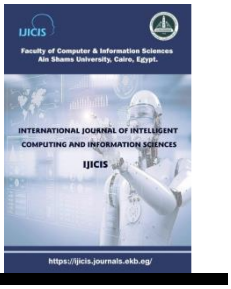
IJICIS, Vol.23, No.4, 112-127
DOI: 10.21608/ijicis.2023.227726.1287

International Journal of Intelligent Computing and Information Sciences

https://ijicis.journals.ekb.eg/


IJICIS
Faculty of Computer & Information Sciences Ain Shams University, Cairo, Egypt.
INTERNATIONAL JOURNAL OF INTELLIGENT COMPUTING AND INFORMATION SCIENCES
IJICIS
https://ijicis.journals.ekb.eg/

SILENT MUTATION FOR GRAY-SCALE IMAGE ENCRYPTION

Nada H. Sharkawy*

Business Information Systems Department,
Higher Institute for Computers & Information Technology,
ElShorouk, Cairo, Egypt
n.hossam@sha.edu.eg

Yasmine M. Afify

Information Systems Department,
Faculty of Computer and Information Sciences, Ain Shams University,
Cairo, Egypt
yasmine.afify@cis.asu.edu.eg

Walaa Gad

Information Systems Department,
Faculty of Computer and Information Sciences, Ain Shams University,
Cairo, Egypt
walaagad@cis.asu.edu.eg

Nagwa Badr

Information Systems Department,
Faculty of Computer and Information Sciences, Ain Shams University,
Cairo, Egypt
nagwabadr@cis.asu.edu.eg

Received 2023-08-07; Revised 2023-11-27; Accepted 2023-12-10

Abstract: *Data embedding into DNA is a conventional encryption technology since it gives the system security and consumes less energy. This work proposes a model that uses the silent mutation method to apply data embedding into DNA within an encryption model. To produce the key, hash functions are applied first to the original image and its metadata followed by the Arnold Transform technique. The key is then sent to LLCS, which uses it to produce a series of coding rules, and transmitted to HCS, which uses it to generate three chaotic sequences. Five chaotic maps are then used to diffuse the output image of the Arnold Transform. DNA operations are applied to the final image using the chaotic HCS sequences followed by the coding rules sequence. After the image has been decoded and silently mutated into a gene sequence, the altered gene is then obtained. Seven evaluation measures are used on ten grayscale images incorporated with two DNA sequences to assess the proposed model's performance. Results indicate a positive improvement in performance as a result of the model's increased security.*

Keywords: *DNA, Data Embedding, Image Encryption, Least Significant Base Substitution (LSBase), Silent Mutation*

***Corresponding Author**: Nada H. Sharkawy

Business Information Systems Department, Higher Institute for Computers & Information Technology, ElShorouk, Cairo, Egypt

Email address: n.hossam@sha.edu.eg

1.Introduction

Due to its high information density storage, energy efficiency, and parallelization ability, Deoxyribonucleic Acid (DNA) is being used in cryptography [1]. Data embedding into DNA has thus become increasingly popular in cryptography systems. It has been quite helpful for both safeguarding the data so that it can be perfectly retrieved and creating new cipher data that cannot be simply cracked.

Data embedding into DNA is a frequent technique used in cryptography. It has been used in a variety of encryption schemes, including those described in [2–4], which provides the system with a huge amount of information and a lot of parallelisms while using less power. However, in addition to integrating data embedding into DNA with an encryption model, there is a lack of applying data embedding into DNA to image data.

In this study, we overcome this issue by incorporating the model from [5] with data embedding into DNA. The suggested methodology involves DNA encoding the image using a variety of coding rules, followed by silent mutation to embed the image into a real DNA sequence.

Seven evaluation measures are used on ten grayscale images incorporated with two DNA sequences to assess the model's performance. Sequence outcomes, nucleotide density, codon composition, pairwise alignment, comparison to other models, histogram analysis, and computational complexity analysis are the metrics covered by the review. The proposed model's experimental results show a positive progression. The nucleotide density analysis suggests that the model preserves the gene's structure, which is also supported by the codon composition. The pairwise alignment demonstrates that the protein sequence is unaffected by the model. The model has a high capacity, preserves the functionality of the sequence, and is blind when compared to other models. This work contribution is as follows:

- Proposing a model constructed by merging the silent mutation technique with the image encryption model.
- Increasing the sensitivity and widening the key space while simultaneously improving the security of the encryption layers.
- Conducting a comprehensive evaluation of the proposed model using seven evaluation metrics on ten popular images and two gene sequences.

The structure of the paper goes as follows. Section 2 presents the relevant data embedding models and differences between them. DNA sequencing, DNA encoding, and the silent mutation approaches are all described in Section 3. The proposed model is described in Section 4. The performance metrics and findings are reported in section 5. Section 6 presents the work's conclusion as well as its future directions.

2.Related Work

There are various methods for data embedding into DNA. One of the methods is substitution method, which depends on replacing the nucleotides of the DNA sequence with the hidden data. First, the data is converted to DNA sequence. Then, the replacement in the real DNA sequence is applied. This method is applied on short DNA sequence but converts it into fake DNA sequence. Another method is silent mutation, which preserves the functionality of the real DNA sequence, and is applied on long DNA sequences. This section represents the previous works which applied substitution and silent mutation in detail.

2.1 Substitution Method

A model [6] proposed in 2015 by El-Sayed et al., is based on double substitution. First, two random DNA sequences are selected, and the data is converted into DNA sequence. Then, the data is substituted using XOR operation with the first random DNA sequence and the first random sequence's ID is appended at the beginning of the sequence. Finally, the resulted sequence is substituted with the second random sequence using a hiding and recovery table, and the second random sequence's ID is appended at the beginning of the final DNA sequence.

Another model [7] was introduced by Pujari et al. in 2017 based on genetic algorithm and substitution method. First, the image is converted into binary image. Next, the image is converted into binary bits which are scrambled using genetic algorithm. Finally, a random DNA sequence is divided into 8 subsequences, where the scrambled image is substituted using XOR operation with the subsequences.

In 2021, Sarosh et al. [8] proposed a model based on Rivest Cipher 6 (RC6) and substitution methods. First, RC6 is applied on the image in Cyclic Block Chaining (CBC) mode. Then, the image is converted to multiple shares using CSIS. Next, the key is generated for each block of shares, where (k-1) keys and the image shares can be public, and the rest of the key shares can be secured by DNA substitution.

2.2 Silent Mutation Method

Jiao et al. proposed a model [2] in 2009 based on silent mutation. The model embeds a text into a Bacillus Subtilis gene (tatAD). It uses only the codons where the four least significant permuted codons code for one amino acid. Therefore, it uses only 32 codons in data embedding. First, the text is binarized using ASCII table. Then, a location is determined on the gene for embedding, where the start codon and end codon numbers are binarized. Next, the binary values for start and end are localized in the first six possible codons on the sequence where it codes for A if the two bits are 00, C if 01, G if 10, and T if 11. Finally, the binary message is localized on the gene with the same rules.

Another model [3] is introduced by Khalifa et al. in 2015 based on Least Significant Base Substitution (LSBase). The model embeds binary bits into DNA or RNA sequences. It applies coding scheme table for the lease significant base of codons. It neglects four codons: *AUG*, *AUA*, *UGG*, and *UGA*. First, the DNA sequence is converted to mRNA. Next, a secret key is used to shuffle the message bits. Finally, the mRNA is divided into codon triplets, LBS substitution is applied, and the sequence is converted back to DNA.

Model [4] is applied in 2018 by Hamad et al. based on codon postfix technique. The model embeds binary bits in a real DNA sequence. It classifies the postfix to two groups: {A or C} coding for 0 and {G or T} coding for 1. It also neglects three codons: *ATG*, *TGG*, and *TGA*. First, a secret key is used to generate a random watermark-bits. Then, these bits are embedded into the DNA sequence using postfix technique.

3.Fundamental Knowledge (Preliminaries)

A detailed description of the DNA sequences, DNA encoding, and the silent mutation methods are described in the following subsections.

3.1 DNA Sequence

DNA - Deoxyribonucleic acid is a biological macromolecule that contains the genetic information that is passed down from one generation to another in living beings [9]. It is constructed out of two strands of phosphodiester-bonded nucleic acid bases. Hydrogen bonds hold these strands together, creating the helix structure. Adenine A, Thymine T, Cytosine C, and Guanine G are the four different types of bases. A and T are purines as a result of bonding together with two hydrogen bonds. C and G are connected by three hydrogen bonds; therefore, they are pyrimidines [10]. Due to its high information density storage, energy efficiency, and parallelization ability, DNA is now employed in cryptography [1].

3.2 DNA Encoding

Since every single base can be represented by a pair of bits in DNA, data can be encoded into the sequence through its binary representation. Based on the complementary model, purines, and pyrimidines proposed by Watson and Crick, there are eight relevant DNA rules that can be obtained. These rules are listed in Table 1 [10].

The image is converted into a DNA sequence of four bases using a DNA encoding process. For example, the binary representation of pixel x with value equal to 125 is "01111101". The DNA-encoded sequence is "GAAG" when the pixel's chosen coding rule is {3}. Each pixel is encoded using the dynamic DNA coding approach as a four-base DNA sequence. DNA is used to encode each pair of two bits in a pixel using certain coding principles. [11]. The DNA-encoded sequence in the same example is "GAGA" with the chosen pixel coding rules are {3, 4, 5, 8}. Based on the set coding rules, the decoding procedure will be conducted.

Table 1 Rules for DNA-Encoding

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

3.3 Silent Mutation Method

In transcription process, DNA sequence is read in triplet bases called codons, where each codon codes for one amino acid. Table 2 represents the codons and their corresponding amino acid [2]. As noticed in the table, there are some codons codes for the same amino acid. For example, {TCT, TCC, TCA, TCG} code for Serine “Ser” amino acid. Therefore, any change in the third nucleotide does not affect the corresponding amino acid, which is called Silent Mutation. Silent mutations can happen within an exon or in a non-coding region (such outside of a gene or inside of an intron) [3].

This can also be applied to codons as {CAT, CAC}, which code for Histidine “His”, and {CAA, CAG}, which code for Glutamine “Gln”. But six codons -highlighted in the table- cannot be used, as {ATG} that codes for Methionine “Met”, the start codon. Also, the three stop codons {TAA, TAG, TGA} are not used. In addition to the complementary to ATG and TGA, which are {ATA, TGG}, which code for Isoleucine “Ile” and Tryptophan “Trp”, respectively.

Table 2 Codons and their Corresponding Amino acids

		Second Position					
		T	C	A	G		
First Position	T	TTT Phe [F] TTC Phe [F] TTA Leu [L] TTG Leu [L]	TCT Ser [S] TCC Ser [S] TCA Ser [S] TCG Ser [S]	TAT Tyr [Y] TAC Tyr [Y] TAA Ter [end] TAG Ter [end]	TGT Cys [C] TGC Cys [C] TGA Ter [end] TGG Trp [W]	T C A G	Third Position
	C	CTT Leu [L] CTC Leu [L] CTA Leu [L] CTG Leu [L]	CCT Pro [P] CCC Pro [P] CCA Pro [P] CCG Pro [P]	CAT His [H] CAC His [H] CAA Gln [Q] CAG Gln [Q]	CGT Arg [R] CGC Arg [R] CGA Arg [R] CGG Arg [R]	T C A G	
	A	ATT Ile [I] ATC Ile [I] ATA Ile [I] ATG Met [M]	ACT Thr [T] ACC Thr [T] ACA Thr [T] ACG Thr [T]	AAT Asn [N] AAC Asn [N] AAA Lys [K] AAG Lys [K]	AGT Ser [S] AGC Ser [S] AGA Arg [R] AGG Arg [R]	T C A G	
	G	GTT Val [V] GTC Val [V] GTA Val [V] GTG Val [V]	GCT Ala [A] GCC Ala [A] GCA Ala [A] GCG Ala [A]	GAT Asp [D] GAC Asp [D] GAA Glu [E] GAG Glu [E]	GGT Gly[G] GGC Gly[G] GGA Gly[G] GGG Gly[G]	T C A G	

4.Proposed Model

The proposed model, its elements, and the encryption and decryption techniques are all thoroughly explained in this section. Modifications to the model from [5] are described in the following subsections.

4.1 Proposed Encryption Model

Figure. 1 provides a description of the suggested encryption model used on the M*N grayscale image. It entails the following actions:

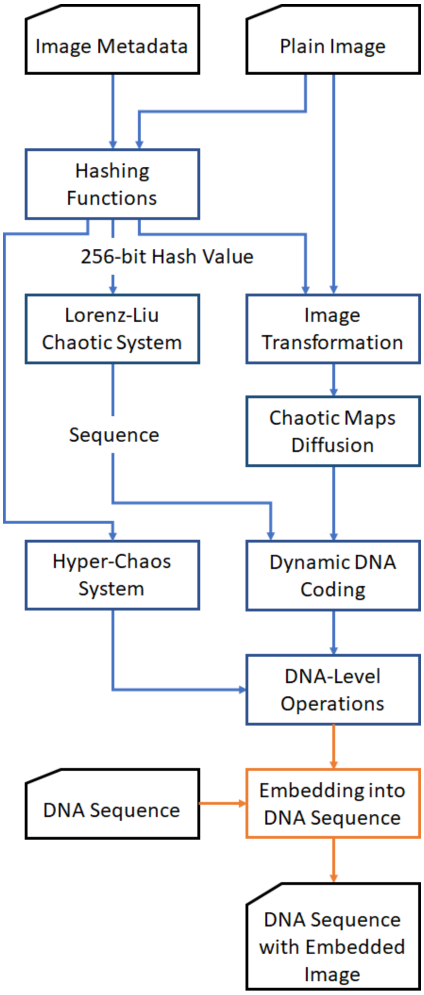


Figure. 1 Proposed Encryption Model

Step 1: Creation of 256-bit Hash Value. The original image and its metadata are subjected to the MD5 hash function, and the produced 128-bit hash values are concatenated and passed to SHA-256, which generates the 256-bit hash value H.

$$H = [h_1, h_2, \ldots, h_{64}]$$

Step 2: Creation of Initial Values and Parameters. There are four parts splits from the image. The Arnold Transform parameters a, b, and c, the Hyper-Chaos system initial values w_0, x_0, y_0, and z_0, the Liu chaotic system parameters r_2, s_2, t_2, u, and v, and the Lorenz-Liu chaotic system parameters r1, s1, t1, and λ are obtained by applying the average of each part and H to the equations in model [10]. The starting values of LCS and LLCS are then determined together using the original model [5] equations.
Step 3: Using Arnold Transform, Transforming an Image. Arnold Transform is used to create a scrambled image I_1 using the original image.
Step 4: Using Lorenz-Liu Chaotic System, obtaining Coding Rules. Using the parameters and initial values, LLCS is iterated, resulting in the three sequences L_1, L_2, and L_3. Equations from the model [5] are utilized in L_1 to create coding rules for a chaotic sequence CR.
Step 5: Creation of Hyper-Chaos Sequences. Memristor HCS is used to create four chaotic sequences, W, X, Y, and Z. These sequences are then applied to the equations in model [10], producing three M*N sequences, Y_2, Z_2, and W_2.
Step 6: Employing Chaotic Maps for Encryption. Tent, Logistic, Piecewise, Gauss, and Henon maps are the order in which five chaotic map sequences of five XOR operations are applied to I_1, resulting in I_2.
Step 7: Appling Dynamic DNA Coding to Image. The 4*M*N DNA sequences I_2, W_2, Y_2, and Z_2 are binarized and DNA-encoded using the CR sequence to establish the coding rule for each pair of bits in the sequences.
Step 8: DNA operations application. I_2 and Y_2 are used in an XOR operation to produce I_3. Subsequently, I_3 is multiplied by Z_2 to get I_4, which is in turn sorted according to W_2 to produce I_5.
Step 9: Creating the modified DNA sequence. I_5 is DNA-decoded using CR to binary sequence I_6. Subsequently, the chosen gene sequence is read, then each codon is checked if it is not one of the unused codons. Next, based on I_6, the codon is altered using rules in Table 3, generating the final gene sequence.

Table 3 Nucleotide Altering Rules

Nucleotide	Bit = 0	Bit = 1
A	A	G
G	A	G
C	C	T
T	C	T

4.2 Proposed Decryption Model

The opposing steps used in the encryption model are precisely the core of the decryption model . The chaotic sequences W, Y, and Z are produced when the HCS is applied to w_0, x_0, y_0, and z_0. Applying LLCS to the following variables results in the CR sequence: l_1, l_2, l_3, m_1, m_2, m_3, r_1, r_2, s_1, s_2, t_1, t_2, u, v, and λ. The gene sequence, the altered gene sequence, and the altering rules are incorporated to produce C. Then, C is DNA-encoded utilizing CR together with W, Y, and Z. I_5, W_2, Y_2, and Z_2 are the generated DNA sequences that are given to the inverse DNA operations: I_5 is inverse sorted using W_2 to get I_4, Z_2 is then subtracted from I_4 to produce I_3, which is then given Y_2 to perform an XOR operation to produce I_2. I_2 is then decoded from DNA. The Arnold Transform is then used to obtain the original image I after applying XOR operations on I_2 and the inverse series of chaotic maps.

5 Experimental Evaluation

The model is implemented with Intel® Core™ i7-4500U CPU @ 1.80 GHz processor and 8 GB RAM in MATLAB R2021b platform on 64-bit machine and Windows 10 Operating System. The following subsections describe the dataset, the evaluation metrics, and the experimental evaluation results in details.

5.1 Dataset

The model is applied on two DNA sequence genes, downloaded from National Center for Biotechnology Information (NCBI) [12]. The first is dystrophin gene (DMD), found in Homo sapiens with length greater than 2 million bases. The encoded protein from DMD forms a component of the dystrophin-glycoprotein complex (DGC), which bridges the inner cytoskeleton and the extracellular matrix. The second is contactin associated protein 2 gene (CNTNAP2), found in Homo sapiens with length greater than 2 million bases. CNTNAP2 encodes a member of the neurexin family which functions in the vertebrate nervous system as cell adhesion molecules and receptors. To assess the proposed model, these two genes are applied on an image dataset, which contains popular ten $256*256$ gray-scale images: Baboon, Cameraman, House, Peppers, Lena, Barbara, White, QR code, Black, and Couple.

5.2 Evaluation Metrics

Seven popular evaluation metrics are used to assess the proposed model, including: sequences results, nucleotide density, codon composition, pairwise alignment, comparison to other models, histogram analysis, and computational complexity analysis.

5.3 Experimental Results

The experimental results are described in details in the following subsections.

5.3.1 Sequences Results

Lena image was embedded into DMD and CNTNAP2 genes, where the results are represented in Table 4. It consists of the sequence name, sequence ID in NCBI, sequence length, the identity percentage of the protein sequence of the updated sequence. In addition, it contains the percentage of special codons that cannot be used in embedding, the actual capacity that can be used in the sequence measured in Kilobytes, and the percentage of the changed codons in the sequence after embedding.

Table 4 Results of Embedding Lena Image into DNA Sequences

Seq Name	Seq ID	Seq Length	Identities	Special Codons	Actual Capacity	Changed Codons
DMD	NC_000023.11	2,220,165 bp	100%	12.29%	79.24 KB	43.68%
CNTNAP2	NC_000007.14	2,304,198 bp	100%	11.99%	82.52 KB	43.74%

5.3.2 *Nucleotide Density Analysis*

It expresses the change in the density of nucleotide types after image embedding into DNA sequence. represents the nucleotide density of the sequence before and after embedding. It shows that there is a slight change in the density of the nucleotides after embedding which implies that the model conserves the structure of the gene.

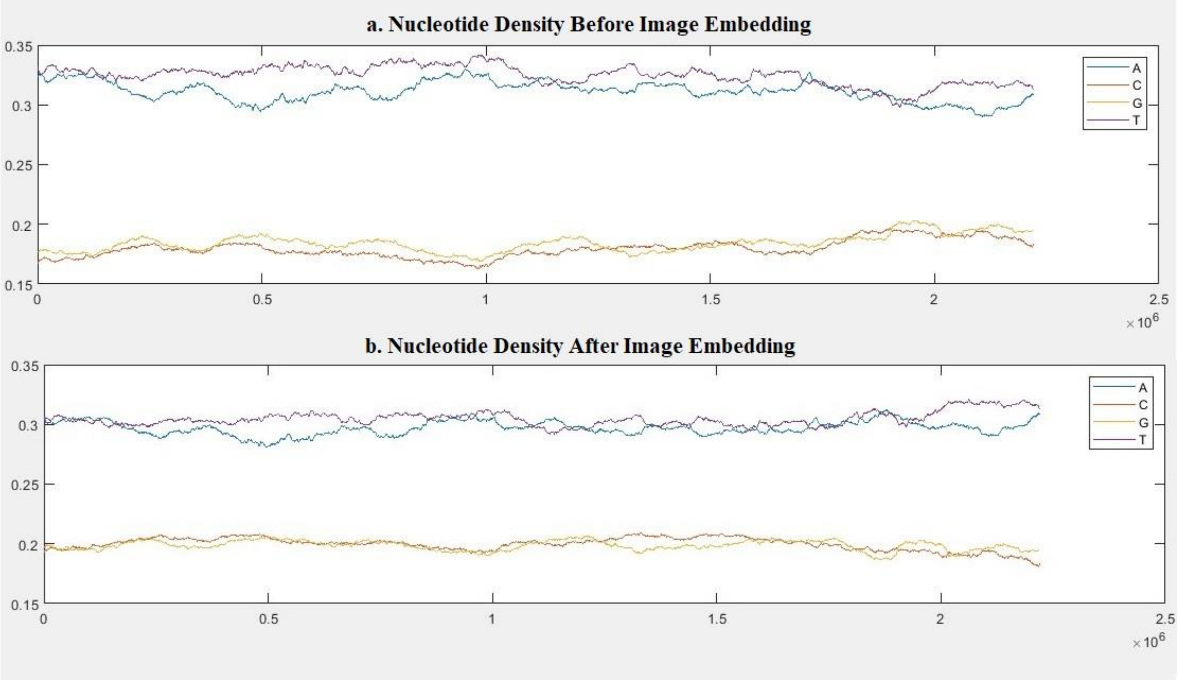


Figure. 2: Nucleotide Density of the DNA sequence Before and After Image Embedding

5.3.3 *Codon Composition Analysis*

It describes the change in the codon composition in the DNA sequence after image embedding. Figure. 3 shows the difference in the codon composition before and after embedding. It shows the change in the codon composition and the effect of the image embedding on the DNA sequence.

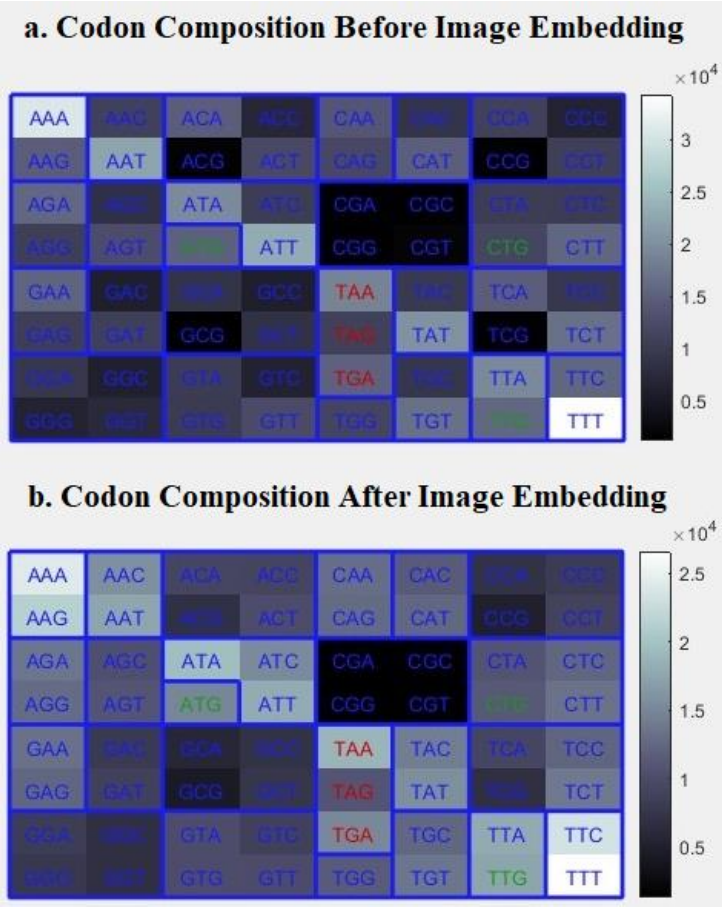


Figure. 3: Codon Composition of the DNA sequence Before and After Image Embedding

5.3.4 *Pairwise Alignment Analysis*

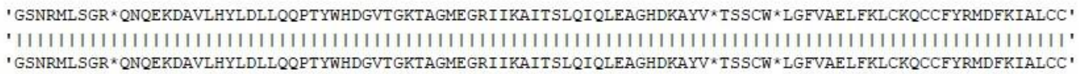
It represents the pairwise alignment of the protein sequences generated from DNA sequence before and after image embedding. Figure. 4 represents the result of the alignment. It shows that there is no change in the amino acids in the protein which implies that the model does not affect the protein sequence.

5.3.5 *Comparison to other Models*

The proposed model is compared to other models in Table 5. The table consists of the model name, the method applied in each model, the type of the embedded data, the size of the data, the capacity of the model, functionality conserved, and blindness of the model. The capacity indicates the maximum capacity of the embedding. It is calculated using Eq. (1):

$$capacity = (size\ of\ data\ in\ bits)/(size\ of\ cover\ in\ bases) = (1/3 * |S|)/|S| = 1/3 \tag{1}$$

Where S is the number of nucleotides that can be covered in the sequence.



```
'GSNRMLSGR*QNQEKDAVLHYLDLLQQPTYWHDGVTGKTAGMEGRIIKAITSLQIQLEAGHDKAYV*TSSCW*LGFVAELFKLCKQCCFYRMDFKIALCC'
'||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||'
'GSNRMLSGR*QNQEKDAVLHYLDLLQQPTYWHDGVTGKTAGMEGRIIKAITSLQIQLEAGHDKAYV*TSSCW*LGFVAELFKLCKQCCFYRMDFKIALCC'
```

Figure. 4: Pairwise Alignment Analysis of the DNA sequence Before and After Image Embedding

The functionality of the model is conserved when the protein sequences generated from DNA sequence before and after embedding are identical. The blindness of the model is that the embedded data can be retrieved without any reference to the DNA sequence. The table 5 shows that the proposed model is applied to image data with extremely higher size than other models. The model implies high capacity, conserves the sequence functionality, and is blind.

Table 5 Proposed Model Comparison with other Models

Model	Method	Data Type	Data Size	Capacity	Functionality Conserved	Blind
[4]	Silent Mutation	Binary Message	----------	0.333	----------	----------
[3]	Silent Mutation	Binary Message	3.7 Kilobytes	0.333	Yes	Yes
[2]	Silent Mutation	Text	----------	---------	----------	----------
[7]	Substitution	Image	64 bytes	1	No	Yes
[6]	Substitution	Text	Max 27 Kilobytes	1	No	Yes
Proposed Model	***Silent Mutation***	***Image***	***64 Kilobytes***	***0.333***	***Yes***	***Yes***

5.3.6 Histogram Analysis

It displays the distribution of pixel intensity values; hence, the histogram's identical distribution suggests a significant restoration of the original image.

Figure. 5

Figure. *14* display the histogram's results for the ten images.

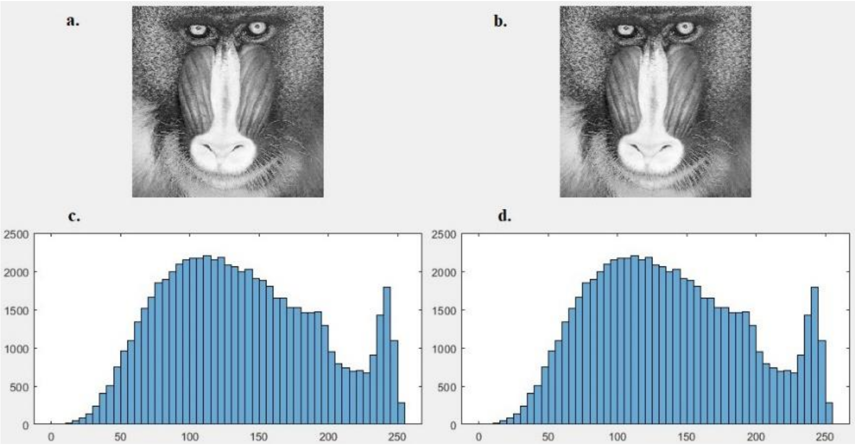


Figure. 5: Histogram of Baboon image. a. original image, b. decrypted image, c. histogram of original image, d. histogram of decrypted image

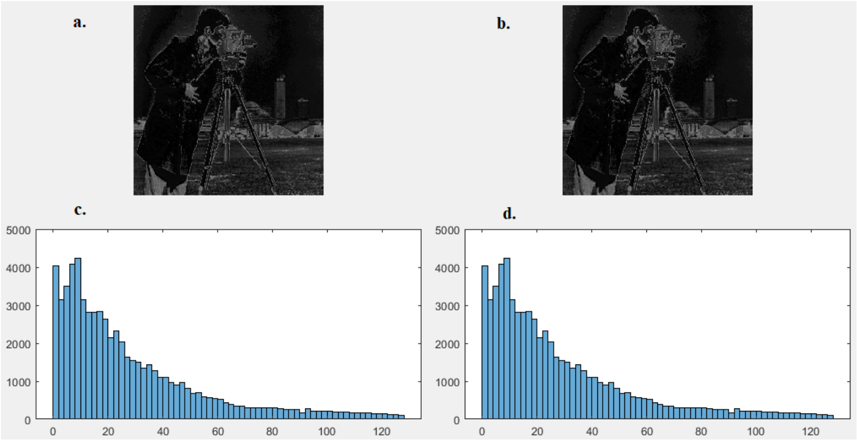


Figure. 6: Histogram of Cameraman image. a. original image, b. decrypted image, c. histogram of original image, d. histogram of decrypted image

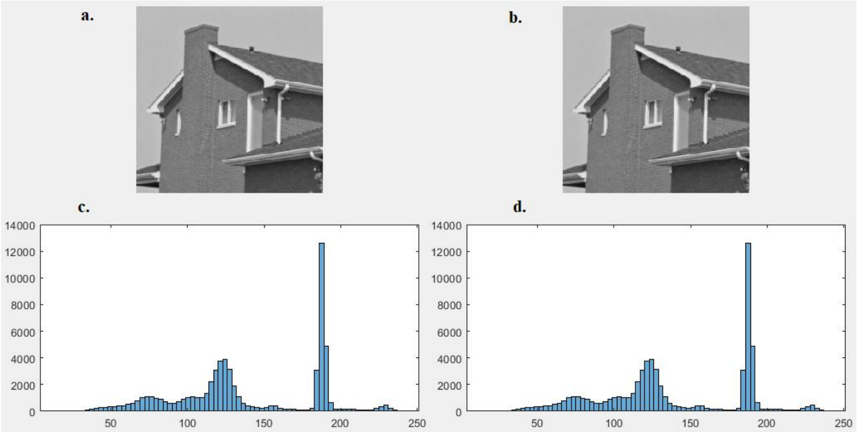


Figure. 7: Histogram of House image. a. original image, b. decrypted image, c. histogram of original image, d. histogram of decrypted image

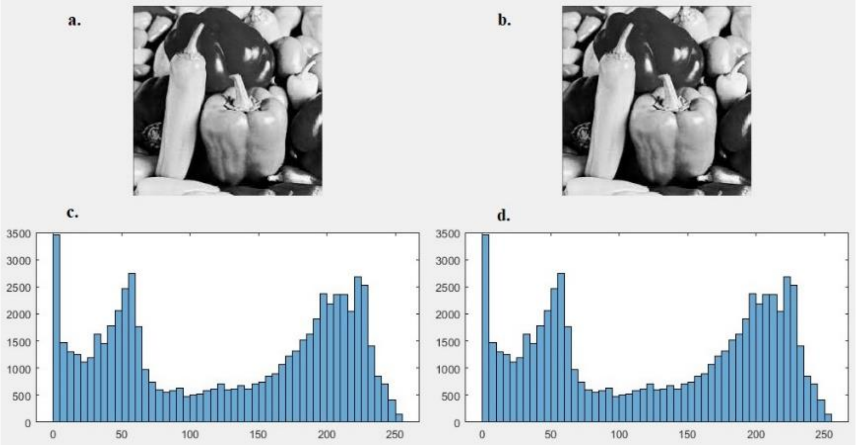


Figure. 8: Histogram of Peppers image. a. original image, b. decrypted image, c. histogram of original image, d. histogram of decrypted image

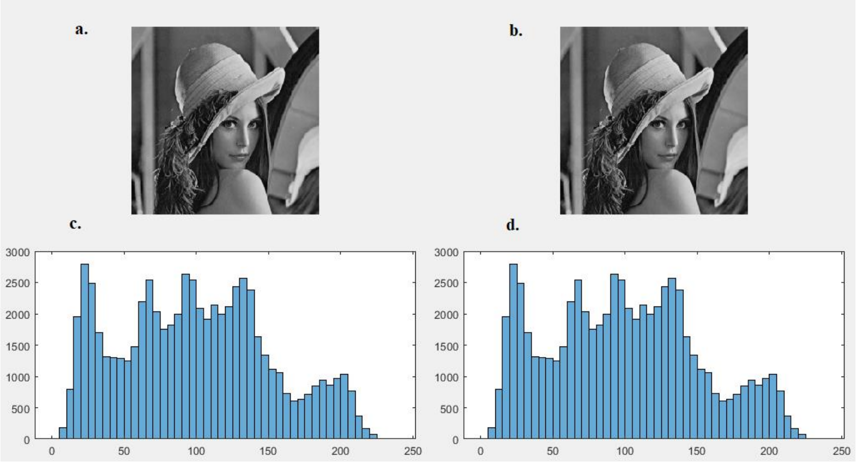


Figure. 9: Histogram of Lena image. a. original image, b. decrypted image, c. histogram of original image, d. histogram of decrypted image

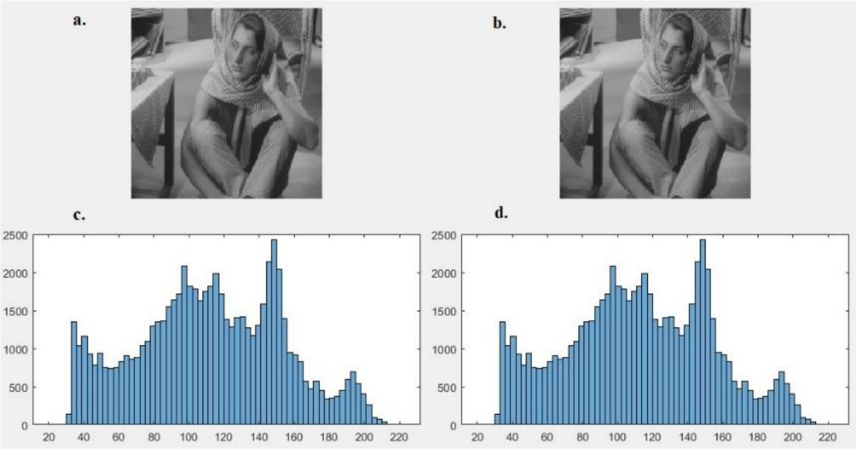


Figure. 10: Histogram of Barbara image. a. original image, b. decrypted image, c. histogram of original image, d. histogram of decrypted image

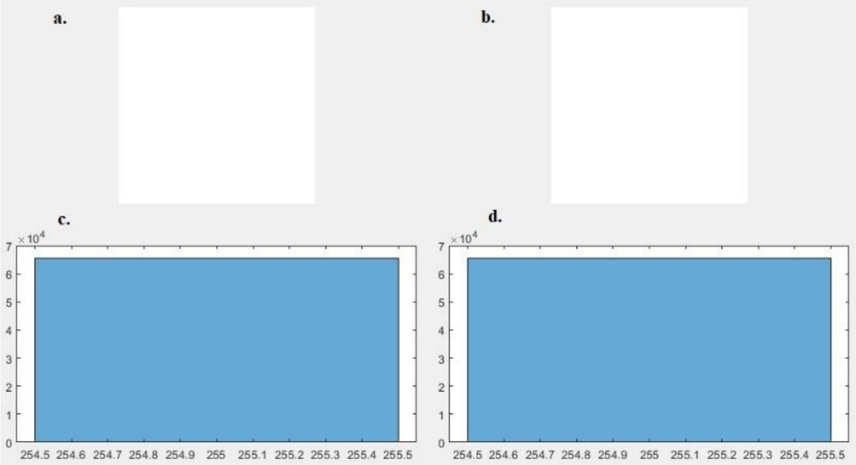


Figure. 11: Histogram of White image. a. original image, b. decrypted image, c. histogram of original image, d. histogram of decrypted image

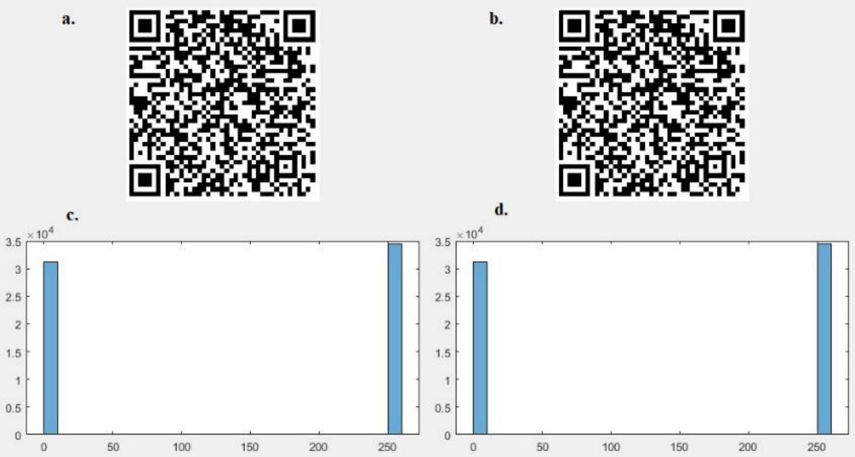


Figure. 12: Histogram of QR Code image. a. original image, b. decrypted image, c. histogram of original image, d. histogram of decrypted image

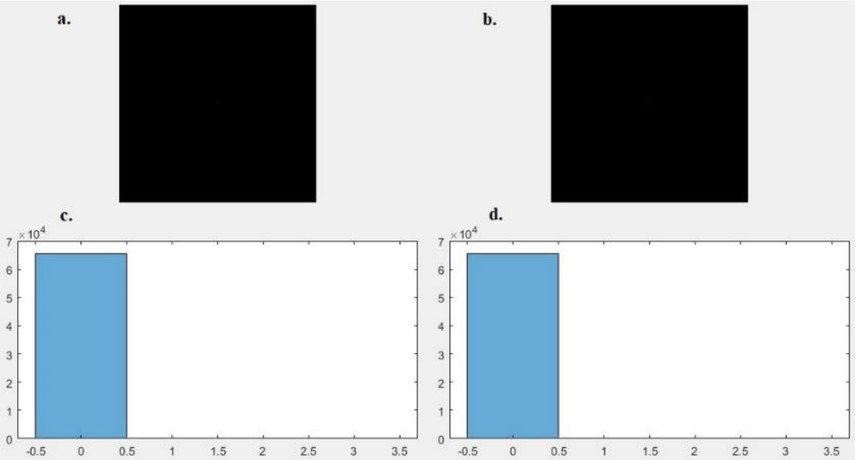


Figure. 13: Histogram of Black image. a. original image, b. decrypted image, c. histogram of original image, d. histogram of decrypted image

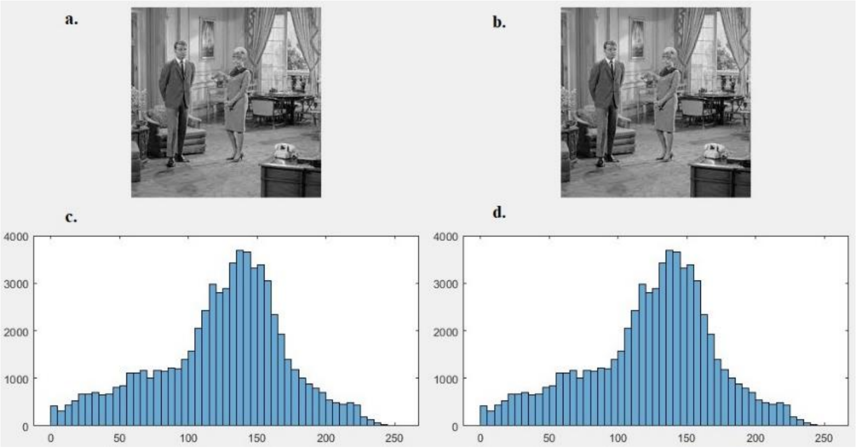


Figure. 14: Histogram of Couple image. a. original image, b. decrypted image, c. histogram of original image, d. histogram of decrypted image

5.3.7 Computational Complexity Analysis

It represents the computational complexity of the steps applied in the encryption model [13]. The computational complexity of the main model is O(M∗N) [5]. The computational complexity of data embedding is O(M∗N). So, the computational complexity of the proposed model is O(M∗N + M∗N) ≈ O(M∗N), which is linear and depends on the original image size.

6 Conclusion

Data embedding into DNA is a common cryptographic approach. It uses less power and provides the system security. This paper applies data embedding into DNA to an encryption model. First, the original image and its metadata are provided to hash functions to generate the key. The image is provided to the Arnold Transform method. After that, the key is sent to HCS, which employs it to create three chaotic sequences, and it is given to LLCS, which used to create a sequence of coding rules. The Arnold Transform's output image is then spread out using five chaotic maps. The resultant image is then generated with DNA operations using the chaotic sequences produced by HCS, and the resulting image is DNA-encoded using the coding rules sequence. The changed gene is then obtained after the image has been decoded and inserted into a gene sequence using silent mutation.

Ten grayscale images combined with two DNA sequences are employed to assess the model's efficiency. Seven popular evaluation metrics are used for the full assessment, including metrics for sequence results, nucleotide density, codon composition, pairwise alignment, comparison to other models, histogram analysis, and computational complexity analysis. Experimental findings confirm the proposed model's suggested progression. The nucleotide density analysis and codon composition both support the notion that the model preserves the structure of the gene. The protein sequence is not impacted by the model, as shown by the pairwise alignment. In comparison to other models, the model is blind, keeps the functionality of the sequence, and has a high capacity. Our next step is to in vitro embed the cipher image into a DNA sequence.

References

1. Sanober A, Anwar S (2022) Crytographical primitive for blockchain: a secure random DNA encoded key generation technique. Multimed Tools Appl. https://doi.org/10.1007/s11042-022-13063-z
2. Jiao S-H, Goutte R (2009) Hiding data in DNA of living organisms. Nat Sci (Irvine) 01:181–184
3. Khalifa A, Hamad S (2015) Hiding Secret Information in DNA Sequences Using Silent Mutations. British Journal of Mathematics & Computer Science 11:1–11
4. Hamad S, Elhadad A, Khalifa A (2018) DNA Watermarking using codon postfix technique. IEEE/ACM Trans Comput Biol Bioinform 15:1605–1610
5. Afify YM, Sharkawy NH, Gad W, Badr N (2023) A new dynamic DNA-coding model for gray-scale image encryption. Complex & Intelligent Systems. https://doi.org/10.1007/s40747-023-01187-0
6. El-Sayed F, Abd HA, Moussa MI, #1 FEI, Abdalkader HM, Moussa MI (2015) Enhancing the Security of Data Hiding Using Double DNA Sequences Evaluating the Effects of Symmetric Cryptography Algorithms on Power Consumption for Different Data Types View project Enhancing the Security of Data Hiding Using Double DNA Sequences.
7. Pujari SK, Bhattacharjee G, Bhoi S (2018) A Hybridized Model for Image Encryption through Genetic Algorithm and DNA Sequence. In: Procedia Comput Sci. Elsevier B.V., pp 165–171
8. Sarosh P, Parah SA, Bhat GM, Heidari AA, Muhammad K (2021) Secret Sharing-based Personal Health Records Management for the Internet of Health Things. Sustain Cities Soc. https://doi.org/10.1016/j.scs.2021.103129
9. De Dieu NJ, Ruben FSV, Nestor T, Zeric NT, Jacques K (2022) Dynamic analysis of a novel chaotic system with no linear terms and use for DNA-based image encryption. Multimed Tools Appl 81:10907–10934
10. Sharkawy NH, Afify YM, Gad W, Badr N (2022) Gray-Scale Image Encryption Using DNA Operations. IEEE Access 10:63004–63019
11. Shen Y, Zou T, Zhang L, Wu Z, Su Y, Yan F (2022) A novel solar radio spectrogram encryption algorithm based on parameter variable chaotic systems and DNA dynamic encoding. Phys Scr. https://doi.org/10.1088/1402-4896/ac65bf
12. National Center for Biotechnology Information (NCBI)[Internet]. Bethesda (MD): National Library of Medicine (US), National Center for Biotechnology Information. https://www.ncbi.nlm.nih.gov/. Accessed 6 Jun 2023
13. Aouissaoui I, Bakir T, Sakly A (2021) Robustly correlated key-medical image for DNA-chaos based encryption. IET Image Process 15:2770–2786