A Fast Image Encryption based on Linear Feedback Shift Register and Deoxyribonucleic acid

Hasan Ghanbari
Islamic Azad University Sari Branch
Rasul Enayatifar (r.enayatifar@gmail.com)
Islamic Azad University, Firoozkooh Branch
Homayun Motameni
Islamic Azad University Sari Branch

Research Article

Keywords: Image encryption, Linear Feedback Shift Register, Deoxyribonucleic acid

Posted Date: May 19th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1662684/v1

A Fast Image Encryption based on Linear Feedback Shift Register and Deoxyribonucleic acid

Hasan Ghanbari[1], Rasul Enayatifar [2,*], Homayun Motameni[3]

1 Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran

2 Department of Computer Engineering, Firoozkooh Branch, Islamic Azad University, Firoozkooh, Iran

3 Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran

* *Corresponding author: r.enayatifar@gmail.com*

Abstract- Recently with the proliferating rates of digital data and the need of sending data through the internet, protection of the contents is crucial. This article proposes a quick method based on a combination of Linear Feedback Shift Register (LFSR) and Deoxyribonucleic acid (DNA) and Tinkerbell chaotic function. This method not only has high protection but also provides high encryption speed. First, a bit shift method is implemented in permutation phase. Shift numbers are determined by LSFR in each step. Then in each step of the permutation phase, specific numbers of pixels are delivered for final encryption in diffusion phase. In this phase, DNA as well as Tinkerbell index converts gray level of image pixels. The results show high security, incredible speed and good resistance against prevailing attacks.

Keywords- Image encryption; Linear Feedback Shift Register; Deoxyribonucleic acid

1. Introduction

The everlasting evolution of internet and social networking has brought about various Internet-based media other than common instruments to send emails. Any data like digital images and videos are usually transmitted through the Internet and consequently how to preserve the contents of these data is a major concern to be considered and so diverse image encryption techniques have been presented recently [1]. The problem with primary techniques of image encryption such as IDEA, AES, RSA and DES is deficiency in correlation coefficient within adjoining vertical, horizontal, and diagonal features [2]. Therefore, the primary techniques are ineffective for robust encryption [3]. The acknowledged encryption of image algorithms contain chaotic-based encryption of image [4,5], developmental algorithm-based encryption of image [6,7], encryption of image in domain transformation [8, 9], DNA-based encryption of image [10, 11].

The chaotic map constituent makes chaos-based procedures to be interesting for researchers. Consequently, it gets the attention of associated researchers. Chaos-based techniques are typically initiated with the utilization of the chaotic map in the arrangement step by rearranging image pixels in a way that the gray degree of the plain image stays persistent. Though, the cipher image in the transmission step is created by transferring the gray level of the plain image in compliance with the chaotic map and the recommended encryption method[15].

The application of Deoxyribonucleic acid (DNA) theory is an immensely recent trend in image encryption [16, 17]. The developed cipher images employing DNA-based techniques have competitive and realistic correlation coefficients and entropy as stated in literature[18]. Two major steps are usually incorporated in DNA-based techniques. First, the plain image is transformed to DNA image by virtue of DNA operation rules. Second, the DNA image is employed, and DNA keys are generated from the chaotic map, then the cipher image is created.

Linear Feedback Shift Register (LFSR) is a linear technique to develop Pseudo Random Sequence Numbers which is prevalent among most linear encryption schemes to cipher stream [19]. Actually, many encryption schemes make use of LFSR to generate keys [20, 21].

This study implements a combination of LFSR, DNA and Tinkerbell chaotic function to offer a fast encryption method. In permutation phase, a bit-level permutation method is proposed based on the combination of LSFR and bit shift. In this approach, a certain number of bits are shifted in rows and columns in each step. Finally, in each step a specific number of image pixels are encrypted in diffusion phase by a combination of DNA and Tinkerbell. The particular combination of permutation and diffusion phases proposed by this article is the main cause of high speed and security in this approach. Actually, the real reason for high speed in this method is that half of the image pixels are encrypted instead of the whole image pixels.

The remainder of the paper is arranged as demonstrated: In Section 2 the primary concepts of Tinkerbell chaotic map, DNA, and LFSR are discussed. Section 3 thoroughly centers on the proposed method. In section 4 experimental results including entropy analysis, brute-force attack, statistical attacks, and differential attacks are presented followed by comparisons. Lastly, the primary findings of this work are outlined by the concluding remarks.

2.Preliminaries

LFSR, Tinkerbell chaotic map and DNA sequence are three preliminary concepts which are significant to fathom the proposed method.

2.1 Linear-Feedback Shift Register (LFSR)

LFSR Structure is a linear sequence of registers to generate pseudo random sequence in most schemes. At any specific point of time, registers manifest the current state of that stage. In addition, shift register is controlled by an external clock. So the feedback content of the register is determined by the XOR of the register content [22]. Therefore, this process is calculated by Eq.1:

$$S_{t+L} = \sum_{i=1}^{L} C_i S_{t+L-i} t \geq 0 \quad (1)$$

In this equation, S_{t+L} represents the feedback content in each stage. C and i manifest the clock and the stage numerator, respectively. Overall structure of the LFSR system is depicted in Fig. 1.

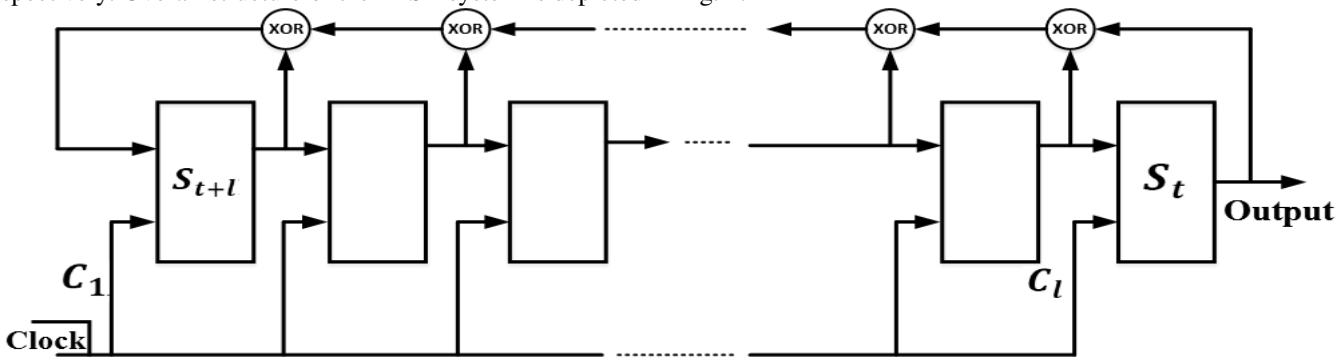


Fig. 1. Overall structure of LFSR system

2.2. Tinkerbell Chaotic Map

Chaotic maps are a group of functions which are fairly perceptive to initial parameters value. These functions exhibit chaotic demeanor where a remote change in initial parameters triggers an enormous alteration in those values which are created by chaotic function. Tinkerbell chaotic map is one of the various types of chaotic functions. It is a 2-dimension function that is defined by Eq. 2 [18].

$$x_{n+1} = x_n^2 - y_n^2 + ax_n + by_n$$
$$y_{n+1} = 2x_n y_n + cx_n + dy_n \quad (2)$$

Fig. 2 shows the Tinkerbell chaotic map behavior with $n = 1, \ldots, 1000$ and $X_1 = 0.1; Y_1 = -0.1$ and $a = 0.9; b = -0.6013; c = 2; d = 0.5$.

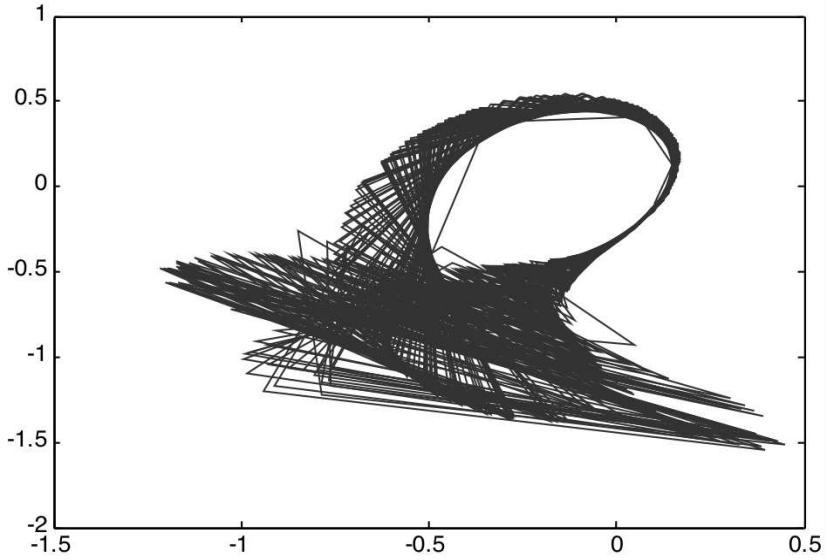


Fig. 2. Tinkerbell chaotic map with starting $X_1 = 0.1; Y_1 = -0.1$ and parameters value $a = 0.9; b = -0.6013; c = 2; d = 0.5$.

2.3. Deoxyribonucleic acid (DNA)

The primary inherited instructions for living and procreation of any creature are transmitted by DNAs. DNAs are made from nitrogen nucleic bases, that is to say; adenine (A), guanine (G), cytosine (C), and thymine (T). In conformity with base pairing rules of DNA, accompaniment chemical nitrogenous bases of the two components are connected together to form a double-stranded DNA in such way that A matches with T and C matches with G [18]. Table 1 presents the binary coding rules for DNA sequences formed by utilizing corresponding rules of DNA sequences as well as corresponding rules of the binary system [23]. Based on Table 1, a pixel with the gray level of 190 and the binary program of $(10111110)_2$ might cause a route to 8 different DNA orders as specified: Rule 1 (CTTC), Rule 2 (GTTG), Rule 3 (CAAC), Rule 4 (GAAG), Rule 5 (AGGA), Rule 6 (TGGT), Rule 7 (ACCA) and Rule 8 (TCCT). Additionally, DNA orders according to Table 2 can describe the function of XOR.

Table 1. Binary coding rules for DNA sequences

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

Table 2. Truth table for XOR operator for 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

3. Proposed method

In the present section, the phases of the suggested method are elaborated in detail. This method consists of three main phases: main key production, permutation and diffusion.

3.1. Key generation

For secret key production, message-digest algorithms (MD5) are used. MD5 consists of a set of random characters (secret key). Then the input string is changed into a hexadecimal number with 32 characters by MD5. This 32-character number is converted to binary equivalent. Finally, a 128-bit strain is provided for the sender. Initial values of Tinkerbell chaotic function and seed value for LSFR (Eq. 3) are calculated by the following equations based on the 128-bit strain.

$$K = \{k_0, k_1, k_2, \dots, k_{31}\} \tag{3}$$

Subject to: $k_i = \{k_{i,0}, k_{i,1}, , k_{i,2}, k_{i,3}\}$

Where in $k_{i,j}$, i denotes the character number and j shows the bit number in k_i. In Tinkerbell chaotic map, initial values (x_0 and y_0) are calculated as Eq.3 and Eq.4.

$$x_0 = \frac{k_{15,0}^{127} + k_{15,1}^{126} + \cdots + k_{8,0}^{63} + \cdots + k_{0,6}^{1} + k_{0,7}^{0}}{2^{128}} \tag{4}$$

$$y_0 = \frac{k_{31,0\dots3} \oplus k_{30,0\dots3} \oplus \dots \oplus k_{0,0\dots3}}{2^8} \tag{5}$$

Where $\oplus$ denotes the exclusive OR in binary system. Owing to Eq. 4 and Eq. 5, both x_0 and y_0 belong to interval $[0,1]$. Initial seed for LFSR is calculated by Eq. 6.

$$LFSR_{Seed} = \{k_{0,0}, k_{2,0}, k_{4,0}, \dots, k_{30,0}\} \tag{6}$$

3.2. Permutation

In permutation phase, bit level shift and LFSR should be used. The proposed LSFR consists of 16 registers to determine LFSR seed by Eq. For permutation, the Plain-image is converted to its binary equivalent (Eq. 7).

$$I(i,j) \rightarrow I(i,k), i \in [1,N], j \in [1,M], k \in [1, M \times 8] \quad (7)$$

The proposed LFSR is used to determine the number of left to right and top to down shifts by Eqs. 8 and 9.

$$Left\_to\_right = \lfloor log_2^{(M \times 8)} \rfloor \quad (8)$$

$$up\_to\_down = \lfloor log_2^{N} \rfloor \quad (9)$$

First LFSR is run one clock to obtain a number in [1, $Left\_to\_right$], then it is run one more clock to determine a number in [1, up_to_down]. Finally, these two numbers are employed in the first row and the first column in binary plain images. The proposed permutation for the first row and the first column is illustrated in Fig. 3. These steps are repeated until all rows and columns are met.

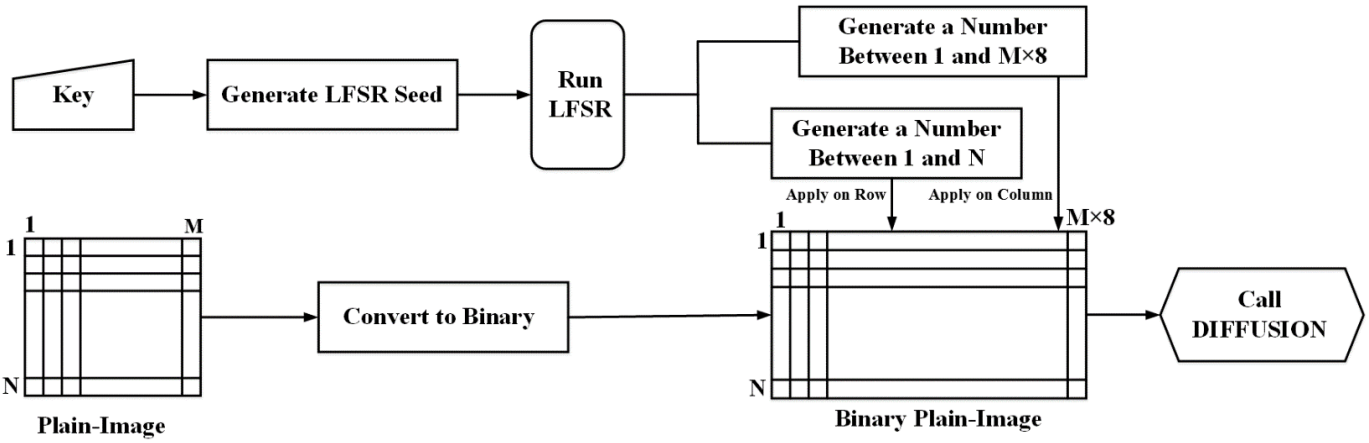


Fig. 3. The proposed permutation

3.3. Diffusion

In this phase, a certain number of pixels in the delivered image from permutation is encrypted. This number is calculated by Eq. 10.

$$\left\lfloor \frac{1}{2} \times log_2^{N \times M} \right\rfloor \quad (10)$$

Where M and N denote the numbers of the row and the column of the image respectivelyA combination of Tinkerbell chaotic function and DNA rules, shown in table 1, is implemented to encrypt a pixel. First, x_0 and $y_0$are calculated by Eq. 3 and Eq.4 before starting with Tinkerbell function.

Owing to Eq. 4 and Eq. 5, both x_0 and y_0 belong to interval$[0,1]$. Tinkerbell map shows totally chaotic behavior when x_0 and y_0 varies in interval$[-0.5,-0.2]$. To do so, obtained x_0 and y_0 by Eq. 4 and Eq. 5 will map to the mentioned interval by using Eq. 11.

$$\{x_0, y_0\} \leftarrow -0.5 + \{x_0, y_0\} \times \big(-0.2 - (-0.5)\big) \quad (11)$$

According to Eq. 11, following relations are always satisfied: $-1.23 < x_n < 0.46$ and$-1.55 < y_n < 0.55$.

Values x_0 and y_0 are employed to generate x_1 and y_1 as the starting points of Tinkerbell chaotic map. The values x and y produced by Tinkerbell function are employed to calculate the location of target pixels with Eqs. 12 and 13.

$$X\_Position \leftarrow \left\lfloor \big(x_n - (-1.23)\big) \times \left(\frac{N}{\big(0.46 - (-1.23)\big)} \right) \right\rfloor + 1 \quad (12)$$

$$Y\_Position \leftarrow \left\lfloor \big(y_n - (-1.55)\big) \times \left(\frac{M}{\big(0.55 - (-1.55)\big)} \right) \right\rfloor + 1 \quad (13)$$

$$Subjectto: Y\_PositionMOD8 = 1$$

In case the condition in Eq. 13 is not set, $Y\_Position$ value is reduced from the obtained value. Eight bits of image are separated from the locations of $X\_Position$ and $Y\_Position$ and converted to DNA sequence by the obtained rule in table one. The number of this rule is calculated by Eq. 14.

$$DNA\_R_1 \leftarrow \left\lfloor \big(x_n - (-1.23)\big) \times \left(\frac{8}{\big(0.46-(-1.23)\big)} \right) \right\rfloor + 1 \quad (14)$$

Then a value between 0 to 255 is produced by Eq. 15.

$$Temp\_Key \leftarrow \left\lfloor \big(y_n - (-1.55)\big) \times \left(\frac{255}{\big(0.55 - (-1.55)\big)} \right) \right\rfloor \quad (15)$$

This value is converted to its binary equivalent, and then to its DNA sequence equivalent by Eq. 16 and table one.

$$DNA\_R_2 \leftarrow \left\lfloor \left(y_n - (-1.55)\right) \times \left(\frac{8}{\left(0.55 - (-1.55)\right)}\right) \right\rfloor + 1 \quad (16)$$

The two DNA sequences are XORed using Table 2. The obtained value is converted to binary format using Table 1. This value is selected as the encrypted 8-bit value. The proposed flow chart for diffusion phase is illustrated in table one.

The two permutation and diffusion phases continue interdependently in smaller dimensions for image (N, M). The final pixels encrypted in diffusion phase can be calculated by Eq. 17.

$$min(N, M) \times log_2^{N \times M} \quad (17)$$

Whole proces to diffuse a pixel is shown in Fig. 4.

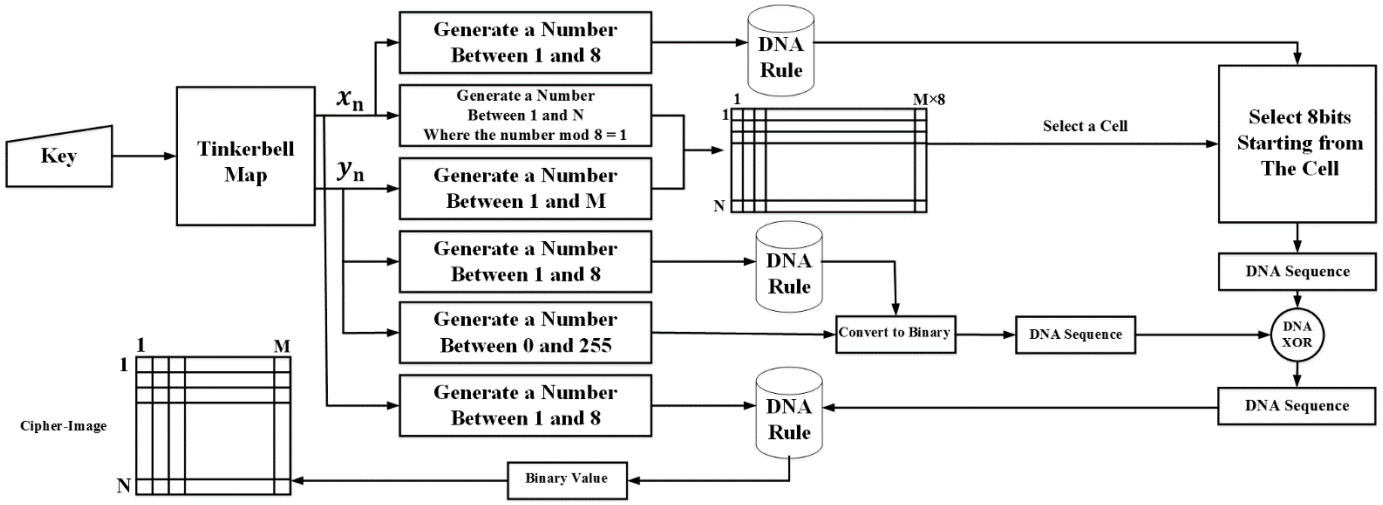


Fig. 4. The proposed diffusion

4. Experimental results

Eight 512×512 images and eight 256×256 images were examined by the proposed method to verify the results over Python 3 on a PC with an Intel Corei7, 2.3 GHz CPU, 8 GB memory and 500 GB hard disk with a Window 8 professional operating system. Then MATLAB 2019a is used for the illustration of the charts. Afterwards, further experiments were carried out on the proposed method to evaluate its efficacy. Therefore, the results of these experiments are provided in the following sections illustrating entropy, correlation coefficients and histogram analyses in statistical attacks, secret key space analyses and key sensitivity analyses in brute force attacks, number of pixel change rates (NPCR) and unified average changing intensity (UACI) in distinctive attacks.

4.1. Permutation test

To test the proposed permutation method, a sample Painter image (Fig 5.a) has been going through the permutation process considering the diffusion phase has been eliminated and only proposing shift permutation is performed. The obtained permuted image is shown in Fig 5.b.

This test has been repeated for all the images in two sizes of 256×256 and 512×512. To evaluate the quality of the proposed permutation, correlation coefficient of the permutated image in vertical, horizontal and diagonal direction for the original image (OI) and permutated image (PI) has been listed in Table 3.

The range of correlation coefficient is between -1 to +1. Moreover, the number of adjacent pixels in a plain image is compatible with its gray degrees. As a result, +1 and -1 can be considered as the perfect positive and negative correlation. However, minimum connection between two adjacent pixels is optimum in a cipher image. Since there is no connection between two adjacent pixels, the perfect correlation coefficient is 0. The correlation coefficient of two neighboring pixels is calculated by Eq. 18.

$$r_{xy} = \frac{\left|cov(x, y)\right|}{\sqrt{D(x)} \times \sqrt{D(y)}} \quad (18)$$

In which x and y are the gray levels of two adjacent pixels. Eq. 18 can be additionally determined by employing Eqs. 19, 20 and 21 in which $E(x)$ and $E(y)$ are the means of x and y variables and $D(x)$ and $D(y)$ are respective variances.

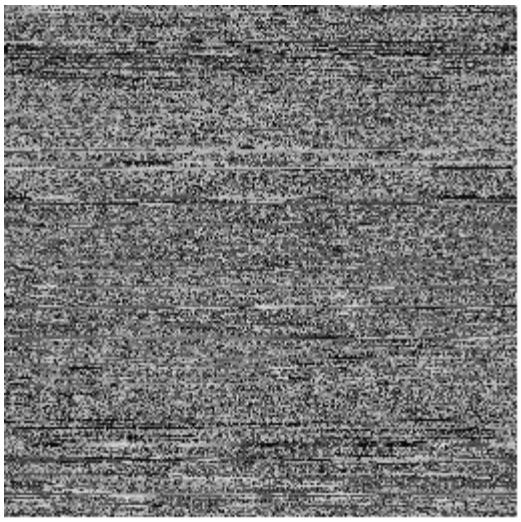
$$cov(x, y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \quad (19)$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{20}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))^2 \tag{21}$$

The obtained correlation coefficients have been listed in Table 3 that demonstrate the reasonable performance of the proposed permutation.

(a)



(b)

Fig. 5. a) Painter image, b) permutated image

Table 3. Correlation coefficients of original image (OI) and permutated image (PI)

			Lena	Cameraman	Peppers	Tree	Baboon	Airplane	Painter	Boat
256×256	Vertical	OI	0.9776	0.9702	0.9538	0.9720	0.9779	0.9718	0.9666	0.9637
		PI	0.0186	0.0185	0.0208	0.0173	0.0216	0.0184	0.0198	0.0275
	Horizontal	OI	0.9795	0.9615	0.9695	0.9635	0.9615	0.9792	0.9615	0.9682
		PI	0.0238	0.0176	0.0251	0.0157	0.0192	0.0193	0.0116	0.0192
	Diagonal	OI	0.9641	0.9589	0.9629	0.9591	0.9641	0.9641	0.9641	0.9641
		PI	0.0199	0.0086	0.0164	0.0155	0.0081	0.0142	0.0120	0.0154
512×512	Vertical	OI	0.9776	0.9702	0.9538	0.9720	0.9779	0.9718	0.9666	0.9637
		PI	0.0086	0.0089	0.0148	0.0120	0.0127	0.0107	0.0172	0.0111
	Horizontal	OI	0.9795	0.9615	0.9695	0.9635	0.9615	0.9792	0.9615	0.9682
		PI	0.0238	0.0114	0.0143	0.0211	0.0122	0.0119	0.0130	0.0162
	Diagonal	OI	0.9641	0.9589	0.9629	0.9591	0.9641	0.9641	0.9641	0.9641
		PI	0.0199	0.0081	0.0076	0.0189	0.0091	0.0094	0.0138	0.0081

4.2. The proposed method performance

The performance of proposed method is evaluated in three stages, and the obtained results are illustrated in Fig. 6. Fig. 6a is related to Peppers image with dimensions of 256×256. Figs. 6b, 6c and 6d are, respectively, the encrypted images after implementing 25%, 50% and 100% of the proposed method.

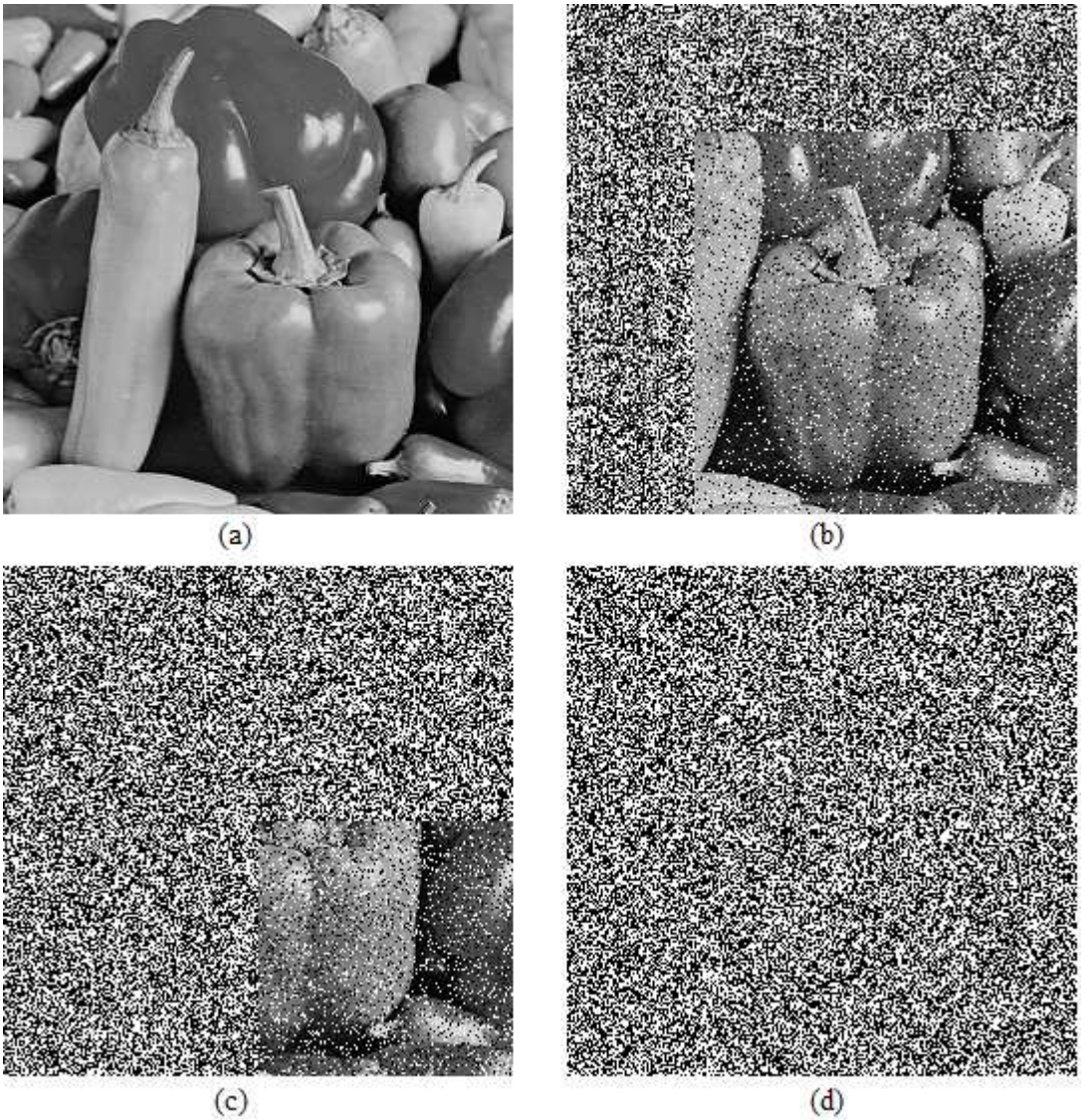


Fig. 6. a) original image, b), c), d) the encrypted images after implementing 25%, 50% and 100% of the proposed method.

4.3. Entropy

The entropy of the image can manifest the pixel gray level distribution via the image histogram. The optimum entropy is 8, showing the even distribution of the image gray level. Eq. 22 is used to determine the entropy.

$$H(s) = \sum_{i=0}^{2^M-1} P(s_i)\, log_2 \frac{1}{P(s_i)} \tag{22}$$

In which $P(s_i)$ is the occurrence probability of s_i. The entropy of the cipher images in Table 4 validates the accuracy of suggested method.

4.4. Statistical attacks

Statistical attacks are composed of correlation coefficient analysis in horizontal, vertical, and diagonal directions along with histogram analysis. In a robust encryption method, the correlation of adjacent pixels should remain minimum. Furthermore, histogram analysis demonstrates the distribution of gray level pixels which should be unvarying.

4.4.1. Correlation coefficient analysis

The correlation coefficient has been calculated based on Eq.18 as explained before. Moreover, horizontal, vertical, and diagonal correlation coefficients for 8000 pairs of neighboring pixels of cipher images based upon Eq. 18 are shown in Table 5.

Table 4. The entropy of cipher-images

	Lena	Cameraman	Peppers	Tree	Baboon	Airplane	Painter	Boat
256×256	7.9990	7.9982	7.9987	7.9988	7.9986	7.9979	7.9981	7.9986
512×512	7.9994	7.9991	7.9993	7.9990	7.9992	7.9994	7.9989	7.9993

Table 5. Correlation coefficients of cipher-images

		Lena	Cameraman	Peppers	Tree	Baboon	Airplane	Painter	Boat
256×256	Vertical	0.0059	0.0131	0.0041	0.0032	0.0036	0.0041	0.0056	0.0024
	Horizontal	0.0037	0.0095	0.0158	0.0033	0.0076	0.0101	0.0014	0.0051
	Diagonal	0.0014	0.0021	0.0075	0.0009	0.0019	0.0064	0.0011	0.0012
512×512	Vertical	0.0016	0.0025	0.0019	0.0062	0.0029	0.0007	0.0009	0.0023
	Horizontal	0.0019	0.0021	0.0019	0.0015	0.0034	0.0016	0.0007	0.0036
	Diagonal	0.0008	0.0020	0.0007	0.0012	0.0018	0.0010	0.0006	0.0009

The correlation of adjacent pixels in Lena's plain and cipher images are depicted in figure 7 vertically, horizontally and diagonally, based on 8000 matches of adjacent pixels. This figure also confirms the almost ideal correlation coefficient of the cipher image during the implemention of the proposed encryption method. The cipher image correlation and Lena's plain image correlation can be seen on the first and second rows of Fig. 7, respectively.

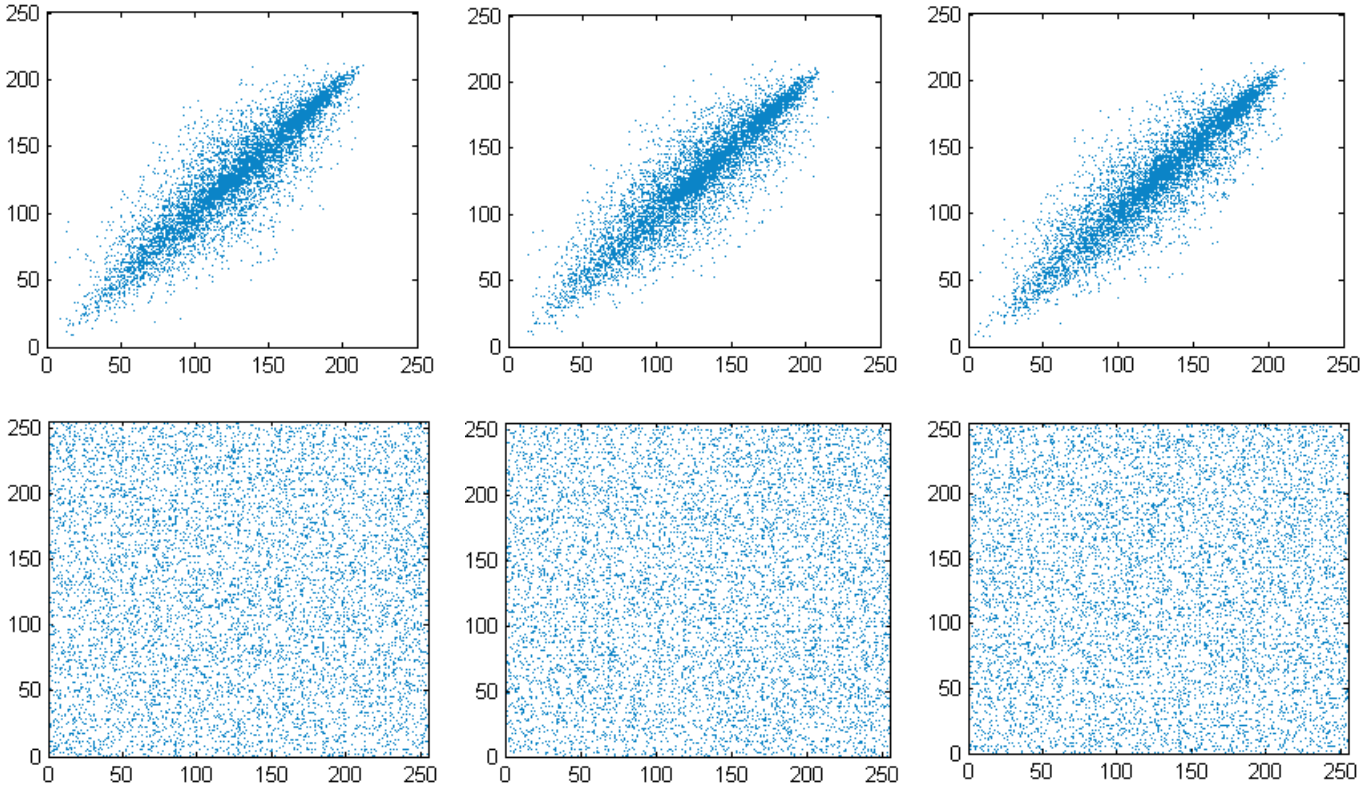


Fig. 7. Correlation plot of two neighboring pixels in different direction before (first row) and after (second row) encryption

4.4.2. Histogram analysis

An initial image has uneven histogram distribution while a robust encrypted image has uniform histogram distribution. The histograms of pepper image (256×256 and 512×512) before and after plain image encryption by proposed method are depicted in Fig.8. Therefore, it can be concluded that the proposed encryption method is robust enough in statistical attacks.

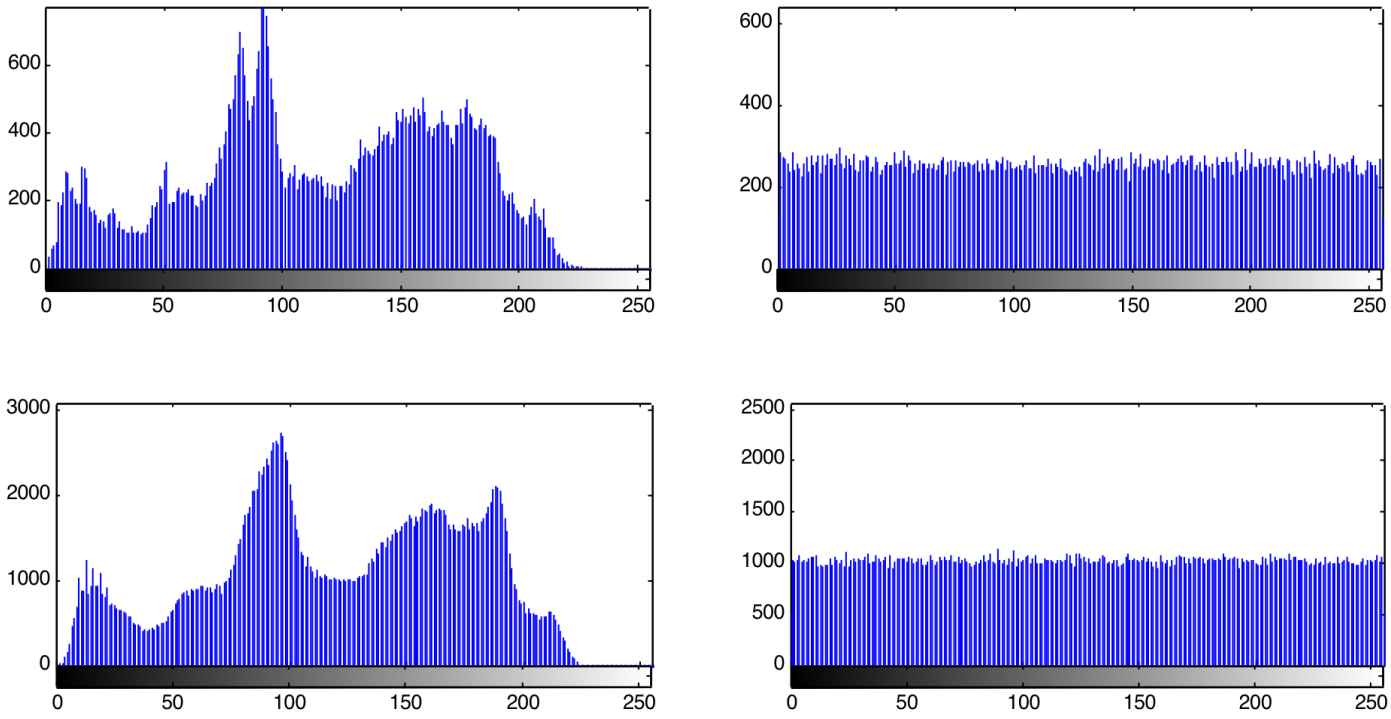


Fig. 8. From left to right in each row: Histogram of plain-image, Histogram of cipher-image

4.5. Brute-force Attack

Secret key sensitivity and the size of the secret key space are evaluated in order to elude brute-force attacks.

4.5.1. Key sensitivity analysis

The proposed method has an acceptable key sensitivity analysis results, and a trivial difference in the original image changes the cipher image. A 128-bit secret key is employed for the encryption of the baboon image (Fig. 9a, 9b, and 9c). Then the encryption continues, and a zero bit changes to one. Fig. 9d verifies the sensitivity of the proposed method to any trivial variation of the original key. The secret key analysis results of all the test images can be seen in table 6.

4.5.2. Secret key space analysis

In the proposed method, a 32 hexadecimal number produced by MD5 changes to a 128-bit key. This key space can tolerate brute-force attacks with 2^{128}value.

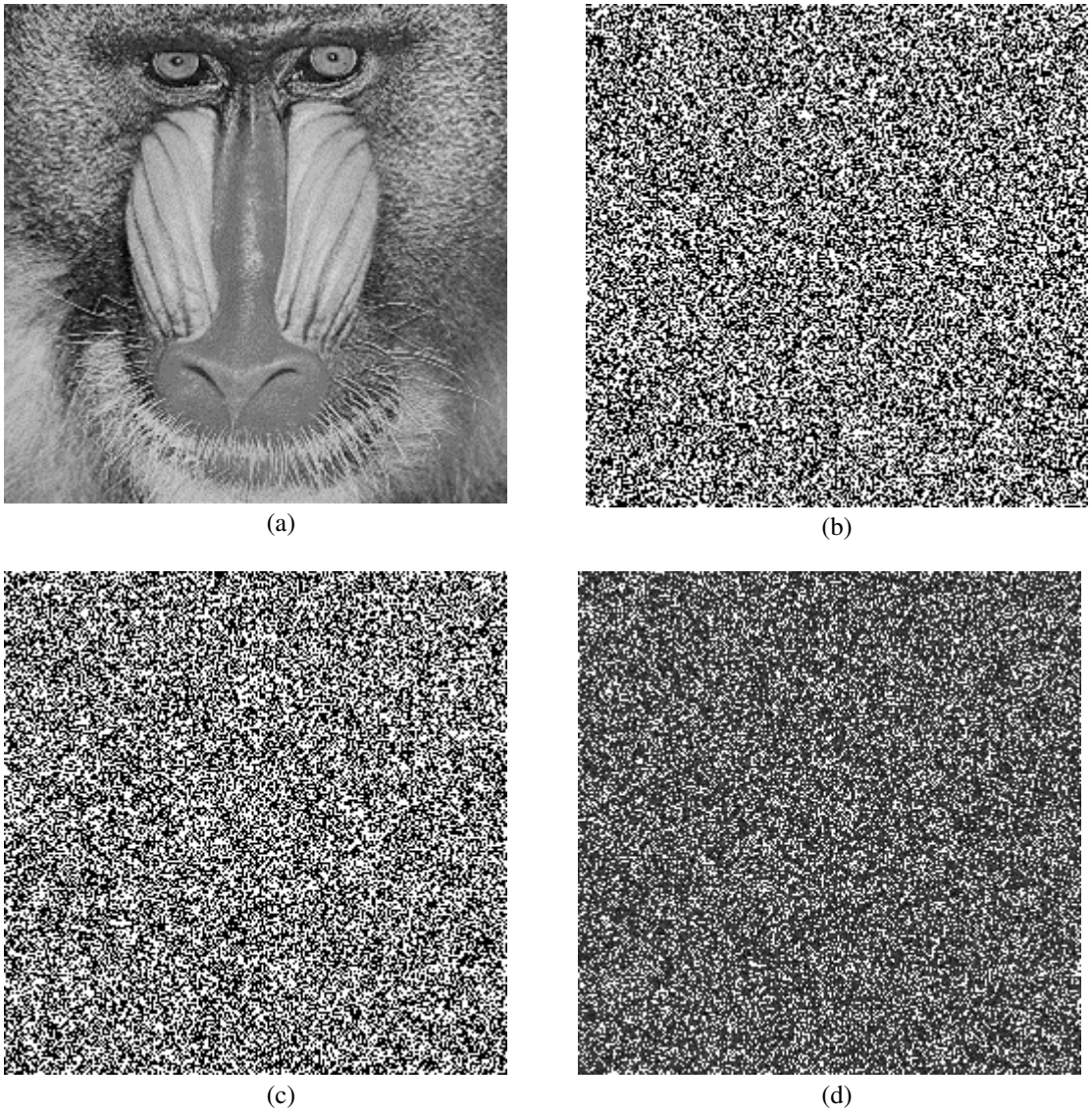


Fig. 9. a) Baboon's image b) Cipher-image with 128-bit secret key c) Cipher-image with the same key as Fig. 9b with alteration 1 bit d) difference between Figs. 9b and 9c

Table 6. Differences between two cipher-images when a 1-bit change is applied to the secret key

	Lena	Cameraman	Peppers	Tree	Baboon	Airplane	Painter	Boat
256×256	99.58 %	99.64 %	99.51 %	99.62 %	99.50 %	99.52 %	99.65 %	99.65 %
512×512	99.66 %	99.69 %	99.62 %	99.74 %	99.61 %	99.57 %	99.46 %	99.65 %

4.6. Differential Attack

An important criterion to evaluate the success rate of the encryption method is differential attack. The main aim of differential attack is to assess whether a minor alteration in the plain image can create a significantly notable variation in the cipher-image or not.

Therefore, unified average changing intensity (UACI) and number of pixels change rate (NPCR) are utilized for determining distinctive attacks as in Eq. 23 and Eq. 24 [3]:

$$NPCR = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} D(i,j)}{M \times N} \times 100\% \quad (23)$$

$$UACI = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} |C_1(i,j) - C_2(i,j)|}{255 \times M \times N} \times 100\% \quad (24)$$

In which $D(i,j)$ is calculated as stated in Eq. 25:

$$D(i,j) = \begin{cases} 0 if\, C_1(i,j) = C_2(i,j) \\ 1 if\, C_1(i,j) \neq C_2(i,j) \end{cases} \quad (25)$$

$C_1$and C_2 are two cipher images whose similar plain-images have solely one-bit difference utilizing the same original key. Table 7 demonstrates the NPCR and UACI of $C_1$and C_2 with the suggested encryption method based on chosen test images. Table 7 shows the sensitivity of the proposed method to minor change in plain images.

Table 7. NPCR and UACI of two cipher images ($C_1$and C_2) whose corresponding plain-images have only one-bit difference

		Lena	Cameraman	Peppers	Tree	Baboon	Airplane	Painter	Boat
NPCR	256×256	0.995026	0.993752	0.994294	0.995185	0.994962	0.995274	0.993031	0.995713
	512×512	0.996219	0.995260	0.995116	0.996107	0.996006	0.996479	0.994920	0.995925
UACI	256×256	0.333398	0.337982	0.333160	0.329381	0.337618	0.334984	0.326956	0.333604
	512×512	0.334162	0.339058	0.338336	0.332574	0.339572	0.339007	0.336822	0.337487

4.7. Comparaison

The performance of this method is compared with GA-DNA [23], CA-DNA[24], BST-DNA[2] and DNA-RNA[15]. For the basis of their encryption is on chaotic function and DNA. The statistical results of qualitative criteria for these five methods are provided in Table 8.

4.7.1. Statistical criteria

The efficacy of the proposed method is evaluated by Lena image of 256×256 and Baboon image of 512×512. Table 8 confirms that the proposed method results are significantly better than the results of CA-DNA and BST-DNA methods. However, GA-DNA and the proposed method have similar results because of their genetic algorithm, an iterative evolutionary process with prolonged performance duration which makes them unapplicable to immediate applications. On the contrary, the obtained time for the proposed method is almost acceptable.

4.7.2. Quality measure

In this section, an image of a Boat with dimensions of 256×256 and an image of a plane with dimensions of 512×512 are employed to compare qualitative criteria of the proposed method with four methods of GA-DNA[23] ،CA-DNA[24] ،BST-DNA [2]،DNA-RNA[15]. The main qualitative criteria considered are Mean square error (MSE)[25] ,Peak signal-to-noise ratio (PSNR)[25] ، *Normalized Absolute error (NAE)[26]* ،*Structural content (SC)[26]* ،*Root Mean Square Error(RMSE).*

Table 8. Comparison of the proposed method and the related works

		Entropy	Correlation Coefficient			NPCR	UACI	Time
			Vertical	Horizontal	Diagonal			
256×256	GA-DNA[23]	7.9990	0.0049	0.0051	0.0015	0.993808	0.336974	16931
	CA-DNA[24]	7.9973	0.0107	0.0119	0.0044	0.989195	0.326002	450
	BST-DNA[2]	7.9981	0.0029	0.0093	0.0021	0.993826	0.335714	941
	DNA-RNA[15]	7.9983	0.0068	0.0082	0.0037	0.992403	0.332836	953
	Proposed method	7.9986	0.0036	0.0076	0.0019	0.994962	0.337618	819
512×512	GA-DNA[23]	7.9994	0.0011	0.0022	0.0007	0.996485	0.335218	67195
	CA-DNA[24]	7.9991	0.0039	0.0025	0.0013	0.993557	0.329485	1827
	BST-DNA[2]	7.9989	0.0012	0.0031	0.0011	0.995209	0.340083	3792
	DNA-RNA[15]	7.9991	0.0028	0.0023	0.0017	0.993862	0.327291	3837
	Proposed method	7.9994	0.0016	0.0019	0.0008	0.996219	0.334162	3266

The mean squared error (MSE) of an estimator calculates the average of the squares of the errors— the average squared difference between the calculated values and the actual value (Eq. 26). Corrupting noise affects PSNR representation. PSNR describes the ratio amongst the corrupting noise and strength of signal (Eq. 27). NAE, normalized absolute error, is calculated by Eq. 28. The similarity between two images can be assessed based on structural content (SC) (Eq. 29). Moreover, this measurement is based on the correlation of neighboring pixels of an image. A high rate of SC, an indicator of the quality of an image indicates the poor quality of the picture. RMSE represents "Root Mean Square Error", evaluating the "square root of mean of the square of all the errors" (Eq. 30). It is employed regularly for great wide span in numerical forecasts.

$$MSE = \frac{1}{M \times N} \sum_{y=1}^{M} \sum_{X=1}^{N} [O(x,y) - E(x,y)]^2 \tag{26}$$

$$PNSR = 10 \times log_{10} \frac{MAX_1^2}{\sqrt{MSE}} \tag{27}$$

$$NAE = \frac{\sum_{y=1}^{M} \sum_{X=1}^{N} [O(x,y) - E(x,y)]}{\sum_{y=1}^{M} \sum_{X=1}^{N} |O(x,y)|} \tag{28}$$

$$SC = \frac{\sum_{y=1}^{M} \sum_{X=1}^{N} [O(x,y)]^2}{\sum_{y=1}^{M} \sum_{X=1}^{N} [E(x,y)]^2} \tag{29}$$

$$RMSE = \sqrt{\frac{\sum_{y=1}^{M}\sum_{X=1}^{N}[O(x,y) - E(x,y)]^2}{M \times N}} \quad (30)$$

Where $O(x, y)$is the original image, $E(x, y)$ is the encrypted image having dimension (M, N). The highest valued figure of Mean Square Error can be considered as the better starting rate.

All the obtained result for above indexes are listed in Table 9. These results are also another demonstration of superiority of the proposed method.

Table 9. Quality measure to compare the proposed method with state-of-art methods

		MSE	PSNR	NAE	SC	RMSE
256×256	GA-DNA[23]	21160	4.8756	126.12	1.1192	103.71
	CA-DNA[24]	21039	4.9006	125.95	1.2139	98.42
	BST-DNA[2]	20481	5.0173	122.41	1.1581	100.63
	DNA-RNA[15]	19726	5.1804	121.04	1.2057	99.48
	Proposed method	21804	4.7454	127.83	1.0395	103.55
512×512	GA-DNA[23]	21648	4.7766	127.22	1.0291	104.26
	CA-DNA[24]	21391	4.8285	126.79	1.0538	100.51
	BST-DNA[2]	21007	4.9072	124.87	1.1087	101.10
	DNA-RNA[15]	20188	5.0799	123.73	1.1525	100.23
	Proposed method	21606	4.7851	127.30	1.0228	103.90

5. Conclusion

This article proposed a quick and safe method for image encryption. This method employed a combination of LSFR and bit-level shift in permutation phase and a combination of DNA and Tinkerbell in diffusion phase. The main reason for high security of the proposed method is the combination of LSFR, DNA and Tinkerbell chaotic function. However, there are two reasons for the respectable speed of the proposed method: first, synchronized permutation and diffusion phases; second the encryption of half of image pixels. The obtained results in section four confirmed high resistance of the proposed method against the prevailing attacks.

References

[1] H.M. Ghadirli, A. Nodehi, R. Enayatifar, An overview of encryption algorithms in color images, Signal Processing, 164 (2019) 163-185.

[2] H. Nematzadeh, R. Enayatifar, M. Yadollahi, M. Lee, G. Jeong, Binary search tree image encryption with DNA, Optik, 202 (2020) 163505.

[3] R. Enayatifar, F.G. Guimarães, P. Siarry, Index-based permutation-diffusion in multiple-image encryption using DNA sequence, Optics and Lasers in Engineering, 115 (2019) 131-140.

[4] N. Louzzani, A. Boukabou, H. Bahi, A. Boussayoud, A novel chaos based generating function of the Chebyshev polynomials and its applications in image encryption, Chaos, Solitons & Fractals, 151 (2021) 111315.

[5] Y. Liu, Z. Jiang, X. Xu, F. Zhang, J. Xu, Optical image encryption algorithm based on hyper-chaos and public-key cryptography, Optics & Laser Technology, 127 (2020) 106171.
[6] A.H. Abdullah, R. Enayatifar, M. Lee, A hybrid genetic algorithm and chaotic function model for image encryption, AEU - International Journal of Electronics and Communications, 66 (2012) 806-816.
[7] R. Enayatifar, A.H. Abdullah, M. Lee, A weighted discrete imperialist competitive algorithm (WDICA) combined with chaotic map for image encryption, Optics and Lasers in Engineering, 51 (2013) 1066-1077.
[8] X. Liu, Y. Wu, H. Zhang, J. Wu, L. Zhang, Quaternion discrete fractional Krawtchouk transform and its application in color image encryption and watermarking, Signal Processing, 189 (2021) 108275.
[9] T. Zhao, L. Yuan, Y. Chi, Image encryption using linear weighted fractional-order transform, Journal of Visual Communication and Image Representation, 77 (2021) 103098.
[10] Q. Cun, X. Tong, Z. Wang, M. Zhang, Selective image encryption method based on dynamic DNA coding and new chaotic map, Optik, 243 (2021) 167286.
[11] X. Wang, Y. Su, Image encryption based on compressed sensing and DNA encoding, Signal Processing: Image Communication, 95 (2021) 116246.
[12] E. Yavuz, A new parallel processing architecture for accelerating image encryption based on chaos, Journal of Information Security and Applications, 63 (2021) 103056.
[13] Y. Zhang, The fast image encryption algorithm based on lifting scheme and chaos, Information Sciences, 520 (2020) 177-194.
[14] J.S. Muthu, P. Murali, Review of Chaos Detection Techniques Performed on Chaotic Maps and Systems in Image Encryption, SN Computer Science, 2 (2021) 392.
[15] M. Yadollahi, R. Enayatifar, H. Nematzadeh, M. Lee, J.-Y. Choi, A novel image security technique based on nucleic acid concepts, Journal of Information Security and Applications, 53 (2020) 102505.
[16] S. Zhang, L. Liu, A novel image encryption algorithm based on SPWLCM and DNA coding, Mathematics and Computers in Simulation, 190 (2021) 723-744.
[17] D. Wei, M. Jiang, A fast image encryption algorithm based on parallel compressive sensing and DNA sequence, Optik, 238 (2021) 166748.
[18] R. Enayatifar, H.J. Sadaei, A.H. Abdullah, M. Lee, I.F. Isnin, A novel chaotic based image encryption using a hybrid model of deoxyribonucleic acid and cellular automata, Optics and Lasers in Engineering, 71 (2015) 33-41.
[19] K.B. Sudeepa, G. Aithal, V. Rajinikanth, S.C. Satapathy, Genetic algorithm based key sequence generation for cipher system, Pattern Recognition Letters, 133 (2020) 341-348.
[20] Z. Jiang, Y. Zhan, D. Chen, Y. Wang, Two methods of directly constructing probabilistic public-key encryption primitives based on third-order LFSR sequences, Applied Mathematics and Computation, 171 (2005) 900-911.
[21] V. Raj, S. Janakiraman, S. Rajagopalan, R. Amirtharajan, Security analysis of reversible logic cryptography design with LFSR key on 32-bit microcontroller, Microprocessors and Microsystems, 84 (2021) 104265.
[22] S. Karunamurthi, V. Krishnasamy Natarajan, VLSI implementation of reversible logic gates cryptography with LFSR key, Microprocessors and Microsystems, 69 (2019) 68-78.
[23] R. Enayatifar, A.H. Abdullah, I.F. Isnin, Chaos-based image encryption using a hybrid genetic algorithm and a DNA sequence, Optics and Lasers in Engineering, 56 (2014) 83-93.
[24] A. Babaei, H. Motameni, R. Enayatifar, A new permutation-diffusion-based image encryption technique using cellular automata and DNA sequence, Optik, 203 (2020) 164000.
[25] A. Shokouh Saljoughi, H. Mirvaziri, A new method for image encryption by 3D chaotic map, Pattern Analysis and Applications, 22 (2019) 243-257.
[26] T. ul Haq, T. Shah, Algebra-chaos amalgam and DNA transform based multiple digital image encryption, Journal of Information Security and Applications, 54 (2020) 102592.