A DNA Cryptographic Solution for Secured Image and Text Encryption

Bahubali Akiwate[1]
Department of Computer Science and Engineering
KLE College of Engineering and Technology, Chikodi, India

Latha Parthiban[2]
Department of Computer Science and Engineering
Pondicherry University, Pondicherry, India

***Abstract*—In recent days, DNA cryptography is gaining more popularity for providing better security to image and text data. This paper presents a DNA based cryptographic solution for image and textual information. Image encryption involves scrambling at pixel and bit levels based on hyperchaotic sequences. Both image and text encryption involves basic DNA encoding rules, key combination, and conversion of data into binary and other forms. This new DNA cryptographic approach adds more dynamicity and randomness, making the cipher and keys harder to break. The proposed image encryption technique presents better results for various parameters, like Image Histogram, Correlation co-efficient, Information Entropy, Number of Pixels Change Rate (NPCR), and Unified Average Changing Intensity (UACI), Key Space, and Sensitivity compared with existing approaches. Improved time and space complexity, random key generation for text encryption prove that DNA cryptography can be a better security solution for new applications.**

***Keywords*—DNA cryptography; image encryption; text encryption; DNA digital coding; DNA sequences**

I. Introduction

Security is often a crucial necessity for sensitive data transmission over communication networks. Various security techniques used to provide information privacy bring benefits to an organization or individual businesses [1]. There exist many benchmarks symmetric and asymmetric cryptographic algorithms such as Advanced Encryption Standard (AES), IDEA (International Data Encryption Algorithm), and RSA (Proposed by Rivest, Shamir, and Adleman) to provide security to text data. But the survey provides evidence that these algorithms are not suitable for image encryption [2, 3, 10, 24]. Image data characteristics like pixel correlation, bulk space, and high redundancy among pixel values make image encryption more challenging compared to text encryption [3,10]. Image Encryption plays a vital role in secured multimedia communication but the existing symmetric and asymmetric algorithms suffer from side-channel attack, Brute Force attack, Differential attack, and other statistical attacks [4]. There is a marked lack of better image cryptographic system. The proposed DNA based image Cryptosystem makes use of chaotic sequences to overcome existing limitations of symmetric and asymmetric cryptographic systems along with its confusion and diffusion properties [3]. To bring dynamicity, better storage, and time complexity, high parallelism, and low power consumption Adleman introduced DNA computing in 1994, which makes DNA cryptography the right choice for today's Internet applications [5]. DNA computing is still an area of interest for many researchers for its massively parallel processing capabilities and high resistance to brute force attacks [6]. The existing image encryption standards and mathematical models combined with DNA cryptography show defects in terms of CPU time, memory consumption, and battery usage [11]. The proposed DNA based approach for image encryption employs a chaotic sequence, which is deterministic and can produce a non-linear sequence [7]. It brings the advantages of unpredictability, pseudo randomness, and extremely sensitive to system control parameters and initial values [4, 7, 10, 13, 24, 27]. Also, Chaos systems can eventually return to the original state from the proceeded state [8-10]. The proposed approach involves a sequence of steps, such as the use of five-dimensional hyperchaotic sequences that produce a strong ciphered image, scrambling at the pixel level and bit level. The analysis of various parameters like Image Histogram, Correlation co-efficient, Information Entropy, NPCR, and UACI, Key Space, and Sensitivity shows that the proposed technique overcomes the limitations of the existing image encryption techniques. This paper also presents DNA based text encryption technique, which is based on the motivation of Kerckhoff's principle, which states that secrecy of transmitted message depends on key during decryption and not on an algorithm for encryption and decryption. At a high level, the algorithm is secure if the cryptanalyst is unable to deduce the key to obtain plaintext from the corresponding ciphertext [26]. This DNA based text encryption method uses the knowledge of random key generation to produce different sequences for the same input to achieve enhanced security performance. The proposed text encryption method outperforms the existing encryption techniques in terms of time and space complexities [11]. In this paper, Section 2 presents a preliminary study of the proposed approach; Section 3 covers image encryption in detail with result analysis. Section 4 discusses text encryption with two cases and time and space requirement analysis.

II. Preliminary Study

Traditional algorithms, including symmetric and asymmetric, are having many drawbacks concerning the exchange or use of a key. Compared to these, DNA cryptography can provide multifold security [45]. It provides an enriched security level [15, 16]. Conventional block cipher algorithms are not suitable for secured multimedia communication over public networks [2, 9, 24, 36, 37]. On the other hand, DNA cryptography is gaining more attention with a variation of chaos-based substitution permutation architecture [8, 16, 29, 30, 34, 36, 37]. It can run with lesser memory and reduced computational overhead when compared with other standards like Elliptic Curve Cryptography, Packet wavelet, Fourier transform, Cellular automata, etc. [3, 8, 12]. The chaos method combined with DNA cryptography proved secure against a chosen-plaintext attack and differential attacks based on the previous studies [3, 17]. Boriga et al. proposed a 1D chaotic image encryption map that would be found weaker [23]. As there is a single variable is used, which makes easy prediction of initial values. Fidrich proposed use of 2D chaotic maps proved better diffusion properties. But, the use of a limited key space makes it easy to decode [18, 19]. Chen et al. proposed 3D chaotic maps proved better confusion and diffusion properties [20, 23]. Chen et al. presented a high complex 4D hyperchaotic system that brought many security advantages [38]. However, lower dimensional chaotic systems can be crackable as computer machines having limited precision. Understandably, only a portion of plaintext or ciphertext will help to get the key back [17]. Hence these are weaker against differential attacks [23].

The following are the demerits of previous works identified:

- Low dimensional chaos methods face difficulty in providing high security [30].
- Existing image encryption techniques work well only for a homogeneous image dataset, like medical or satellite images.
- There is no such practically used chaos-based DNA cryptography that exists worldwide in different application areas [4].

A. *Our Contributions*

The following are the major contributions of this proposed work:

- Capable of encrypting and retrieving highly sensitive images those are heterogeneous. The main features of these images are continuity, the large volume of data, and the strong association of adjacent pixels.
- During image encryption, pixel-level and bit-level permutation will render stronger cipher, which is difficult for an attacker to crack.
- The use of a 5D hyperchaotic system can ensure enhanced complexity and hence able to achieve improved security.
- Both image and text encryptions are possible over a single framework with better CPU latency.

III. DNA BASED IMAGE ENCRYPTION

The proposed image encryption technique performs a series of operations like scrambling using a 5D (five-dimensional) hyperchaotic method, XOR operations, and complementary rules to produce a solid ciphered image. [12,36]. It also includes scrambling at pixel and bit levels along with basic encoding rules and decomposition operations [17, 18]. Scrambling at the pixel level can be made according to a predefined concept [35,36]. DNA cryptography with a chaotic system depends on mathematical applications [19]. Edward Lorenz proposed the chaos theory in 1963 for the first time [35]. Confusion and Diffusion are two main properties of a chaotic system. With the confusion property, we can ensure the exchange of image pixel position randomly without affecting actual pixels. Diffusion mainly focuses on substituting one-pixel value with other pixel values by applying some mathematical operations over image pixels. Here only pixels will be permuted. Scrambling at bit level can make more difficulty in breaking the cipher by an attacker inducing reordering at bit levels. The proposed solution is a mixture of all these that can result in increased complexity for processes of encryption and decryption, making it harder for an attacker to crack.

A. *DNA Digital Coding*

DNA has four bases of Deoxyribo Nucleic Acid, namely Adenine (A), Thymine (T), Cytosine (C), and Genuine (G) [14, 19]. All the A and T bases complement each other according to the Watson - Crick Model. The bases C and G are mutually complementary [9,10,16,23]. Table I shows the DNA XOR operation between these bases [16, 21-23, 32].

TABLE I. XOR USE OF DNA SEQUENCES

XOR	**A**	**T**	**C**	**G**
A	A	T	C	G
T	T	A	G	C
C	C	G	A	T
G	G	C	T	A

Where A represents the binary value 00(Decimal Value 0), C represents the binary value 01(Decimal Value1), G represents binary value 10(Decimal Value 2) and T represents the binary value 11(Decimal Value 3) [16]. Every pixel represents a DNA sequence of length 4 in an 8-bit grayscale image. The proposed approach uses complementary rules for every character produced using DNA sequences. The complementary rule says about the base pairs. The bases, Adenine and Thymine can make one pair, and Cytosine and Genuine can make another pair [11-14, 21-23].

Suppose n_i be the set of bases A, T, C and G. Then complementary principle says the base string n_i of the encoding bases as follows:

$$n_i \neq C(n_i) \neq C(C(n_i)) \neq C(C(C(n_i)))$$

$$n_i = C(C(C(C(n_i))))$$

Where n_i and $C(n_i)$ are complementary and are base pairs. According to above statements, six complementary rules are as follows:

A → T,T → C,C → G,G → A

A → T,T → G,G → C,C → A

A → C,C → T,T → G,G → A

A → C,C → G,G → T,T → A

A → G,G → T,T → C,C → A

A → G,G → C,C → T,T → A

TABLE II. DNA ENCODING RULES

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

For each base of DNA sequence in terms of A, T, G, C their encrypted values will not remain same or equal [22].

DNA encoding and decoding operations are needed to map binary sequences into DNA bases and vice versa. Table II shows DNA encoding rules. There are 8 rules which satisfy the Watson–Crick complementary model [17, 19, 21, 24, 36, 37]. Here the selection of DNA bases such as A, T, C, and G can be made by following the DNA encoding rules. We consider two digits of the binary value for the mapping at a time.

Assume the original image P has scale matrix M X N. To bring out-diffusion property for an image P, Scrambling at pixel and bit levels are performed [15-17]. It permutes the bits of an image by considering the pixel values. Also it uses 5-D hyperchaotic system to obtain chaotic sequences as discussed below.

B. Hyperchaotic System

The proposed system uses five-dimensional (5-D) chaotic systems to enhance security and to bring increased complexity for image encryption [18]. It is covered below. A hyperchaotic 5-D system represented as the following equations (1):

$$c_1 = t(c_2 - c_1) + c_2c_3c_4$$

$$c_2 = u(c_1 + \ c_2) + c_5 - c_1c_3c_4$$

$$c_3 = -vc_2 - wc_3 - xc_4 + c_1c_2c_4$$

$$c_4 = -y_4 + c_1c_2c_3$$

$$c_5 = -z(c_1 + c_2) \qquad (1)$$

Where t, u, v, w, x, y, z are system control parameters and c_1, c_2, c_3, c_4 and c_5 are system state variables. There are many existing approaches which used different algorithms to bring security for encryption. Since image data is highly sensitive, the chaotic sequence approach could be better to maintain security as there is the ability to overcome dependency on image pixels by confusion and diffusion properties [19-21].

C. Scrambling at Pixel Level

First, compute the initial values c_1, c_2, c_3, c_4 and c_5 of the 5-D hyper chaotic system as shown in equations (2) below:

$$c_1(1) = mod\left(\sum_{i=1}^{5} c_j^0, 1\right)$$

$$c_i\,(1) = mod\big(c_{i-1}(1) + c_j^0, 1\big) \text{ Where i=2, 3, 4, 5} \qquad (2)$$

Where $c^0{}_1$, $c^0{}_2$, $c^0{}_3$, $c^0{}_4$ and $c^0{}_5$ are initial keys [22].

Producing chaotic series for image encryption might cause a transient effect that is impermanent and can have a sudden change of the state. Scrambling at the pixel and bit levels of an image can cause transient effect by internal or nearby values. So, there is a need for having several cycles to avoid such transient effect over hyperchaotic system N times, as shown in equation (3) below:

$$N = 200 + mod(\Big(\big(\textstyle\sum_{i=1}^{5} c_i^0\big) - \ \big|\sum_{i=1}^{5} c_i^0\big|\Big) \times 10^{15}, 200) \qquad (3)$$

By continuing the cycles or iterations over this chaotic system up to MN times, three chaotic sequences s_1, s_2 and s_3 are obtained.

Suppose the original plain image is P and its positions are (x, y). Let P’ be the scrambled image of P and its positions are (x’, y’). Then x’ and y’ can be calculated as below equations (4):

$$x' = x + mod(\big(abs\ (k_1(x) - \big|abs\big(k_1(x)\big)\big|)\big)\ \times\ 10^{15}, M - x)$$

$$y' = y + mod(\big(abs\ (k_2(y) - \big|abs\big(k_2(y)\big)\big|)\big)\ \times\ 10^{15}, N - y) \ (4)$$

Where absolute values of x and y indicates the rounding of nearest integer values either lesser or equal to x and y.

By taking P as the input image, the scrambled image P' is obtained as follows (5):

$$P'(x, y) = P(x', y'), P(x', y') = P(x, y) \qquad (5)$$

In above equation (5), Scrambled image P’ is said to be positioned at (x, y), where, x=1,2,...M, y=1,2,…N.

D. Scrambling at Bit Level

Scrambled image P’ is now converted into a sequence of one-dimensional values for P’ from P’(1) to P’(MN) beginning with leftmost upper side to rightmost lower side of an image.

Now generate chaotic sequence s_3 as follows (6):

$$s_3'(r) = mod\ ((abs(s_3(r) - \big|abs\big(s_3(r)\big)\big|)) \times 10^{15}, 8) \qquad (6)$$

Where r = 1, 2,…MN, $s_3^{'}(r) \epsilon [0, 7]$

Scrambled image P’ and decimal sequenced values of $s_3^{'}$ will be then transformed into binary sequences respectively. Then scrambled sequence C is obtained by having a circular shift over binary sequence P’(r) by considering the least bit of s_3’ as following equation (7):

$$C(r) = circularshift[P'(r), LSB\big(s_3'(r)\big), s_3'(r)] \qquad (7)$$

Finally, conversion from binary sequences into its equivalent decimal values can be required.

E. DNA Encryption

Calculate the initial values $c_1^{'}$, $c_2^{'}$, $c_3^{'}$, $c_4^{'}$ and $c_5^{'}$ of 5D hyper chaotic system (1) as following equations (8):

$$c_1'(1) = mod\ (\sum_{j=1}^{6} c_j^0, 1)$$

$$c_i' = mod\ (c_{i-1}'(1) + c_i^0, 1) \tag{8}$$

Where i=2,3,4,5 and j=1,2,3,4,5,6

This system can have sudden changes over its state as there is image data. There is a need to iterate 5D hyperchaotic system N times, as shown below (9):

$$N' = 200 + mod\left((\textstyle\sum_{i=1}^{6} c_i^0) - \left|\textstyle\sum_{i=1}^{6} c_i^0\right| \right) \times 10^{15}, 200) \tag{9}$$

The chaotic sequences a_1, a_2, a_3 and a_4 are performed as follows (10-14):

$$a_1(i) = mod\left(\left(abs\left(a_1(i)\right) - \ |a_1(i))|\right) \times 10^{15}, 6\right) + 1 \tag{10}$$

$$a_2(i) = mod\left(\left(abs\left(a_2(i)\right) - \ |a_2(i))|\right) \times 10^{15}, 4\right) \tag{11}$$

$$a_3(i) = mod((abs\left(a_3(i)\right) - |a_3(i))| \times 10^{15}, 256) \tag{12}$$

$$a_4(i) = mod((abs\left(a_4(i)\right) - |a_4(i))| \times 10^{15}, 256) \tag{13}$$

Where $a_1 \in [1, 6]$, $a_2 \in [0, 3]$, $a_3 \in [0, 255]$, $a_4 \in [0, 255]$, i = 1, 2,.., 4MN.

Each C(r) and a_3(r) are expressed as below equations (14):

$$C(r) = \sum_{s=0}^{3} c_{4r-s}(r).4^s, c_{4r-s} \in \{0,1,2,3\}$$

$$a_3(r) = \textstyle\sum_{s=0}^{3} d_{4r-s}(r).4^s, d_{4r-s} \ \in \{0,1,2,3\} \tag{14}$$

The sequences $\{c(i)\}^{4MN}_{i=1}$ and $\{d(i)\}^{4MN}_{i=1}$ can be constructed later. Where r = 1, 2, . . , MN.

Then Constructed sequences can be converted into DNA sequences. DNA sequence F(i) is obtained by using XOR operation as below (15):

$$F(i) = c'(i) \in d'(i) \tag{15}$$

Where i=1,2,…,4MN.

By using DNA replacement operation and complementary rules, get F'(i). F' can now be decoded to binary sequence G. Binary sequence G then translated into a decimal sequence.

Finally, an encrypted image R is obtained as below equations (16):

$$R(1) = a_4(1) \oplus \ mod(a_4(1) + D(1), 256) mod\left(\textstyle\sum_{j=1}^{6} x_j^0 \times 10^{15}, 256\right)$$

$$R(i) = a_4(i) \in mod(a_4(i) + D(i), 256) \oplus R(i-1) \tag{16}$$

F. DNA Decryption

The decryption procedure is described as follows:

First, the chaotic sequences are generated and used (as shown above). Then decimal sequence D is obtained by the following equations (17):

$$D(1) = mod(a_4(1) \oplus R(1) \oplus mod\left(\sum_{j=1}^{6} x_j^0 \times 10^{15}, 256\right) - a_4(1), 256)$$

$$D(i) = mod\ (a_4(i) \oplus R(i) \oplus R(i-1) - a_4(i), 256) \tag{17}$$

By following DNA complementary rules and reverse replacement operations, now able to get F(i) as the DNA sequences. Now conversion from F(i) DNA sequence to C sequence is done by having bit-level scrambling. Then Convert C sequence into pixel-level scrambled image P'. Finally, the original image P is recovered from ciphered image P'.

Where D(i) is the value of the decimal sequence, a(i) is the value of the chaotic sequence, and R(i-1) is the value of the previous cipher pixel and R(i) is the value of the output cipher.

Fig. 1 shows the flowchart of the proposed image encryption system. The original image is converted into DNA sequences by the use of chaotic sequences obtained from the 5D hyperchaotic system and scrambling at pixel and bit levels. The DNA sequences are then converted into an encrypted image by following DNA XOR and Complementary rules [22, 41, 42]. Table III shows the images that were considered in the proposed approach.

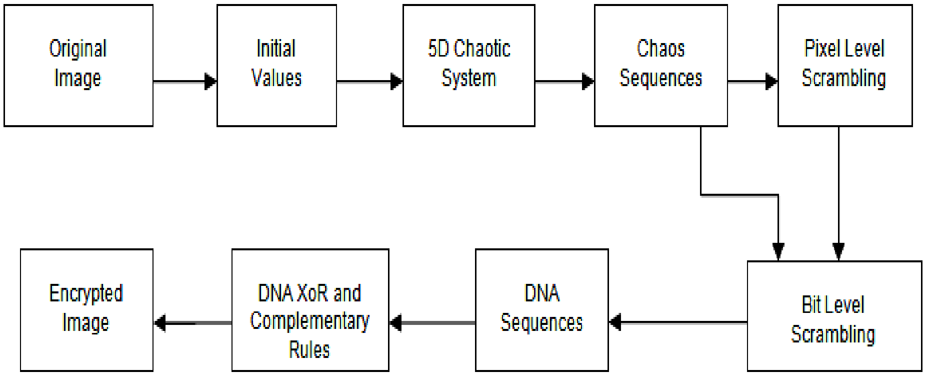
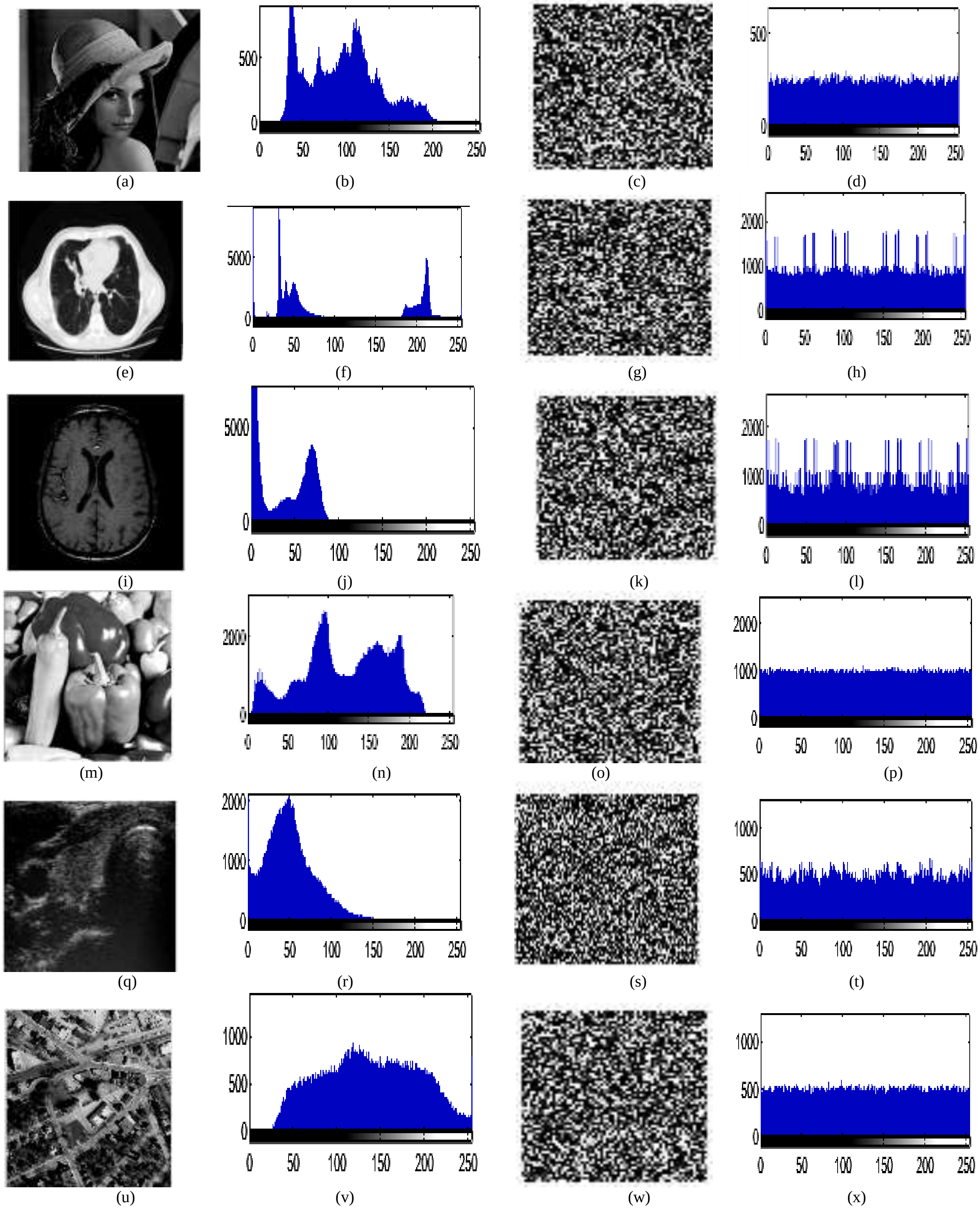


Fig. 1. Flowchart of Proposed Image Encryption System.



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

(e) (f) (g) (h)

(i) (j) (k) (l)

(m) (n) (o) (p)

(q) (r) (s) (t)

(u) (v) (w) (x)

Fig. 2. (a) Original "Lena" Image (b) Initial Image Histogram "Lena" (c) Encrypted "Lena" Image (d) Final Image Histogram "Lena" (e) Original “ChestCT” Image (f) Initial Image Histogram "ChestCT " (g) Encrypted " ChestCT " Image (h) Final Image Histogram "ChestCT" (i) Original "MRI" Image (j) Initial Image Histogram "MRI" (k) Encrypted "MRI" Image (l) Final Image Histogram "MRI" (m) Original "Peppers" Image (n)) Initial Image Histogram " Peppers " (o) Encrypted " Peppers " Image (p) Final Image Histogram " Peppers " (q) Original "Ultrasound_Thyroid" Image (r) Initial Image Histogram " Ultrasound_Thyroid " (s) Encrypted " Ultrasound_Thyroid" Image (t) Final Image Histogram " Ultrasound_Thyroid" (u) Original " Aerial" Image (v) Initial Image Histogram "Aerial" (w) Encrypted "Aerial" Image (x) Final Image Histogram "Aerial".

TABLE III. IMAGES CONSIDERED

Image	**Dimension (Width × Height)**	**Horizontal Resolution**	**Vertical Resolution**	**Bit Depth**
Lena	256X256	96 dpi	96 dpi	8
Peppers	256X256	96 dpi	96 dpi	8
MRI	256X256	72 dpi	72 dpi	8
ChestCT	256X256	72 dpi	72 dpi	8
Ultrasound_Thyroid	256X256	72 dpi	72 dpi	8
Aerial	256X256	72 dpi	72 dpi	8

G. Image Encryption Results

1) Image histogram analysis: Image histogram represents the numbers of pixels that make an Image. Histogram Analysis helps us to understand the quality of image encryption [24]. A ciphered image histogram should have a uniform distribution [3,10,17,18,24,32]. Fig. 2 shows some centralized values for plain images(b,f,j,n,r,v) whereas there are more flat values for ciphered images(d,h,l,p,t,x) exists, which makes that the proposed system could withstand statistical attacks.

2) Key space and sensitivity metrics: The key space shows that all possible keys have been used [48]. Here chaotic sequences produced and used are combined along with precision value 10^{-15} to bring accurate refinement such as $c^0{}_1$+ 10^{-15}, $c^0{}_2$+ 10^{-15}, $c^0{}_3$+ 10^{-15}, $c^0{}_4$+ 10^{-15}, $c^0{}_5$+ 10^{-15},…Hence, it leads to a larger key space around $(10^{15})^6=10^{90}=2^{298}$ which makes, this approach strong against brute force and dictionary attacks [16,17,31,40].

Key sensitivity refers to how much change in the key can impact to produce a ciphered image. Again this can be measured by parameters such as NPCR and UACI discussed above [18, 19]. A good approach is always sensitive, even a small change in the key to bringing out more diffusion or permutation in an image [39]. Hence the proposed approach is said to be resistive against differential and statistical attacks.

3) Correlation co-efficient analysis: Correlation coefficient values indicate the relationship between the pixels which are adjacent to each other [16, 17]. Smaller the values of correlation co-efficient show the greater security against attacks as resisting ability against them [5, 16, 27, 31]. Table IV lists the correlation coefficient values of six different images with diagonal, horizontal, and vertical values.

The proposed system could able to produce smaller co-efficient values either in diagonal, horizontal, or vertical directions when compared to existing approaches listed in the following Table IV for images Lena, Aerial, and Peppers. The correlation coefficient c_{pq} is computed as follows (18) [1,18,21]:

$$c_{pq} = \frac{cov(p,q)}{\sqrt{D(p)D(q)}} \tag{18}$$

TABLE IV. CORRELATION CO-EFFICIENT VALUES

Images	**Diagonal**	**orizontal**	**Vertical**
Lena	0.001809	0.001851	0.002981
ChestCT	0.002234	0.001511	-0.001400
MRI	-0.000581	-0.000074	0.000328
Peppers	0.000790	0.000901	0.004058
Ultrasound_ Thyroid	-0.000291	-0.000806	0.000476
Aerial	0.002243	-0.000986	0.003369
Lena[16]	0.0110	3.4459e-004	−0.0064
Lena[22]	0.0010	0.0068	−0.0054
Lena[29]	0.008006	0.011816	−0.017311
Aerial[16]	0.0110	−0.0109	−0.0211
Peppers[18]	−0.0107	−0.0010	−0.0292
Peppers[19]	0.0020185	−0.0013436	0.0066809
Peppers[29]	−0.009679	−0.015974	0.035035

Where p and q are adjacent pixels. $cov\,(p,q)$ is the covariance between two pixels p and q. It is given as follows (19):

$$cov(p,q) = \left(\frac{1}{N}\right)\sum_{i=1}^{N}\big(p_i - E(p)\big)\big(q_i - E(q) \tag{19}$$

Where

$$E(p) = \left(\frac{1}{N}\right)\sum\nolimits_{i=1}^{N} p_i$$

$$D(p) = \left(\frac{1}{N}\right)\sum\nolimits_{i=1}^{N} (p_i - E(p))^2$$

4) Information entropy: Shannon introduced Information Entropy in the year 1948, which describes how much information is provided by an image. It helps us to understand the uncertainty or randomness level of an image [3, 16, 29, 43]. Uncertainty level before and after image encryption is measured. Reduced value of entropy indicates lesser the information provided by the encrypted image [18, 27, 31, 32]. The entropy of information should be 8 for an 8 bit image [30]. And it must be having the range 0 through 8. Table V shows Information entropy values for six different images considered. The proposed approach can produce a higher value of information entropy for an image sample Peppers when compared to some existing approaches.

Let n be the source of information. So the entropy of information can be measured as follows (20):

$$H(n) = -\sum_{i=1}^{L} p(n_i) log_2 p\,(n_i) \tag{20}$$

Where $p(n_i)$ is the probability of appearance of variable n_i and L indicates the length of an information in terms of total number of pixel variables.

TABLE V. INFORMATION ENTROPY VALUES

Images	**Entropy**
Lena	7.996402
ChestCT	7.951614
MRI	7.939766
Peppers	7.999182
Ultrasound_Thyroid	7.989553
Aerial	7.998492
Peppers[17]	7.9973520
Peppers[19]	7.9963
Peppers[22]	7.9967
Peppers[29]	7.997275

5) Analysis of NPCR and UACI metrics: The NPCR (Number of Pixels Change Rate) represents the number of different pixels in two images. In other words, NPCR helps us to understand the effect of change of single-pixel over an image. The UACI (Unified Average Changing Intensity) represents the difference in average pixel values of intensity between two images [1-5, 16-18,27, 29, 30-32]. Here the original image and an encrypted image for computations are considered. Ciphered image will significantly change if there is a tiny change in the pixel of a plain image.

Suppose C_1, C_2 are two ciphered images. Assume that these ciphered images are having a single pixel difference with their corresponding plain images. Let $C_1(i,j)$ and $C_2(i,j)$ at row i and column j respectively are two gray-level representations of the ciphered images C_1 and C_2.

The NPCR is obtained as follows (21):

$$NPCR = \left(\sum_{i=1}^{W}\sum_{j=1}^{H} R(i,j) \times 100\%\right)/(W \times H) \qquad (21)$$

Where

$$R(i,j) = \begin{cases} 1\ C_1(i,j) \neq C_2(i,j) \\ 0\ C_2(i,j) = C_2(i,j) \end{cases}$$

The UACI is calculated as follows (22):

$$UACI = \left(\sum_{i=1}^{W}\sum_{j=1}^{H} \frac{C_1(i,j)-C_2(i,j)}{255}\right) \times 100\%)/(W \times H) \qquad (22)$$

Where W and H correspond to the width and height of the image. Table VI shows the values of NPCR and UACI of various images. It is found that when compared with existing approaches, the proposed approach can obtain a higher UACI value for the encrypted Lena picture. The NPCR value for encrypted Lena image is almost nearer to existing approaches. A higher value of the NPCR and UACI means the system is safer against differential attacks.

TABLE VI. NPCR AND UACI VALUES

Images	**NPCR**	**UACI**
Lena	99.6279	49.7571
ChestCT	11.3342	2.8501
MRI	99.2271	24.9121
Peppers	99.2271	24.8966
Ultrasound_Thyroid	11.3074	5.7084
Aerial	99.2344	26.1513
Lena[12]	99.7570	39.12
Lena[16]	99.6067	33.4951
Lena[17]	99.6135	30.9255
Lena[19]	99.5892	33.4358
Lena[22]	99.61	33.46
Lena[29]	99.608337	33.431251

IV. DNA BASED TEXT ENCRYPTION

Symmetric algorithms are quicker in performing computations. One problem with these algorithms is more security breaches since single key usage. It means if an intruder receives the single key shared over a public channel, it can hack the entire network [9, 28]. Public key cryptographic algorithms, on the other hand, have proved adequate security for the systems. But here, more time is required to perform computations. There are some recent works in which the concept of DNA cryptography is combined with traditional algorithms such as AES, RSA, and ECC have provided better security for text-related encryption and transmission in the current computing age [20,23,33]. Due to its uniqueness, randomness, increased storage capabilities, high parallelism DNA Computing is gaining more popularity in cloud computing, Ubiquitous computing areas [15].

This section discusses, along with the study of its performance, DNA-based text encryption and decryption processes.

A. DNA based Text Encryption/Decryption

The text encryption/decryption uses DNA encoding rules including a single-point fusion, mutation, and complementary rules. Single-point cross over is the one where two bases are merged in order to build other bases. Mutation means modification in a DNA sequence by some means [25]. Here in every encryption, a random key will be generated and is used to encrypt data. The same is used during decryption to decrypt the ciphertext. It is important to generate random keys that are to be used to preserve the dynamicity of the proposed work [26, 28]. It means different transactions use different keys to produce different ciphertexts. It makes a more difficult cipher to break by an attacker. The Algorithm below shows the step-by-step procedure for the text encryption method of the proposed model. The decryption technique is precisely the reverse of an encryption process [44, 46].

Algorithm: Text Encryption Process

```
1. Start:
2. Read the plain text;
3. Generate Random Key
        Display Key length in bits and its value
4. Initialize round_no and decryption key
5. Invoke generate_preprocessing_tables ()
        Conversion from two bits to DNA bases
        Conversion from DNA bases to bits
6. Invoke generate_mutation_tables ()
        Conversion from four bits to two DNA bases
        Conversion from two DNA bases to four bits
7. Start Encryption Time
8. Invoke dnaChaosSecFuncE ()
        Define the number of rounds
        Get binarized data of text
        Convert binarized data into DNA sequences
        Display Initial DNA sequence
9. While (number of rounds > 0)
        Conversion from DNA bases into bits and encrypt the
        data using Key
10. Invoke crossover ()
        Invoke single_point_crossover ()
        Invoke rotate_crossover ()
11. Invoke mutation()
        Follow DNA complementary rules
12. Invoke reverse_reshape()
        Return reverse_reshape
13. End While ().
14. Display Final DNA Sequence
15. End Encryption Time
16. Display total execution time for encryption
17. End
```

In the above algorithm two bits values used are '00', '01', '10', '11'. Initial DNA bases are 'A', 'C', 'G', 'T'. Four bits values are '0000', '0001', '0010', '0011', '0100', '0101', '0110', '0111', '1000', '1001', '1010', '1011', '1100', '1101', '1110', '1111'. Two DNA bases used are 'TA', 'TC', 'TG', 'TT', 'GA', 'GC', 'GG', 'GT', 'CA', 'CC', 'CG', 'CT', 'AA', 'AC', 'AG', 'AT'.

The following content shows the operations that are taken place during the decryption of encrypted text.

```
<reshape>4<reshape><crossover><type>both<type><rotate><rot
ation_offset>2<rotation_offset><rotation_types>right|left|right|ri
ght|right|<rotation_types><rotate><single_point>2|3|<single_poi
nt><crossover><mutation><mutation_table>{'A':'C','C':'A','T':'G',
'G':'T'}<mutation_table><chromosome><complement_mutation>
(0,5)<complement_mutation><alter_mutation>(1,3)<alter_mutati
on><chromosome><chromosome><complement_mutation>(5,6)
<complement_mutation><alter_mutation>(3,3)<alter_mutation>
<chromosome><chromosome><complement_mutation>(2,5)<co
mplement_mutation><alter_mutation>(3,3)<alter_mutation><chr
omosome><chromosome><complement_mutation>(3,6)<comple
ment_mutation><alter_mutation>(0,1)<alter_mutation><chromos
ome><chromosome><complement_mutation>(1,4)<complement
_mutation><alter_mutation>(2,3)<alter_mutation><chromosome
><mutation><round><round><reshape>2<reshape><crossover>
<type>rotate_crossover<type><rotate><rotation_offset>2<rotatio
n_offset><rotation_types>right|left|right|right|left|left|right|left|rig
ht|left|<rotation_types><rotate><crossover><mutation><mutatio
n_table>{'A':'T','T':'A','C':'G','G':'C'}<mutation_table><chromoso
me><complement_mutation>(0,3)<complement_mutation><alter
_mutation>(0,0)<alter_mutation><chromosome><chromosome>
<complement_mutation>(3,3)<complement_mutation><alter_mu
tation>(1,1)<alter_mutation><chromosome><chromosome><co
mplement_mutation>(1,1)<complement_mutation><alter_mutati
on>(0,0)<alter_mutation><chromosome><chromosome><compl
ement_mutation>(2,3)<complement_mutation><alter_mutation>(
0,1)<alter_mutation><chromosome><chromosome><complemen
t_mutation>(2,)<complement_mutation><alter_mutation>(0,1)<a
lter_mutation><chromosome><chromosome><complement_mut
ation>(3,3)<complement_mutation><alter_mutation>(1,1)<alter_
mutation><chromosome><chromosome><complement_mutation
>(0,0)<complement_mutation><alter_mutation>(1,1)<alter_muta
tion><chromosome><chromosome><complement_mutation>(0,0
)<complement_mutation><alter_mutation>(1,1)<alter_mutation>
<chromosome><chromosome><complement_mutation>(1,2)<co
mplement_mutation><alter_mutation>(0,0)<alter_mutation><chr
omosome><chromosome><complement_mutation>(1,3)<comple
ment_mutation><alter_mutation>(0,0)<alter_mutation><chromos
ome><mutation><round><round><reshape>4<reshape><crossov
er><type>single_point_crossover<type><single_point>0|3|<singl
e_point><crossover><mutation><mutation_table>{'A':'G','G':'A','
T':'C','C':'T'}<mutation_table><chromosome><complement_mut
ation>(2,6)<complement_mutation><alter_mutation>(3,3)<alter_
mutation><chromosome><chromosome><complement_mutation
>(6,6)<complement_mutation><alter_mutation>(0,0)<alter_muta
tion><chromosome><chromosome><complement_mutation>(6,7
)<complement_mutation><alter_mutation>(0,1)<alter_mutation>
<chromosome><chromosome><complement_mutation>(4,5)<co
mplement_mutation><alter_mutation>(0,1)<alter_mutation><chr
omosome><chromosome><complement_mutation>(2,3)<comple
ment_mutation><alter_mutation>(2,3)<alter_mutation><chromos
ome><mutation><round><round><reshape>4<reshape><crossov
er><type>single_point_crossover<type><single_point>3|0|<singl
e_point><crossover><mutation><mutation_table>{'G':'A','A':'G','
T':'C','C':'T'}<mutation_table><chromosome><complement_mut
ation>(3,5)<complement_mutation><alter_mutation>(2,3)<alter_
mutation><chromosome><chromosome><complement_mutation
>(4,5)<complement_mutation><alter_mutation>(3,3)<alter_muta
tion><chromosome><chromosome><complement_mutation>(5,6
)<complement_mutation><alter_mutation>(0,0)<alter_mutation>
<chromosome><chromosome><complement_mutation>(4,5)<co
mplement_mutation><alter_mutation>(1,2)<alter_mutation><chr
omosome><chromosome><complement_mutation>(3,5)<comple
ment_mutation><alter_mutation>(2,2)<alter_mutation><chromos
ome><mutation><round><round><reshape>10<reshape><crosso
ver><type>both<type><rotate><rotation_offset>6<rotation_offse
t><rotation_types>left|right|<rotation_types><rotate><single_poi
nt>8|<single_point><crossover><mutation><mutation_table>{'G'
:'A','A':'G','T':'C','C':'T'}<mutation_table><chromosome><compl
ement_mutation>(13,15)<complement_mutation><alter_mutatio
n>(2,4)<alter_mutation><chromosome><chromosome><comple
ment_mutation>(14,16)<complement_mutation><alter_mutation
>(0,4)<alter_mutation><chromosome><mutation><round>
```

B. *Text Encryption Results*

The proposed system can support Image and text encryption under a single framework. Text encryption begins upon selecting the “Text” radio button followed by clicking the Process Encryption and Decryption Operation button, as shown in Fig. 3.

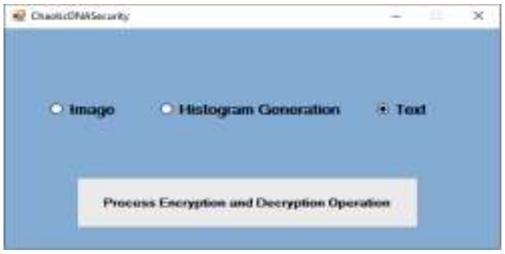


Fig. 3. User Interface to Select Text to Process Encryption and Decryption Operation.

In this proposed approach DNA cryptographic functions are designed and implemented with the use of Python script. The front end and the algorithm calling functions are designed using C # language written over Microsoft Visual C # Express. The following results were obtained on running python scripts over Spyder (Python 3.6) platform.

Case 1:

Encryption Process

Text: Hello

Key: 128 bits

0011100000100101001010011010001100011000101010000010110010110010101101011001101000100000101001001001100100100100101001111101001100

Initial DNA sequence: CAGACGCCCGTACGTACGTT

Final DNA sequence: CTTAACACGTAGCTCCAGTA

Decryption Process

Encrypted text: CTTAACACGTAGCTCCAGTA

Key: 128 bits

0011100000100101001010011010001100011000101010000010110010110010101101011001101000100000101001001001100100100100101001111101001100

Initial DNA sequence: CTTAACACGTAGCTCCAGTA

Decrypted text: Hello

Case 2:

Encryption Process

Text: Hello

Key: 128 bits

0101001000110011001101010110101000110011010010010100111000110100011100100110111101010001010001110111010101000111010001110011100 0

Initial DNA sequence: CAGACGCCCGTACGTACGTT

Final DNA sequence: TGAGTGGTTCCGCAAGCAGA

Decryption Process

Encrypted text: TGAGTGGTTCCGCAAGCAGA

Key: 128 bits

01010010001100110011010101101010001100110100100101001110001101000111001001101111010100010100011101110101010001110100011100111000

Initial DNA sequence: TGAGTGGTTCCGCAAGCAGA

Decrypted text: Hello

The above example illustrates the proposed model for text encryption using DNA sequences with a text sample as “Hello”. First, the key will be computed then it can be used to produce initial and final DNA sequences. Meantime, DNA encoding rules along with single-point crossover, mutation, and complementary rules are used to obtain encrypted text [46]. During the decryption process, the same procedure is repeated and reversed, with the same key value as shown in Case 1 [47].

When there is another communication session with the same text input as “Hello” as shown in Case 2, then it computes a key value that is different than the previous session key. This will bring the proposed approach to high dynamicity and randomness. Since the keys generated and used were different during the various sessions/transactions makes it difficult to access the key computationally. Hence plaintext recovery is infeasible for an attacker. Table VII shows time required values in seconds for encryption and decryption processes. The time required for encryption and decryption is often found to be comparatively closer and takes less time. The decryption needs little more time than the encryption.

Table VIII shows the memory allocation (in bytes) of the proposed DNA Cryptographic method for encryption on disk. Table values indicate there is no need for more memory than the input file size.

TABLE VII. TIME REQUIRED VALUES

Case	Encryption Time(sec)	Decryption Time(sec)
Case 1	0.0032889842987060547	0.007483005523681641
Case 2	0.006028413772583008	0.009949922561645508

TABLE VIII. MEMORY REQUIREMENT FOR ENCRYPTION

Input Text size	Input file Size on disk	Encrypted file Size	Encrypted file Size on disk
86 bytes	4 KB (4096 bytes)	4.03 KB (4,128 bytes)	8.00 KB (8,192 bytes)
996 bytes	4.00 KB (4,096 bytes)	46.6 KB (47,808 bytes)	48.0 KB (49,152 bytes)

V. CONCLUSION

The proposed new DNA based cryptographic framework provides security to both image and text. The proposed approach can resist Differential attack, Brute Force attack, Chosen Plaintext attack, Dictionary attack, and other Statistical Attacks. The experimental results have shown that the histograms of encrypted images are uniformly distributed. Correlation coefficient values are found to be smaller in one or more directions. The proposed image encryption technique is better than existing in terms of information entropy, NPCR, and UACI values. Information entropy value for Peppers image is found to be 7.999182, which is 0.30% improvement over existing works. NPCR and UACI values for Lena image are 99.6279 and 49.7571, respectively. UACI value of the proposed method shows 1.06% improvement for encrypted Lena image over existing approaches. These parameters have shown that the proposed method is stronger. Proposed work support image files such as .jpg, .png, jpeg, .tiff formats, and text characters as input. In addition to Image encryption, there can be a text encryption option is also provided under the same framework. Compared with conventional algorithms, the overall execution time is considerably reduced in the proposed text encryption method. For the suggested solution, space consumption is less. In the future, this system may include audio encryption and can support very large files with different file formats for encryption and decryption processes.

ACKNOWLEDGMENT

I Bahubali Akiwate take this opportunity to express my deep sense of gratefulness to my guide and mentor Dr. Latha Parthiban, Pondicherry University, Pondicherry, India for her valuable advice, expert guidance, and unceasing support at all the stages during this research work. I extend my sincere gratitude to Dr. Veena Desai, Gogte Institute of Technology, Belagavi, Karnataka, India for her timely suggestions and encouragement in accomplishing this work.

REFERENCES

[1] M.A. Ben Farah, R. Guesmi, A. Kachouri, M. Samet, "A novel chaos based optical image encryption using fractional Fourier transform and DNA Sequence operation" Optics and Laser Technology 121 (2020) 105777.

[2] Kang Xuejing, Guo Zihui, "A new color image encryption scheme based on DNA encoding and spatiotemporal chaotic system", Signal Processing: Image Communication 80 (2020) 115670.

[3] K.C. Jithin, Syam Sankar, "Colour image encryption algorithm combining Arnold map, DNA sequence operation, and a Mandelbrot set", Journal of Information Security and Applications 50 (2020) 102428.

[4] Je Sen Teh, Moatsum Alawida, You Cheng Sii, "Implementation and practical problems of chaos-based cryptography revisited" Journal of Information Security and Applications 50 (2020) 102421.

[5] Qiang Zhang, Ling Guo, Xiaopeng Wei, "Image encryption using DNA addition combining with chaotic maps", Mathematical and Computer Modelling 52 (2010) 2028-2035.

[6] Ivan Jiron, Susana Soto, "A new DNA-based model for finite field arithmetic" Heliyon 5 (2019) e02901.

[7] Ahmed M. Elshamy, Aziza I. Hussein, "Color Image Encryption Technique Based on Chaos", Procedia Computer Science 163 (2019) 49–53.

[8] Said HRAOUI, Faiq Gmira, "A New Cryptosystem of Color Image Using a Dynamic-Chaos Hill Cipher Algorithm", Procedia Computer Science 148 (2019) 399–408,

[9] Saswat K Pujari, Gargi Bhattacharjee, "A Hybridized Model for Image Encryption through Genetic Algorithm and DNA sequence", Procedia Computer Science 125 (2018) 165–171.

[10] N. Sasikaladevi, K. Geetha, A. Revathi, "EMOTE – Multilayered encryption system for protecting medical images based on binary curve", Journal of King Saud University – Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2019.01.014.

[11] Manreet Sohal, Sandeep Sharma," BDNA-A DNA inspired symmetric key cryptographic technique to secure cloud computing", Journal of King Saud University – Computer and Information Sciences (2018), https://doi.org/10.1016/j.jksuci.2018.09.024.

[12] Bhaskar Mondal, Tarni Mandal, "A light weight secure image encryption scheme based on chaos & DNA computing", Journal of King Saud University – Computer and Information Sciences (2017) 29, 499–504.

[13] F.J. Farsana, V.R. Devi, "An audio encryption scheme based on Fast Walsh Hadamard Transform and mixed chaotic key streams", Applied Computing and Informatics, https://doi.org/10.1016/j.aci.2019.10.001.

[14] Mumthas S, Lijiya A, "Transform Domain Video Steganography Using RSA, Random DNA Encryption and Huffman Encoding", Procedia Computer Science 115 (2017) 660–666.

[15] Md. Rafiul Biswas, "A technique for DNA cryptography based on dynamic mechanisms", Journal of Information Security and Applications 48 (2019) 102363.

[16] Junxin Chen, "Exploiting self-adaptive permutation–diffusion and DNA random encoding for secure and efficient image encryption", Signal Processing 142 (2018) 340–353.

[17] Xingyuan Wang, Yu Wang, "A novel chaotic algorithm for image encryption utilizing one-time pad based on pixel level and DNA level", Optics and Lasers in Engineering 125 (2020) 105851.

[18] Wenjian Gao, Jie Sun, "Digital image encryption scheme based on generalized Mandelbrot- Julia set", Optik - International Journal for Light and Electron Optics 185 (2019) 917–929.

[19] Yu-Guang Yang, Bo-Wen Guan, "Image compression-encryption scheme based on fractional order hyperchaotic systems combined with 2D compressed sensing and DNA encoding", Optics and Laser Technology 119 (2019) 105661.

[20] Hossein Movafegh Ghadirli, "An overview of encryption algorithms in color images", Signal Processing 164 (2019) 163–185.

[21] Rasul Enayatifar, "Index-based permutation-diffusion in multiple-image encryption using DNA sequence", Optics and Lasers in Engineering 115 (2019) 131–140

[22] Shuliang Sun, "A Novel Hyperchaotic Image Encryption Scheme Based on DNA Encoding, Pixel-Level Scrambling and Bit-Level Scrambling" IEEE Photonics Journal, Volume 10, Number 2, April 2018.

[23] Taiyong Li, Minggao Yang, "A Novel Image Encryption Algorithm Based on a Fractional-Order Hyperchaotic System and DNA Computing", Hindawi Complexity Volume 2017, Article ID 9010251, https://doi.org/10.1155/2017/9010251.

[24] Z. Azimi S. Ahadpour, "Color image encryption based on DNA encoding and pair coupled chaotic maps" Multimedia Tools and Applications https://doi.org/10.1007/s11042-019-08375-6, part of Springer Nature 2019.

[25] Hatem M. Bahig, "DNA-Based AES with Silent Mutations", Arabian Journal for Science and Engineering (2019) 44:3389–3403.

[26] Oinam Bidyapati Chanu, " A survey paper on secret image sharing schemes", International Journal of Multimedia Information Retrieval (2019) 8:195–215. (part of Springer Nature 2018).

[27] Siyamol Chirakkarottu, Sheena Mathew, "A novel encryption method for medical images using 2D Zaslavski map and DNA cryptography", https://doi.org/10.1007/s42452-019-1685-8, Springer Nature Switzerland AG 2019.

[28] Ahmed Elhadad, "Data sharing using proxy re-encryption based on DNA computing", Soft Computing https://doi.org/10.1007/s00500-019-04041-z, Springer-Verlag GmbH Germany, part of Springer Nature 2019.

[29] M. A. Ben Farah, A. Farah, “An image encryption scheme based on a new hybrid chaotic map and optimized substitution box”, Springer Nature B.V. 2019.

[30] Manjit Kaur, Vijay Kumar, “A Comprehensive Review on Image Encryption Techniques”, Archives of Computational Methods in Engineering (Springer 2018).

[31] Xingyuan Wang, Huaihuai Sun, ” A chaotic image encryption algorithm based on zigzag-like transform and DNA-like coding”, Multimedia Tools and Applications (2019) 78:34981–34997, Part of Springer Nature 2019.

[32] Xiangjun Wu, ” Lossless chaotic color image cryptosystem based on DNA encryption and entropy” ,Nonlinear Dyn (2017) 90:855–875 DOI 10.1007/s11071-017-3698-4.

[33] Kareem Ahmed, Ibrahim El-Henawy, “Increasing robustness of data encryption standard by integrating DNA cryptography”, International Journal of Computers and Applications, VOL. 39, NO. 2, 91–105, 2017.

[34] P. K. Naskar & A. Chaudhuri, “Secured secret sharing technique based on chaotic map and DNA encoding with application on secret image”, The Imaging Science Journal, 64:8, 460-470, DOI: 10.1080/13682199.2016.1239427, 2016.

[35] Shahna k.U., Anuj Mohamed, “A novel image encryption scheme using both pixel level and bit level permutation with chaotic map”, Applied Soft Computing Journal 90 (2020) 106162.

[36] Junxin Chen,Lei Chen, Yicong hou, “Cryptanalysis of a DNA-based image encryption scheme”, Information Sciences 520 (2020) 130-141.

[37] Abdorreza Babei, Homayun Motameni, “A new permutation-diffusion-based image encryption technique using cellular automata and DNA sequence”, Optik-International Journal for Light and Electron Optics 203 (2020) 164000.

[38] Yujia Liu, “Optical image encryption algorithm based on hyper-chaos and public-key cryptography”, Optics and Laser Technology 127 (2020) 106171.

[39] Hongye Niu — Changjun Zhou, “Splicing Model And Hyper–Chaoti System For Image Encryption”, Journal of ELECTRICAL ENGINEERING, VOL 67 (2016), NO2, 78–86.

[40] Wei Feng, Yigang He. "Cryptanalysis and improvement of the hyper-chaotic image encryption scheme based on DNA encoding and scrambling", IEEE Photonics Journal, 2018.

[41] Shuliang Sun, Yongning Guo, Ruikun Wu. "A Novel Image Encryption Scheme Based on 7D Hyperchaotic System and Row-column Simultaneous Swapping", IEEE Access, 2019.

[42] K. Abhimanyu Kumar Patro, Bibhudendra Acharya, Vijay Nath. "Various dimensional colour image encryption based on non overlapping block-level diffusion operation", Microsystem Technologies, 2019.

[43] Grasha Jacob, Murugan Annamalai. "DNA Sequence Based Cryptographic Solution for Secure Image Transmission", IGI Global, 2016.

[44] Wang, Xing-Yuan, Ying-Qian Zhang, and Xue- Mei Bao. "A novel chaotic image encryption scheme using DNA sequence operations", Optics and Lasers in Engineering, 2015.

[45] Sohal M, Sharma S, "DNA Inspired Symmetric Key Cryptographic Technique to Secure Cloud Computing", Journal of King Saud University - Computer and Information Sciences, 2018.

[46] Ying-Qian Zhang, Yi He, Pi Li, Xing-Yuan Wang. "A new color image encryption scheme based on 2DNLCML system and genetic operations", Optics and Lasers in Engineering, 2020.

[47] Mohammad Seyedzadeh, S.. "A fast color image encryption algorithm based on coupled two dimensional piecewise chaotic map", Signal Processing, 201205.

[48] Roayat Ismail Abdelfatah. "Audio Encryption Scheme Using Self-Adaptive Bit Scrambling and Two Multi Chaotic-Based Dynamic DNA Computations", IEEE Access, 2020.