coding to change the area of pixels and pixel values. The process starts by converting the computerized picture into a DNA coding lattice, then dividing it into four sub-lattices, rearranging them, and finally translating the DNA framework into a picture matrix.[38]

2. Methodology

In our proposed algorithm there are two stages. In first stage extract the pixel from the images need to encrypt and find the value using chat map algorithm. The logistic chaotic map is a polynomial map of depth two. The mathematical definition of this map is as follows

$$X_{n+1} = (1-X_n). \qquad (1)$$

Therefore, our algorithm confuses the pixels using the logistic map and suppose that the size of the original image I is $M \times N$. In summary, the main idea is as follows.

Step-1. Suppose two arrays, R and Cl, are, respectively, used to record the rows and columns of an image:

$$R = \{1, 2, \ldots, M\} \text{ and } \quad Cl = \{1, 2, \ldots, N\} \qquad (2)$$

Step-2. Generate two pseudorandom sequences A and B, with respective sizes m and n, using the logistic map:

$A = \{a1, a2, \ldots,\}$, and $B = \{a1, a2, \ldots, an\}$. (3)

Step-3. Arrange the sequences A and B in descending order and record their locations. Thus, we can obtain the descending indexes, Index1 and Index2, of this pseudorandom sequence:

Index1 = $\{i1, i2, \ldots i_m\}$ andIndex2 = $\{j1, j2, \ldots, j_n\}$. (4)

The primary objective of this step is to find the index of the largest number from the sequence with size m and then store it in i_1. Next, we find the index of the second largest number and store it in i_2. We repeat this process until the descending sequence indexes are all stored in array Index 1. Similarly, we can obtain the index array Index 2 by arranging the sequence B in descending order and taking its index.

Step-4. According to indexes Index 1 and Index 2, we get new row R' and column C'. We can then obtain new arrays of the exchanged row R' and column C' and can confuse the image pixels.

In second step represent the values in DNA sequence as our table described. We out of all possibilities of allotting the binary digits the four deoxynucleotides "A," "T," "G," and "C", we chose 8 ways according to the given table below:

TABLE 2
deoxynucleotides to the binary bits

1	2	3	4	5	6	7	8
00-A	00-A	00-C	00-C	00-G	00-G	00-T	00-T
01-C	01-G	01-A	01-T	01-A	01-T	01-C	01-G
10-G	10-C	10-T	10-A	10-T	10-A	10-G	10-C
11-T	11-T	11-G	11-G	11-C	11-C	11-A	11-A

After the allotment of the deoxynucleotides, we now create an extract 4 deoxyneucleotides from the list X and exchange the positions of each, in the following steps:

Step 1: we extract the middle two letters and put them in the front and the mean position letters at the end. For example if the group of 4 letters is: “TCAG”, the combination becomes “CATG”.

Step 2: we now create an array of binary codes between the decimal digits of 0 to 15 and arrange them with the help of the logistic maps rule, explained above.

Step 3: let us say if we name the binary elements array Y. Then from 4 letters arranged above in step 1, we extract 2 letters every time For example: from the combination of “CATG”, firstly we take “CA” and allot them binary binary bits of 4 digits according to the table 3. And then we extract “TG” and allot them binary bits of 4 digits according to table 3. We can have possible 16 arrangements between the letters.

Step 4: For every four letters now we get 4+4=8bits of binary digits. Now we get an array of binary bits each of 8bits.

TABLE 3
Allotment of the group of two letters with 4-digit array given in the list Y.

	A	C	T	G
A	AA Y[0]	AC Y[1]	AT Y[2]	AG Y[3]
C	CA Y[4]	CC Y[5]	CT Y[6]	CG Y[7]
T	TA Y[8]	TC Y[9]	TT Y[10]	TG Y[11]
G	GA Y[12]	GC Y[13]	GT Y[14]	GG Y[15]

2.1 Encryption Algorithm

1. Import necessary libraries:
   - PIL is imported for image encryption.
   - Math is imported for mathematical operations.
2. Open two image files, main image and input_image1, for encryption. These images are assumed to be in the specified file path.
3. Extract the pixel map of the main image and the width and height of the image.
4. Initialize an empty list ‘C’ to store the XOR operation results of the corresponding pixel values in both images.
5. Iterate through each pixel in the main image:
   - Get the RGB values of the current pixel in both images.
   - Perform bitwise XOR operation on the corresponding RGB values and store the result in ‘C’.
6. Convert the decimal values in ‘C’ to 8-bit binary strings and store them in the list ‘D’.
7. Iterate through each binary string in ‘D’ and convert pairs of bits into characters (‘A’, ‘C’,’T’, ‘G’) according to a predefined mapping.
8. Create a new list ‘tmp’ to store the converted character pairs.
9. Split the characters in ‘tmp’ into two separate list ‘C’ and ‘D’.
10. For each pair of characters in ‘C’ and ‘D’ converted them back to binary strings based on reverse mapping.
11. Append the resulting binary strings to ‘tmp’.
12. Open a text file for writing and write the dimensions of the ‘main image’ (width and height) and the binary data from ‘tmp’.
13. Convert the binary data in ‘tmp’ back to integers and store them in the file ‘m’.
14. Iterate through each pixel in the ‘main image’.
    - Update the RGB values of the pixel with values from ‘m’.
15. Save the modified ‘main image’ as “encryptedIMG” and display it.
16. Print “ENCRYPTED DONE”.

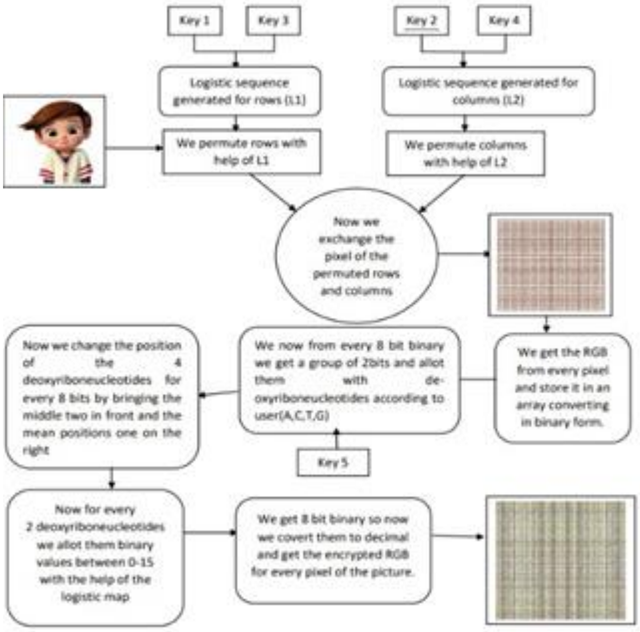

Key 1
Key 3
Key 2
Key 4
Logistic sequence generated for rows (L1)
Logistic sequence generated for columns (L2)
We permute rows with help of L1
We permute columns with help of L2
Now we exchange the pixel of the permuted rows and columns
We get the RGB from every pixel and store it in an array converting in binary form.
We now from every 8 bit binary we get a group of 2bits and allot them with de-oxyriboneucleotides according to user(A,C,T,G)
Key 5
Now we change the position of the 4 deoxyriboneucleotides for every 8 bits by bringing the middle two in front and the mean positions one on the right
Now for every 2 deoxyriboneucleotides we allot them binary values between 0-15 with the help of the logistic map
We get 8 bit binary so now we covert them to decimal and get the encrypted RGB for every pixel of the picture.

Fig-1: Encryption Algorithm Flowchart

2.2 Decryption Algorithm

1. Import the necessary libraries:
   - PIL is imported for image encryption.
   - Math is imported for mathematical operations
2. Open the main encrypted image ("E:/encrytedIMG.png") and another input image ("E:/picture.png").
3. Extract the pixel map of the main image.
4. Extract the width and height of the main image.
5. Initialize an empty list 'C' to store the RGB values of the pixels in the main image.
6. Loop through the main image's pixels using nested loops:
   - Get the RGB pixel values (r, g and b) at the current pixel location.
   - Convert each RGB pixel value to binary and ensure it is represented as an 8-bit string
   - Append these binary values to the list 'C'.
7. Create two empty lists, 'A' and 'B', to store the upper and lower halves of the binary representation of the RGB values, respectively.
8. Loop through the binary values in 'C':
   - Append the upper 4 bits (first half) to the list 'A'.
   - Append the lower 4 bits (second half) to the list 'B'.
9. Create empty lists 'C' and 'D' to store the decrypted values of 'A' and 'D' respectively.
10. Loop through the values in 'A':
    - Perform decryption based on specific patterns (e.g "0000" maps to 'AA').
    - Append the decrypted values to the list 'C'.
11. Loop through the values in 'B':
    - Perform decryption based on specific patterns.
    - Append the decrypted values to the list 'D'.
12. Initialize an empty list 'K' to store the combined values in 'C' and 'D', combining them as D[i][0] + C[i][0]+C[i][1]+ D[i][1] and append the results to 'K'.
13. Initialize an empty list 'tmp' to store the final binary representation of the combined values in 'K' where each character is converted to its binary representation of the combined values in 'K', where each character is converted to its binary representation of the combined values in 'K', where each character is converted to its binary representation (e.g. 'A' becomes '00').
14. Loop through the combined values in k, and for each character, convert it to binary and append to 'tmp'.
15. Initialize an empty list tmp1 to store the decimal values of the binary representations in tmp.
16. Loop through the binary values in tmp, convert each to decimal, and append the results to tmp1.
17. Calculate the number of RGB values in tmp1 (should be len(tmp1) // 3).
18. Initialize a count variable to keep track of the current position in tmp1.
19. Loop through the width and height of the main image:
    - Get the RGB pixel values r, g, and b from tmp1.
    - Get the corresponding RGB values r1, g1, and b1 from the input image.
    - Apply the XOR operation between each pair of values.
    - Set the resulting RGB values in the pixel map of the main image.
20. Save the decrypted image as "E:/decrypted_final.png" in PNG format.
21. Display the decrypted image.
22. Print "DECRYPTION DONE.. ☺☻" to indicate the completion of the decryption process

2. Results and Discussion

3.1 Analysis of Histogram

An image's histogram is used to show how an image's pixel values are distributed. Histogram analysis is used in image encryption to verify the consistency of the histogram of the cipher pictures. A cipher picture is thought to be more resistant to statistical attacks if it has a uniform histogram. The histograms of the main images IMG-1, and IMG-2, along with the corresponding ciphers, are displayed in Figure 2 and figure-3 respectively. The uniformly distributed histograms in the cipher images demonstrate the proposed algorithm's effectiveness in concealing the contents of plaintext images and its resistance to histogram analysis attacks.

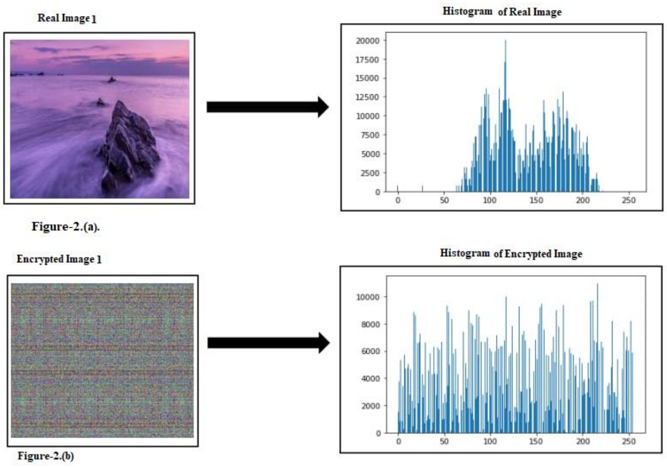


Figure-2(a) Original Image-1 and Its Histogram **(b)** Encrypted Image-1 and Its histogram

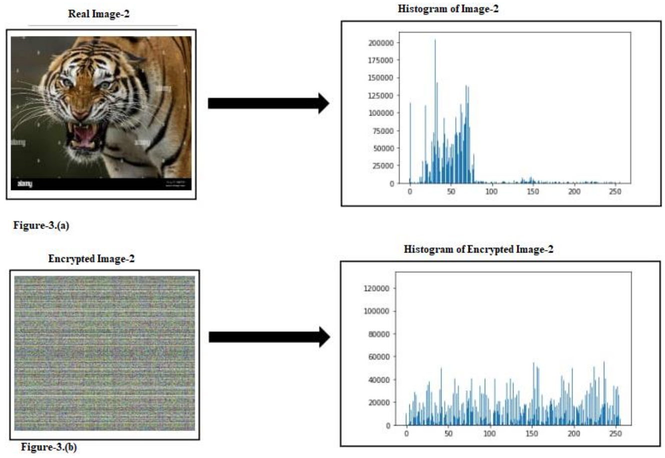


Figure-3 (a) Original Image-2 and Its Histogram **(b)** Encrypted Image-2 and Its histogram

3.2 Analysis of Correlation Coefficients

From the original image and the encrypted image, we randomly choose 2500 pairs of nearby pixels (in vertical, horizontal, and diagonal directions), and we compute the correlation coefficients of these pairs of pixels using the following formula [39]:

$$r_{xy} = \frac{cov\ (x,y)}{(\sqrt{D(x)})(\sqrt{D(y)})}$$

where

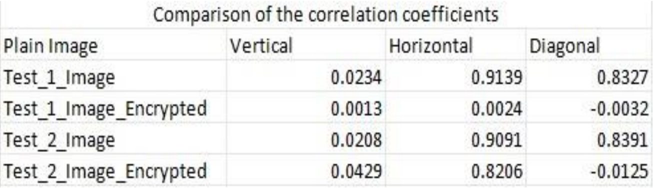
$$cov(x, y) = \frac{1}{N}\sum_{i=1}^{N} \quad (x_i - E(x))\ (y_i - E(y))$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i\ ,\quad D(x) = \frac{1}{N}\sum_{i=1}^{N} x_i\ (xi - E(x))^2$$

Table 4 shows the results of correlation coefficients of two adjacent pixels

TABLE 4

The comparison of the correlation coefficients of IMG-1 and IMG-2.



Comparison of the correlation coefficients			
Plain Image	Vertical	Horizontal	Diagonal
Test_1_Image	0.0234	0.9139	0.8327
Test_1_Image_Encrypted	0.0013	0.0024	-0.0032
Test_2_Image	0.0208	0.9091	0.8391
Test_2_Image_Encrypted	0.0429	0.8206	-0.0125

4. Conclusion and Future Scope

In this paper we proposed an image encryption using 2-D chaotic map and DNA encoding technique. The confusion and diffusion method of this proposed algorithm is efficient and novel. The uniformly distributed histograms in the cipher images demonstrate the proposed algorithm's effectiveness in concealing the contents of plaintext images and its resistance to histogram analysis attacks. The correlation coefficient is also low which indicate the algorithm is good.

5. References

[1] Shah, T., Jamal, S.S., et al.: An improved chaotic cryptosystem for image encryption and digital watermarking. Wireless Personal Commun. 110(3), 1429–1442 (2020)

[2] 5. Zeng, W., Lei, S.M.: Digital image scrambling for image coding systems. US Patent 6,505,299 (2003)

[3] 6. Morkel, T., Eloff, J.H., Olivier, M.S.: An overview of image steganography. In: ISSA, vol 1 (2005)

[4] 7. Bhowmik, S., Acharyya, S.: Image cryptography: The genetic algorithm approach. In: 2011 IEEE International Conference on Computer Science and Automation Engineering, IEEE, vol. 2, pp. 223–227 (2011).

[5] Lili Liu, Qiang Zhang, Xiaopeng Wei "A RGB image encryption algorithm based on DNA encoding and chaos maps", Computers & Electrical Engineering, vol. 38, issue 5, pp. 1240–1248, 2012.

[6] P. R. Sankpal and P. A. Vijaya, "Image Encryption Using Chaotic Maps: A Survey", 2014 Fifth International Conference on Signal and Image Processing, Jeju Island, pp. 102-107, 2014.

[7] Stallings W, "Cryptography and network security principles and practice", Int. J. Eng. Comput.Sci., pp. 121–136, 2012.

[8] Noshadian, S., Ebrahimzade, A., Kazemitabar, S.J.: Optimizing chaos based image encryption. Multimedia Tools Appl. 77(19):25,569–25,590 (2018)

[9] Furht, B., Muharemagic, E., Socek, D.: Multimedia encryption and watermarking, vol 28. Springer Science & Business Media (2006)

[10] Peng, Z., Yu, W., Wang, J., Zhou, Z., Chen, J., Zhong, G.: Secure communication based on microcontroller unit with a novel fivedimensionalhyperchaotic system. Arab. J. Sci. Eng., pp. 1–16 (2021)

[11] Som, S., Dutta, S., Singha, R., Kotal, A., Palit, S.: Confusion and diffusion of color images with multiple chaotic maps and chaosbased pseudorandom binary number generator. Nonlinear Dyn. 80(1), 615–627 (2015)

[12] Xy, Wang, Xm, Bao: A novel block cryptosystem based on the coupled chaotic map lattice. Nonlinear Dyn. 72(4), 707–715 (2013)

[13] Lan, R., He, J., Wang, S., Gu, T., Luo, X.: Integrated chaotic systems for image encryption. Signal Processing 147, 133–145 (2018)

[14] Sankpal, P.R., Vijaya, P.: Image encryption using chaotic maps: a survey. In: 2014 Fifth international Conference on Signal and image Processing, IEEE, pp. 102–107 (2014)

[15] Wei-Bin, C., Xin, Z.: Image encryption algorithm based on Henon chaotic system. In: 2009 International Conference on Image Analysis and Signal Processing, IEEE, pp. 94–97 (2009)

[16] Krishna, P.R., Teja, C.V.S., Thanikaiselvan, V., et al.: A chaos based image encryption using tinkerbell map functions. In: 2018 Second International Conference on Electronics, pp. 578– 582. Communication and Aerospace Technology (ICECA), IEEE (2018)

[17] Pak, C., Huang, L.: A new color image encryption using combination of the 1d chaotic map. Signal Process. 138, 129–137 (2017)

[18] Hua, Z., Jin, F., Xu, B., Huang, H.: 2d logistic-sine-coupling map for image encryption. Signal Process. 149, 148–161 (2018)

[19] Shan, L., Qiang, H., Li, J., Zq, Wang: Chaotic optimization algorithm based on tent map. Control Decision 20(2), 179–182 (2005)

[20] Fang, D., Sun, S.: A new secure image encryption algorithm based on a 5d hyperchaotic map. Plos one 15(11):e0242,110 (2020)

[21] Mira, C. *Chaotic Dynamics: From the One-Dimensional Endomorphism to the Two-Dimensional Diffeomorphism*; World Scientific: Singapore, 1987. [**Google Scholar**]

[22] Avrutin, V.; Gardini, L.; Sushko, I.; Tramontana, F. *Continuous and Discontinuous Piecewise-Smooth One-Dimensional Maps: Invariant Sets and Bifurcation Structures*; World Scientific: Singapore, 2019.

[23] Sharma M, Kowar MK. Image encryption techniques using chaotic schemes: a review.Int J EngSci Technol. 2010;2(6):2359–63.

[24] Parvaz R, Zarebnia M. A combination chaotic system and application in color image encryption. Opt Laser Technol. 2018;101:30– 41.

[25] Xiong H, Shang P, He J, Zhang Y. Complexity and information measures in planar characterization of chaos and noise. Nonlinear Dyn. 2020;100(2):1673–87

[26] Safi HW, Maghari AY (2017) Image encryption using double chaotic logistic map. In: Proceedings 2017. International Conference on Promising Electronic Technologies (ICPET), pp 66–70.

[27] Krishnamoorthi R, Murali P. A selective image encryption based on square-wave shuffling with orthogonal polynomials transformation suitable for mobile devices. Multimed Tools Appl. 2017;76(1):1217–46.

[28] Mansouri A, Wang X. A novel one-dimensional sine powered chaotic map and its application in a new image encryption scheme. InfSci (N Y). 2020. https:// doi.org/ 10. 1016/j. ins. 2020.02. 008.

[29] Hua Z, Zhou B, Zhou Y. Sine chaotification model for enhancing chaos and its hardware implementation. IEEE Trans Ind Electron. 2019;66(2):1273–84. 30. Hua Z, Yi S, Zhou Y, et al. Designing hyperchaotic cat maps with any desired number of positive lyapunov exponents. IEEE Trans Cybern. 2018;48(2):463–73.

[30] Bhatnagar G, Wu QMJ. Selective image encryption based on pixels of interest and singular value decomposition. Digit Signal Process. 2012;1:1–16.

[31] deCarvalho RE, Leonel ED. Squared sine logistic map. Phys A Stat Mech Appl. 2016;463:37–44.

[32] Han C. An image encryption algorithm based on modified logistic chaotic map. Opt Int J Light Electron Opt. 2019;181:779–85

[33] ua Z, Zhou Y, Huang H. Cosine-transform-based chaotic system for image encryption. InfSci (NY). 2019;480:403–19.

[34] Wu J, Liao X, Yang B (2017) Color Image Encryption Based on Chaotic Systems and Elliptic Curve ElGamal Scheme, Signal processing, PII: S0165–1684(17)30134 2, https://doi.org/10.1016/j. sigpro.2017.04.006

[35] Mansouri A, Wang X. A novel one-dimensional sine powered chaotic map and its application in a new image encryption scheme. InfSci (N Y). 2020. https:// doi.org/ 10. 1016/j. ins. 2020.02. 008.

[36] Lindahl, Tomas. "The world of DNA in glycol solution." Nature Reviews Molecular Cell Biology 17.6 (2016): 335-336

[37] Karimi, M., &Haider, W. (2017). Cryptography using DNA Nucleotides. Int. J. Comput. Appl, 168, 16-18

[38] Srinivasu, P. Naga, andSeshadriRao. "A Multilevel Image Encryption based on Duffing map and Modified DNA Hybridization for Transfer over an Unsecured Channel." International Journal of Computer Applications 120.4 (2015)

R. Rhouma, S. Meherzi, S. Belghith, OCML-based colour image encryption, Chaos, Solitons & Fractals 40 (1) (2009) 309–318.

BIOGRAPHIES

Mr. Sanjit Mazumder obtained his MTech in Computer Science & Engineering from West Bengal University of Technology and presently pursing his PHD. His field of interest is in Cryptography and Network Security, Machine Learning etc. He has more than eight publication in different international journals

Mr. Sada Nanda Banerjee Obtained his BTech in Computer Science & Engineering from MAKAUT. His field of interest is in Network security, ML

https://isjem.com/pay-online/