Analysis of Encrypted data using DNA Cryptography and Chaotic Systems

Madhav Dhakal[1,2*], Subarna Shakya[3]

[1]Graduate School of Science and Technology, Mid-West University, Surkhet, Nepal

[2]Central Department of Computer Science and Information Technology, Tribhuvan University, Nepal

[3]Department of Electronics and Computer Engineering, Institute of Engineering, Pulchowk Campus Tribhuvan University, Kathmandu, Nepal

**Correspondance: madhav.dhakal@mu.edu.np; Ph. 9841452329*

Keywords

DNA Computing
Cryptography
Chaotic system
Data security

Received: 29 October 2024
Revised: 15 November 2024
Accepted: 25 November 2024

ISSN: 3059 - 9687

Abstract

In the modern world of technology, ensuring security of information is essential for security in both transmitter and receiver of data. In this work, to achieve the goal hybridization concept of DNA based cryptography and chaotic system is implemented. This involves converting binary data to DNA bases, followed by transcription from DNA to mRNA, translation from mRNA to tRNA, and finally, transforming tRNA into reverse tRNA along with this chaotic sequence generated from initial values of chaotic system is implemented to generate the permutation value. Here, the encryption method supports each block sizes of 256 bits and results during the algorithm implementation demonstrated strong security, with an average avalanche effect 76.64%, indicating high sensitivity while key changes. Even a tiny modification to the key value can render the plain message undecryptable, thereby maintaining the confidentiality of the message. Overall, algorithm proposed in this study enhances the security of data transmission over untrusted public networks, offering robust encryption methods that are resistant to modern attacks. Key stakeholders include data security professionals, policymakers, and all individuals who are interested in sending and receiving private data with confidentiality and integrity.

Introduction

Cryptography is regarded as one of the most effective remarkable methods for securing data. It plays a significant role in our daily technological lives; especially while conducting online transactions, it is essential to safeguard information from hacking, interference, and noise. Its aim is not only to protect data during communication but also to ensure the safety and reliability of the system. It is concerned with providing security in the context of IT systems as well in every entity where electronic data is exchanged (Kumar and Wolling, 2006). Its primary purpose is to enable the transfer of information from the sender to the receiver over an unreliable communication channel, ensuring that unauthorized individuals cannot access the original data. All cryptography fundamentally based on the key and classified on the basis of key size used for secure the data. The security of a cryptosystem relies entirely on the quality of the key rather than the design of the encryption algorithm. Encryption and Decryption are fundamental processes in cryptography.

Encryption is the process of converting the original readable data (plain message) into an unreadable form (cipher text), while decryption is the reverse process of encryption. Many modern cryptographic approaches, such as quantum cryptography and DNA cryptography, are popularly used to secure data during transmission.

Symmetric key encryption and asymmetric key encryption are two types of cryptography. Where, in symmetric key based cryptography, same single key i.e. private key is used by both sender and receiver parties for encrypt and decrypt the message. The key is kept secret between two parties. Whereas, two separate keys, public key and private key is used for encrypt and decrypt the message in sender and receiver in asymmetric key based cryptography. Here, public key is publicly available to all who want interest with this key but the private key is kept secret only between concern two parties. Asymmetric methods are more robust than symmetric ones due to their longer key lengths and more complex underlying operations; however, they require more processing time (Das and Namasudra, 2022). A stronger and more secure encryption method is necessary to improve data security and ensure its confidentiality and integrity.

Traditional based cryptography including AES, DES, 3DES, RSA are suitable for text-based data and due to the high volume of information and randomness in image-based data these techniques are not highly appropriate. Thus, for image data chaotic system is appropriate for encryption purpose. In present, among the various data security technique, DNA computing technology has been applied to the cryptography. DNA-based encryption has emerged as one of the most promising and widely adopted technologies for data security with the concept of biological structure. Numerous researchers have explored DNA computing due to its intricate structure and unique characteristics for efficient and secure transmission of data through the unsecure channel. While some have focused on leveraging DNA computing, others have integrated the biological properties of DNA strands and sequences into their encryption algorithms. This method leverages biological technology for encryption, using DNA as the carrier medium and modern biological techniques as application tools. However, it faces challenges such as costly experimental equipment, complex procedures, and intricate biotechnology, limiting its widespread use in cryptography. To address these issues, certain DNA computing operations are employed to obscure information (Wang *et al.*, 2015).

In this research, we propose an algorithm based on DNA computing and 3D chaotic systems to secure data and test its avalanche effect and the execution time of an algorithm. DNA is employed to store the digital data in the form of DNA nucleotides and chaotic systems generate the high randomness among those data. Through avalanche effect analysis, we clarify the impact of the cryptographic algorithm. The general concept behind the avalanche is where flipping a single bit in the input results in a significant change in the output, with approximately half of the output bits being altered. For instance, a one-bit change in the plaintext leads to the output bits changing with a probability of ½ in the cipher text.

This work aims to analyze the execution time and security mechanism of encrypted data with the

concept of key sensitivity analysis and determine the avalanche effect of data using DNA computing and chaotic system.

Related Works

Various research studies in DNA-based cryptography explore its diverse applications, including its use in complement operations, digital encoding, polymerase chain reactions, and as a security mechanism. DNA cryptography combines mathematical principles with biotechnology, making data transmission more secure compared to traditional cryptographic methods. Most researchers in data security focus on combining DNA-based computing with other technologies, such as traditional cryptography and chaotic systems. Chaotic systems are also implemented for data security purposes with the concept of cryptography due to their high sensitivity to initial conditions and system parameters. This means that even a tiny change in input data can cause a drastic difference in the output. Chaotic systems are particularly used in fields such as biomedical engineering, random number generation, data communication, quantum and fuzzy (Cavusoglu *et al.*, 2019).

This research focuses on the hybridization of DNA-based cryptography with a chaotic system to determine the execution time of the proposed algorithm and to test the randomness of the cipher text generated by the implemented algorithm with the concept of entropy and avalanche effect analysis.

The development of DNA computing has introduced a new research paradigm in cryptography for various researchers. The theoretical use of the molecular computer developed by Feynman is considered the beginning of biological computing. In 1994, Adleman was the first to implement DNA computing to solve the "Seven-Vertex Hamiltonian Path" problem, demonstrating the potential of DNA molecules for computation with the advantages of vast parallelism and extraordinary information density (Fu & Beigel, 1999).

The authors in (Babaei, 2013) presents a hybrid encryption algorithm (OTP) for text and image encryption by integrating chaos theory with DNA computing. In this approach, each bit or character of the plaintext is encrypted through modular addition with a corresponding bit or character from a random key generator. If the secret key is genuinely random, the cipher text becomes theoretically unbreakable. A logistic chaotic map is used to generate the input for the OTP algorithm. DNA computing provides benefits such as immense data storage and parallel processing capabilities, while chaos theory introduces high sensitivity to initial conditions, significantly boosting security. The study emphasizes that for optimal security, each secret key in the OTP algorithm must be used only once.

A novel text and image encryption technique was proposed in work (Signing *et al.*, 2021) that leverages the unpredictable chaotic behavior of a Jerk system with a hump structure, combined with DNA coding. Through both analytical and numerical analyses, the chaotic Jerk system is shown to exhibit complex dynamics, including hysteresis, bifurcation bubbles, and the unique presence of six symmetrical equilibrium points, despite the system's inherent asymmetry. The encryption process harnesses this chaotic behavior alongside DNA sequences to effectively encrypt and decrypt textual and digital image data.

In Niyat & Moattar (2020), a method was proposed to improve communication security by creating encryption algorithms based on the features of DNA computing and three distinct chaos generators i.e. Logistic Map, Pinchers Map, and Sine-Circle Map for encryption in various scenarios, with performance assessed through time and entropy metrics. This paper also presents an innovative approach for encrypting text messages using pseudorandom numbers generated by a hyperchaotic sequence. These numbers are used as an index to encode the message via DNA coding rules and DNA computing operations.

The authors in (Oleiwituama *et al.*, 2021), a novel chaos-based bit-level permutation scheme using the concept of DNA computing and hyperchaotic system is proposed to enhance text encryption by improving the diffusion process and reducing computational complexity. Unlike traditional text cipher permutations, the proposed scheme leverages bit-level manipulation to maintain strong security while lowering the complexity of the encryption process. The research findings reveal that DNA cryptography is highly resistant to differential attacks and can effectively withstand conventional linear, noise, and cropping attacks. The use of DNA “addition operations” instead of binary operations enhances the algorithm's efficiency and unpredictability.

Preliminaries

DNA Coding

DNA is a double-stranded helix composed of two complementary strands purines and pyrimidines. The purine bases include adenine (A) and guanine (G), while the pyrimidine bases consist of thymine (T) and cytosine (C) (Kumar and Namasudra, 2023). These A, C, G, and T are key components of cells of living organisms that carry genetic information, can also be utilized used to encrypt and generate the randomness of a plain message through the technique of DNA cryptography. DNA nucleotides, Adenine match with the Thymine and Cytosine match with the Guanine. During the time of conversion into DNA bases, input data are first converted into binary form and after that converted into DNA bases. DNA bases have been characterized with binary digits 0 and 1. For illustration, Consider the encryption of character ‘M‘ and the binary value of ‘M’ is 01001101, by using the DNA encoding rule, the character is represented as ‘CATC’. DNA molecules possess large data storage capacity, with 1 gram capable of holding approximately 10^{21} DNA bases, equivalent to 10^{8} terabytes, hence few grams of DNA have the possible to store the entire world's data (Gehani *et al.*, 2003). This vast storage potential allows DNA cryptography to significantly minimize space complexity.

Chaotic System

Chaos is an important feature of a nonlinear dynamic system that is sensitive to the initial condition. This means even tiny changes occur in the initial value, then produce the drastic, unpredictable changes in the output result. Due to its unpredictability and extreme sensitivity to initial conditions, the chaotic system is also referred to as the "butterfly effect." Chaos theory has been applied to secure large amounts of data, such as images or documents, by utilizing fundamental cryptographic techniques, including permutation and substitution (Fridrich, 1997) Initial value sensitivity, Lyapunov exponent and Bifurcation diagram is the common features of chaotic system. Lyapunov exponent measure

the rate of convergence or divergence of trajectory in dynamic system. The value of Lyapunov exponent may be positive, zero or negative, if the chaotic system is three dimensional then there must be at least one positive value among the tree values and if it is four dimensional i.e. hyperchaotic then it must have at least two positive values from the four values of Lyapunov exponent. Where, positive value indicates the chaotic system, zero means periodic system and negative represent the stable equilibrium of system (Sproot, 2010).

Methodology

Data

The main purpose of this research is to encrypt and decrypt data, including both text and images, using the proposed algorithm. The text dataset is sourced from Kaggle, while the grayscale image datasets are taken from the SIPI Image Database (https://sipi.usc.edu/database/database.php).

Encoding Rules

In binary representation, the digits 0 and 1 are opposites, which leads to pairs such as 01 and 10 being considered opposites, as well as 00 and 11. Accordingly, the nucleic acid bases are encoded as follows: Guanine (G) as 10, Cytosine (C) as 01, Adenine (A) as 00, and Thymine (T) as 11. While there are 24 possible encoding patterns (4! = 24), only eight encoding and decoding rules for DNA bases adhere to the complementary pairing principle, using Watson-Crick complementary rules, which are specified below:

Table 1. Binary to DNA Encoding Rules.

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

Encryption and Decryption Algorithm

Encryption Algorithm

1. Transform the Input data
   - If the input data is text, then, convert in to ASCII value and then binary code
   - Otherwise, if the data is image, then take the size of image of 512 x 512 and translate the pixel value in to binary form
2. Size of binary data of is divided into ‘n’ numbers
3. Take the key with same size as the size of input data
4. Perform mathematical XOR operation between binary value of plain text and key value.
5. Consider the initial value and parameter values of chaotic system
6. Use the chaotic system value to generate the seed values of permutation
7. Apply DNA rule and perform the operation of Central Dogma of Molecular Biology:
   a. *Generate mRNA Sequence:* The sequence is generated with using the DNA nucleotides and replace the Thymine (T) by Uracil (U).
   b. *Generate tRNA Sequence:* For this operation, swap A with U and vice versa, U with A and vice versa, G with C and vice versa.

c. *Obtain Reverse tRNA Sequence*: For this, perform the inverse operation of step-a
d. *Convert to Binary:* Transform the result obtained from step-c into binary form

8. Transform encrypted form of binary data to a cipher text

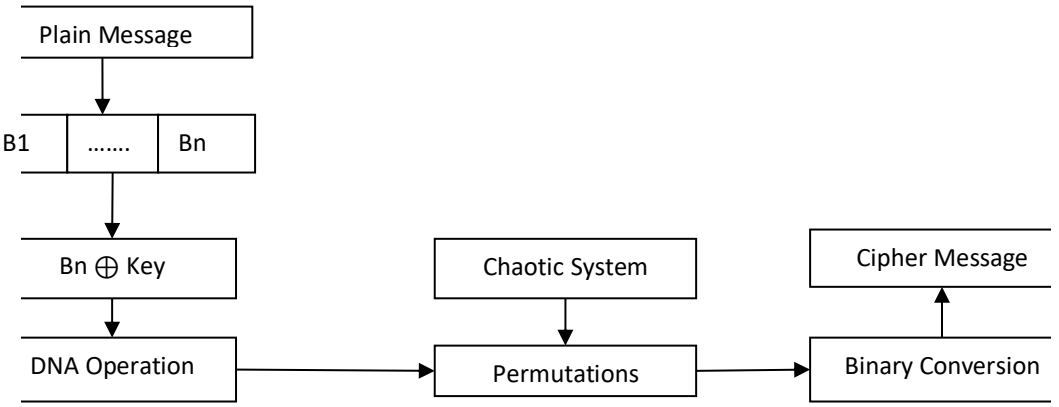

Fig. 1. Working flow of encryption.

Decryption Algorithm

The decryption process takes place on the receiver's end and follows the reverse of each encryption step. Specific details of each decryption step are not outlined in this section, but the process begins by converting the ciphertext back into binary form, followed by reversing the DNA encoding and Central Dogma transformations. An XOR operation is then performed using the original key. Finally, the binary data is used to reconstruct the original data, whether it be text or an image.

Generation of Permutation Table

To generate a permutation box, two inputs are used: a seed value and the number of bits in the permutation box. First, an empty list, 'nums', is initialized to store the unique random numbers. The seed value is obtained from the initial value and system parameters of chaotic system and set as input seed. After that, random numbers are generated. If the generated random number, 'rand_num', is not already present in 'nums', it is appended to the list. This process repeats until' nums' contains the required number of unique random numbers based on the 'bit' input.

Table 2. Permutation table of 256-bit size.

0, 55, 138, 71, 78, 79, 113, 119, 52, 36, 116, 97, 100, 225, 94, 191, 183, 141, 108, 124, 235, 89, 57, 195, 6, 35, 209, 88, 50, 18, 1, 245, 198, 103, 38, 54, 244, 58, 11, 205, 23, 131, 67, 105, 146, 151, 188, 149, 186, 48, 189, 53, 15, 4, 253, 10, 102, 152, 111, 181, 5, 185, 194, 87, 49, 251, 210, 180, 168, 26, 162, 217, 127, 42, 246, 255, 236, 170, 145, 234, 110, 249, 142, 33, 196, 76, 65, 164, 123, 25, 243, 91, 77, 228, 173, 104, 230, 215, 16, 19, 128, 254, 167, 47, 199, 221, 229, 178, 8, 93, 92, 66, 136, 248, 233, 132, 17, 204, 211, 37, 163, 192, 252, 231, 122, 69, 134, 24, 80, 75, 175, 242, 98, 147, 159, 82, 27, 156, 86, 140, 160, 190, 219, 214, 129, 212, 165, 250, 187, 95, 84, 13, 3, 200, 207, 70, 238, 154, 9, 7, 112, 126, 232, 96, 193, 177, 64, 45, 216, 222, 172, 99, 59, 68, 223, 73, 150, 226, 239, 120, 60, 101, 158, 62, 237, 130, 81, 144, 176, 115, 32, 30, 106, 227, 43, 61, 220, 31, 208, 139, 56, 241, 114, 143, 39, 14, 135, 74, 148, 117, 153, 2, 21, 44, 213, 83, 184, 118, 155, 40, 169, 20, 202, 85, 121, 201, 46, 72, 137, 107, 182, 161, 157, 197, 34, 109, 22, 206, 29, 133, 179, 218, 166, 240, 171, 63, 224, 174, 125, 247, 51, 28, 12, 203, 90, 41

Experimental Testing and Security Analysis

Experimental Environment

The proposed method was implemented using Python, and the results were analyzed and tested on a computer equipped with an Intel(R) Core(TM) i7-8550U, CPU @ 1.80GHz, 16 GB of RAM, and a 64-bit Windows 10 operating system.

Text data Analysis

Encryption/Decryption Time with Throughput

Encryption time refers to the time required to convert plaintext into ciphertext and decryption operation is performed at the receiver side. The decryption time indicate the time taken to obtain the original message from the cipher text message. The efficiency of an encryption algorithm is inversely related to the time it takes for this process. In this test, two datasets of varying sizes are used, and the encryption and decryption times are measured across 100 iterations. The average time for each dataset is then recorded for analysis.

The efficiency of an algorithm can be assessed by examining its throughput. Throughput is directly linked to the algorithm's performance (Thabit *et al.*, 2021).

$$Throughput = Plaintext\ size / Encryption\ time$$

Table 3. Performance Analysis of Algorithms.

Dataset Size	Average Encryption Time (In Sec.)	Average Decryption Time (In Sec.)	Throughput
10.76 KB	0.1318	0.0876	81.6350
172.75KB	2.1609	1.4378	79.9444

Entropy Analysis

In data security, entropy analysis is conducted to evaluate the level of randomness in the ciphertext, helping to assess the strength and unpredictability of the encryption (Zhang and Tang, 2018). In the proposed scheme, the source message is composed of four symbols: A, G, C, and T. The calculated entropy values were $1.9986 \approx 2$, which is a highly efficient ratio given that there are four possible outcomes, reflecting a balanced distribution of probabilities. In this research, entropy is calculated using the Shannon entropy formula as follows:

$$H(x) = -\sum_{i=0}^{L-1} p(x_i) log p(x_i)$$

Where, L is the total number of symbols,$x_i \in p(x)_i$ is the probability of occurrence of x_i

Key sensitivity Analysis

To ensure data security, an encryption system must exhibit key sensitivity, meaning that even slight alterations to the key should prevent the correct recovery of the original data, thus blocking attackers from exploiting identical keys to compromise the system (Wang *et al.*, 2023) . In this proposed work, even minimal changes to the original key, result in completely different in cipher text and becomes impossible to recover the original data. The experimental results confirmed the validity of this approach.

Table 4. Analysis of key sensitivity.

Key value	Decrypted Text
secretkey	This is my Plaintext
secretkez	뢎粭峆\uec6a忽읊戈強앱읃高碥ㅋl曩婓멱룡糞嵃\uec07徑뭢矛彝었뭷駹硇ㅗ書婔멱(Grabled Information)

By analyzing the Table-4, Decrypting the ciphertext using a slightly altered key will result in a significant amount of garbled information, demonstrating the system's sensitivity to even minor key changes.

Brute force Attack Analysis

A brute force attack is a method used for password cracking, where an attacker systematically attempts multiple guesses to gain unauthorized access to legitimate information by exploiting weak or compromised passwords. In this approach, an attacker possesses an encrypted file that contains an encryption key needed to unlock the password. To decrypt the password, the attacker systematically tries every possible combination of passwords until they find the correct one, which ultimately results in accessing the decrypted file (Gautam and Jain, 2015)

In this study, there are 'n' blocks, each consisting of 256 bits. Consequently, there are 2^256 possible key combinations for each block, resulting in a total of n * 2^256 combinations for cryptanalysis.

The time taken for a brute force attack is computed based on the CPU performance capacity and the total time in seconds per year.

Assumptions

- CPU performance capacity: 1.8 GHz = 1.8×10^9 cycles per second (Hz).
- Average time in a year: 3×10^7 seconds.

Formula [18]: $$T = \frac{n*2^{Key\ Length}}{CPU\ speed \quad x\ Seconds\ per\ year}$$

Where 'n' is the number of blocks

Calculation for a 256-bit key:

$$\frac{n*2^{256}}{(1.8*10^{9})\ x(\ 3*10^{7})} = n * \frac{1.15x10^{77}}{0.054x10^{18}} = n * \ 21.2962 * 10^{59} \text{ years}$$

This result demonstrates the infeasibility of brute force attacks on such strong encryption keys.

Avalanche effect Analysis

The Avalanche Effect refers to a phenomenon where even a small modification in the input data results in a significant and widespread change in the output. An ideal value for this effect is close to 0.5, indicating that approximately half of the output bits change when a single input bit is altered. This property makes it extremely difficult for an attacker to predict the output based on partial knowledge of the input, thereby strengthening the security of the encryption system (Mohamed *et al.*, 2022; Devi & Kumar, 2018).

In this study, the four DNA nucleotides A, C, G, and T are used to represent information in cipher text. Due to the nature of 16-bit Unicode

encoding, many bits remain unchanged, so altering a single bit may not effectively demonstrate the avalanche effect. Therefore, the avalanche effect is calculated by modifying characters rather than individual bits and average impact of avalanche in the proposed coding scheme is 76.64%, providing a more accurate assessment of how small changes impact the overall encryption process.

Experimental Testing and Security Analysis of Image Data

Encryption/Decryption Time

The total time taken by the algorithm to encrypt and decrypt the image is dependent on the size of the image. Here, 512 x 512 size single image is used for experimental purposes, which is taken from the image database SIPI. The obtained detail of time is recorded as:

Table 5. Encryption/Decryption time of image.

Image (512 x512)	Average Encryption Time (In Sec.)	Average Decryption Time (In Sec.)
Man	6.0312	6.1406

Histogram Analysis

In image security, a histogram is used to analyze the distribution of pixels in a cipher image. A properly designed encryption technique will result in a uniformly distributed pixel pattern in the cipher image, indicating that the encryption technique is effective (Hu *et al.*, 2017). In this research, distribution of pixels in cipher image is given in Figure 2.

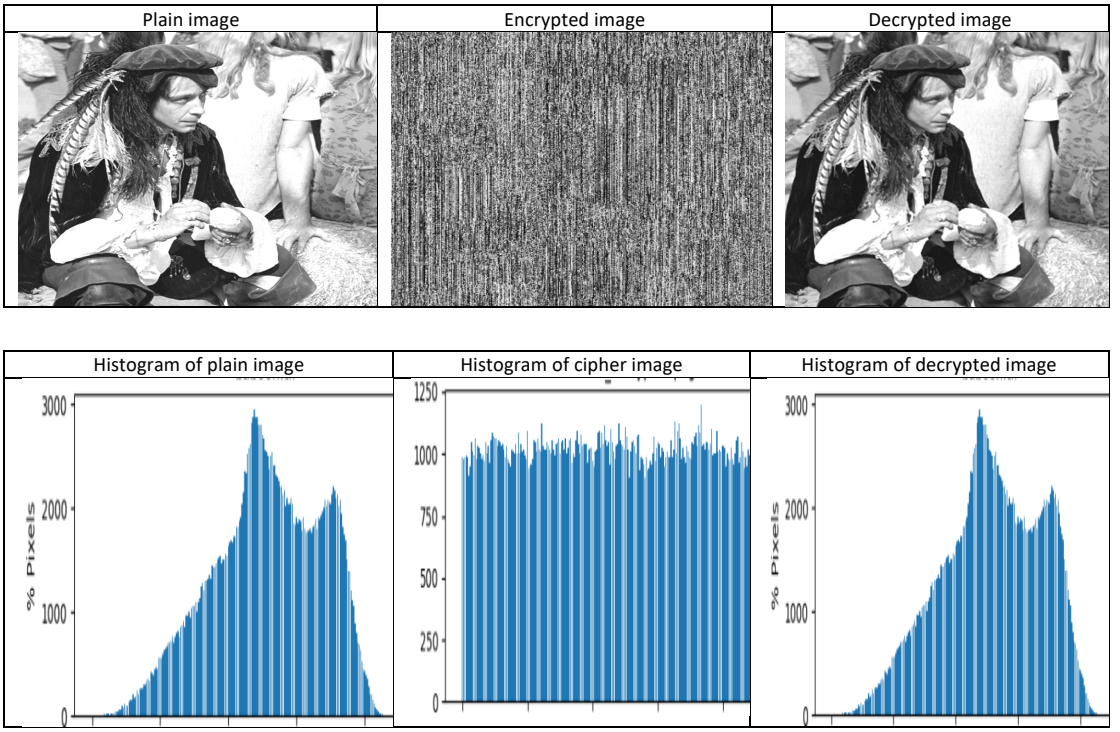

Fig. 2. Histogram analysis of image.

Chi-square Test

The Chi-square test is used to quantitatively assess the uniformity of pixel distribution in an image. In a grayscale image, the ideal Chi-square test value is 293.24 with significance level 0.05. The pixels in an image are considered uniformly distributed when the Chi-square value is less than this ideal value. In the present research, the Chi-square value of the selected image is below the ideal value, indicating that the pixel distribution in the cipher image is uniform, which is a sign of maintaining the image's secrecy.

Table 6. Chi-square value of image

Image(512 x 512)	Chi square(χ^2)value	P-value
Man	236.25	0.79

Correlation Analysis

Correlation indicates to the connection between pixels of image. An encryption technique is considered effective when the correlation between the pixels in the image is low, i.e., no correlation in randomized image (Zefreh, 2020). Low correlation makes it difficult to extract information, thus preserving the confidentiality of the data. In this research, correlation of image in horizontally, vertically and diagonally is given Table 5.

Table 7. Correlation of image before and after encryption

Image(512 x 512)	Directions	Plain Image	Cipher Image
man	Horizontal	0.9561	0.0080
	Vertical	0.9597	0.3570
	Diagonal	0.8820	0.0013

Table 7 shows that the correlation values in the plain image are higher, i.e., close to one, whereas the values in the cipher image are lower, i.e., closer to zero. This demonstrates that the proposed coding scheme effectively maintains randomness in the pixels of the cipher image, thereby ensuring confidentiality.

Entropy Analysis

In a grayscale image, pixel intensities range from 0 to 255. The theoretical value of information entropy is 8. In this research, the entropy of the plain image and cipher image is listed as Table 6.

Table 8 shows the entropy value of the cipher image is close to the ideal value, indicating that the pixel values in ciphertext image are well-randomized in the image.

Table 8. Entropy analysis of image.

Image	**Entropy**	
	Original image	**Cipher image**
Man	7.2367	7.9975

Conclusion

In today's internet-enabled world, all types of data, including text and images, are transmitted through public networks, creating a risk of vulnerability. During data transmission, both the sender and receiver must pay close attention to the confidentiality of the transmitted data.

In this study, the main goal is to achieve the secure encryption scheme during transmitting between two parties; this is achieved by integrating the DNA based computing and chaotic system in novel way. Here, in text data encryption; entropy analysis, avalanche effect analysis, key sensitivity analysis, and brute force attack analysis are used for the security analysis. And for image data security, additional tests such as histogram analysis, chi-square test, correlation analysis, and entropy analysis are performed to assess the randomization of pixels in the image. To further evaluate the performance of the proposed algorithm, encryption and decryption times are recorded, and the results indicate that the algorithm executes efficiently.

References

Babaei, M. (2013). A novel text and image encryption method based on chaos theory and DNA computing. *Natural Computing, 12*(1), 101–107. https://doi.org/10.1007/s11047-012-9334-9

Çavuşoğlu, Ü., Panahi, S., Akgül, A., Jafari, S., & Kaçar, S. (2019). A new chaotic system with hidden attractor and its engineering applications: Analog circuit realization and image encryption. *Analog Integrated Circuits and Signal Processing, 98*(1), 85–99. https://doi.org/10.1007/s10470-018-1252-z

Das, S., & Namasudra, S. (2022). A novel hybrid encryption method to secure healthcare data in IoT-enabled healthcare infrastructure. *Computers & Electrical Engineering, 101*, 107991. https://doi.org/10.1016/j.compeleceng.2022.107991

Devi, P. B., & Kumar, D. R. K. (2018). Inspired Feistel DNA-based crypto system using D-Box. *Journal of Cryptographic Research, 13*(5).

Folifack Signing, V. R., Fozin Fonzin, T., Kountchou, M., Kengne, J., & Njitacke, Z. T. (2021). Chaotic jerk system with hump structure for text and image encryption using DNA coding. *Circuits, Systems, and Signal Processing, 40*(9), 4370–4406. https://doi.org/10.1007/s00034-021-01665-1

Fridrich, J. (1997). Image encryption based on chaotic maps. *IEEE International Conference on Systems, Man, and Cybernetics*, 1105–1110.

Fu, B., & Beigel, R. (1999). Length bounded molecular computing. *Biosystems, 52*(1–3), 155–163. https://doi.org/10.1016/S0303-2647(99)00042-8

Gautam, T., & Jain, A. (2015, November). Analysis of brute force attack using TG—Dataset. *2015 SAI Intelligent Systems Conference (IntelliSys)*, 984–988. IEEE. https://doi.org/10.1109/IntelliSys.2015.7361263

Gehani, A., LaBean, T., & Reif, J. (2003). DNA-based cryptography. In *Proceedings of a cryptography conference* (pp. 167–188). https://doi.org/10.1007/978-3-540-24635-0_12

Hu, T., Liu, Y., Gong, L.-H., Guo, S.-F., & Yuan, H.-M. (2017). Chaotic image cryptosystem

using DNA deletion and DNA insertion. *Signal Processing, 134*, 234–243. https://doi.org/10.1016/j.sigpro.2016.12.008

Kumar, S., & Wollinger, T. (2006). Fundamentals of symmetric cryptography. In K. Lemke, C. Paar, & M. Wolf (Eds.), *Embedded security in cars: Securing current and future automotive IT applications* (pp. 125–143). Springer.

Kumar, T., & Namasudra, S. (2023). Introduction to DNA computing. In *Advances in Computers* (Vol. 129, pp. 1–38). Elsevier. https://doi.org/10.1016/bs.adcom.2022.08.001

Mohamed, K., Pauzi, M. N. M., Ali, F. H. H. M., & Ariffin, S. (2022, October). Analysis on avalanche effect in cryptography algorithm. *International Conference on Sustainable Practices, Development and Urbanisation*, 610–618. https://doi.org/10.15405/epms.2022.10.57

OleiwiTuama, S., Kadum, S. A., & Hussein, Z. (2021, December). Text encryption approach using DNA computation and hyperchaotic system. *2021 2nd Information Technology to Enhance e-Learning and Other Applications (IT-ELA)*, 100–105. IEEE.

Şatir, E., & Kendirli, O. (2022). A symmetric DNA encryption process with a biotechnical hardware. *Journal of King Saud University - Science, 34*(3), 101838. https://doi.org/10.1016/j.jksus.2022.101838

Sproot, J. C. (2010). *Elegant chaos: Algebraically simple chaotic flows*. World Scientific.

Thabit, F., Alhomdy, S., & Jagtap, S. (2021). A new data security algorithm for the cloud computing based on genetics techniques and logical-mathematical functions. *International Journal of Intelligent Networks, 2*, 18–33. https://doi.org/10.1016/j.ijin.2021.03.001

Wang, L., Wei, X., Zhang, Y., Gao, Y., & Niu, Q. (2023). A double encryption protection algorithm for stem cell bank privacy data based on improved AES and chaotic encryption technology. *PLOS ONE, 18*(10), e0293418. https://doi.org/10.1371/journal.pone.0293418

Wang, Y., Lei, P., Yang, H., & Cao, H. (2015). Security analysis on a color image encryption based on DNA encoding and chaos map. *Computers & Electrical Engineering, 46*, 433–446. https://doi.org/10.1016/j.compeleceng.2015.03.011

Yaghouti Niyat, A., & Moattar, M. H. (2020). Color image encryption based on hybrid chaotic system and DNA sequences. *Multimedia Tools and Applications, 79*(1–2), 1497–1518. https://doi.org/10.1007/s11042-019-08247-z

Zefreh, E. Z. (2020). An image encryption scheme based on a hybrid model of DNA computing, chaotic systems, and hash functions. *Multimedia Tools and Applications, 79*(33–34), 24993–25022. https://doi.org/10.1007/s11042-020-09111-1

Zhang, Y., & Tang, Y. (2018). A plaintext-related image encryption algorithm based on chaos. *Multimedia Tools and Applications, 77*(6), 6647–6669. https://doi.org/10.1007/s11042-017-4577-1.