Color images resistant encryption based on hyperchaotic functions and DNA sequences

Mohammadreza ZARIF[1], Hamid reza GHAFFARY[2], Masoud DAVOUDI[3]

[1]MSc in Computer Systems Architecture, Department of Computer Engineering, Islamic Azad University, Ferdows Branch, Iran

[2] Department of Computer Engineering, Islamic Azad University, Ferdows Branch, Iran

[3]MSc in Artificial Intelligence, Department of Computer Engineering, Islamic Azad University, Ferdows Branch, Iran

Corresponding Author: Mohammadreza ZARIF

M.R.Zarif@ferdowsiau.ac.ir, H.R.Ghaffary@ferdowsiau.ac.ir, M.Davoudi@ferdowsiau.ac.ir

Abstract: One of the effective ways of preventing unauthorized access to sensitive information such as medical and military images, is encryption. DNA encryption is one of the newly-emerged fast paced technologies which works based on DNA calculation concept and could be used to store and transfer data. It can be demonstrated that its advantages are fast speed, its minimum storage, and power usage in the context of DNA calculations. This article presents a new color image encryption concept. First, the color image is divided into three color factors of red, green and blue, then we disassemble pixel places of the color factors using Lorenz hyperchaotic Model, to run the pixel connections together. Therefore, each of the color factors of red, green, blue, transforms into DNA sequences using DNA encryption rules. Thus, the Chen hyperchaotic systems are repeated, three chaotic sequences emerge which transform into DNA sequences. Chaotic sequences were used for DNA sequence encryption. Security analyzes and experimental results show that the proposed method with a large key space, is resistant to various attacks. The correlation between the pixels next to each other is greatly decreased in the encrypted image and the amount of entropy is 7.9991.

Keywords: image encryption, DNA sequences, Lorenz hyperchaotic Model, Chen hyperchaotic model.

1. Introduction

Image encryption architecture based on chaos was first presented in 1998. According to this structure, the two steps of chaos and distribution are performed in one encrypting step (Singh & Sinha, 2008). In the first step, almost all the pixels get relocated, as it causes high degrees of correlation of pixels next to one another. After this step, the basis of the elements of the image, bits or pixel amounts, are distributed equally. Chaos system procedure has different features such as high sensitivity to the initial conditions, certainty, and ergodicity. Chaos sequences are made by chaos mappings which are random sequences and these structures are very complicated and are hard to analyze and predict. A chaos system can be used properly as a randomizing source in the procedure of chaos and distribution (Shujun et al., 2005; Shannon, 1949; Pisarchik & Zanin, 2008). Today there are several new methods for differing image encrypting concepts based on DNA. For instance, several image encrypting concepts have been created by the usage of DNA coding and chaos mapping whose goal is to increase efficiency and the security of encryption concepts by a certain procedure.

For the first time, Adleman published his findings in DNA calculations in 1994 [Adleman, 1998]. DNA calculations studies made DNA encryption a new research field which uses DNA as a data transporter (Xiao et al., 2006). In (Zhang & Fu, 2012), an innovative image encryption based on DNA calculations has been presented and DNA sequence has been used to code the data and using XOR logical procedures, the image was encrypted. In (Ailenberg & Rotstein, 2009), an optimized version of Hoffman DNA coding is presented. Zhang et al. (2009) presented a new image-encrypting algorithm based on the procedure of collecting DNA sequences. Liu et al. (2012) proposed an RGB image-encrypting algorithm based on DNA coding along with chaos mapping. The main idea of the presented algorithm was to utilize DNA sum to disassemble R, G, and B pixel factor amounts and encryption of the disassembled image took place next. The presented image encryption algorithms in chaos and hyperchaotic systems have been mixed with DNA procedures (Wei et al., 2012; Zhang & Fu, 2012; Zhang & Wei, 2013).

https://doi.org/10.33436/v30i3y202006

Liu et al., (2014) have analyzed the security of the proposed algorithm as a new image-encrypting algorithm based on the procedure of collecting DNA sequences. Liu et al. (2012) found two security problems: (1) The hidden keys equal to encrypting algorithm could easily be rebuilt using only a pair of identified texts or encryption text; (2) The results of encryptions are not sensitive to main images` transformations. Enayatifar and his assistants have designed an image encryption algorithm by the usage of DNA mask and genetic algorithm mixture, and they've used Logistic encryption as the key in production source of the primary population for genetic algorithm (Enayatifar et al, 2015). Enayatifar et al. (2015) developed their work and presented the new encryption concept based on the combination of cellular automata and Tinker Bell chaotic mapping. All the cellular automata rules have been used to produce semi-random numbers in this concept (Enayatifar et al, 2015). As color pictures have more data compared to gray images, they have been noticed more. In (Leyuan Wang et al, 2016) an image encryption algorithm is presented which is based on Chen and Lorenz systems. In (Hongjun Liu and Abdurahman Kadir, 2015) a color image encryption model was designed and circular shifts were used to disassemble the image. Chang'e Dong (2013) presents a color image encryption method based on chaos mapping combinations. In (Xingyuan Wang et al, 2016), a color image encryption method has been presented using the structures of intermittent chaos mappings. In (Xingyuan Wan, Hui-li Zhang, 2015), a color image encryption method was presented which was based on non-homogeneous and perturbation bitmaps permutation. In (Wu et al, 2015), a color image encryption model is presented which is based on DNA sequences and improved one-dimensional chaos mappings. In (Annaby et al, 2018), a Color image encryption model is presented which is using random transforms, phase retrieval, chaotic maps, and diffusion. In (Valandara et al, 2019), a fast color image encryption technique is presented which is based on three-dimensional chaotic maps. In (Xuejing & Zihui, 2020) a new color image encryption model is presented which is based on DNA encoding and spatiotemporal chaotic system.

In this article, first, we divide the color image into red, green and blue actors. We use Lorenz chaos system to relocate the main image pixels. Then we transform the R, G, and B factors into binaries and by the usage of DNA coding rules, we actually implement a sequence of DNAs on the binaries. Chen hyperchaotic system has been used to produce the three chaos sequences and use them when encrypting. After producing the chaos sequences, we also transform the intended sequences into a DNA sequence and afterwards, by using XOR on the color image DNA sequences and chaos sequences, we encrypt the image. The rest of this article is organized as follows. Section 2 describes the basic theory of the presented algorithm. Section 3 defines the presented algorithm. Some of the security analyzes are presented in section 4. Finally, the conclusion is set forth in Section 5.

2. Materials and Methods

2.1. DNA Encoding and Decoding for The Image

A DNA strain (sequence) consists of four acid-alkaline nucleotides: A (adenine), C (c tevzin), G (guanine), T (thymine) as T and A are each other's' supplement and C and G the same. As they are each other's supplement in 0 and 1 binaries, then they are a supplement for each other in 11 and 00 and 10 and 01. For 8 bits of gray images, each pixel could be shown as a DNA sequence with the length of 4. For instance, if the amount of the first pixel of the primary image were 173, we transform it to the binary sequence of (10101101), with the usage of DNA coding rules, we can receive the sequence as (GGTC). As 11, 10, 01, 00 respectively show T=11, G=10, C=01, A=00. Table (1) shows the encryption rules of DNA sequences. For instance, if the amount of a pixel with gray area was 157, the binary value of it would be (10011101), and the DNA code of this binary amount for each rule of 8 rules in Table (1) is as follows:

Rule 1 (CGTG), Rule 2 (GCTC), Rule 3 (CGAG), Rule 4 (GCAC), Rule 5 (ATGT), Rule 6 (TAGA), Rule 7 (ATCT), Rule 8 (TACA).

The fast development of DNA calculations presented by researchers and lead to biological operators and algebraic operators based on DNA sequences like the XOR operator. The XOR operator is performed for the DNA sequences by XORing in the binary system which is shown in Tables (1, 2).

Table 1. DNA encoding and decoding rules (Enayatifar, Abdullah & Isnin, 2014)

	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. A type of XOR operator for DNA sequence (Enayatifar, Abdullah & Isnin, 2014)

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

2.2. The Lorenz Chaos System

The Lorenz Chaos System is expressed in relation number 1. In relation 1, a, b, and c, are system parameters, if these parameters have the amounts of a=10, b=8.3, c=28, the Lorenz system is positioned in the chaotic state and can create 3 chaos sequences (Zhang & Wei, 2013; Wang, Song & Liu, 2016). The Lorenz attractor is shown in fig. 1.

$$\begin{cases} \dot{x} = a\,(y - x); \\ \dot{y} = cx - xz - y; \\ \dot{z} = xy - bz; \end{cases} \tag{1}$$

2.3. Chen Hyperchaotic System

Due to the features of the hyperchaotic functions such as sensitivity to initial conditions and system parameters, pseudo-random, non-permanent, and definitive properties, the importance of these functions in encrypting is clear. The hyperchaotic functions are more secure with regard to the large space of their key spaces than the normal functions. The Chen's hyperchaotic system is described in relation (2):

$$\begin{cases} \dot{x} = a\,(y - x); \\ \dot{y} = (c - a)x - xz + cy; \\ \dot{z} = xy - bz; \end{cases} \tag{2}$$

In relation (2), **a**, **b** and **c** are system parameters, if these parameters own the amounts of **a**=35, **b**=3, **c**$\in$28.4 (Wang et al., 2016) hyperchaotic system of Chen is positioned in the chaotic state and can cause the creation of three chaos sequences. The initial amounts can be regarded as the keys. The Chen hyperchaotic attractor is shown in figure 2.

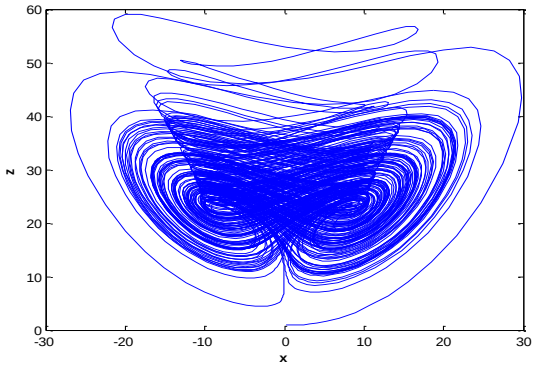


Figure 1. The Lorenz system attractor

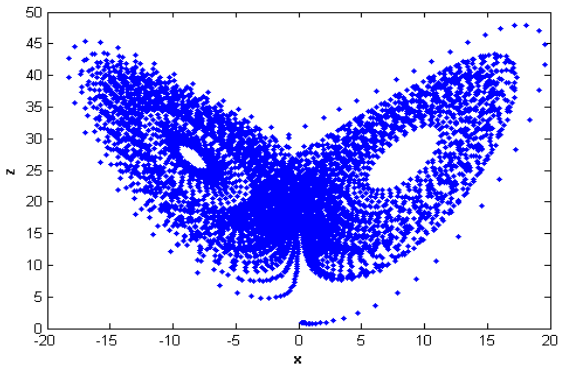


Figure 2. The Chen system attractor

3. The Proposed Method

3.1. Generate an Array for Permutations of Pixels

Permutation algorithms play an important role in image encryption. As the goal in the first step of the encryption system is to relocate all the pixels and an auxiliary step is released for the operation. Concerning the permutation algorithms based on order, only a few of the numbers actually need arranging. To dissolve the pixels using column and row permutations, we need to base our permutation procedure on arrays. Using Lorenz system, three chaos sequences of X, Y, and Z with initial values with the length of M and three chaos sequences of R, S, and T, with the initial values and length of N are created.

First, the chaos sequences are arranged as follows:

$$\begin{cases} [lx, fx] = sort(x); \\ [ly, fy] = sort(y); \\ [lz, fz] = sort(z); \end{cases} \tag{3}$$

$$\begin{cases} [lr, fr] = sort(r); \\ [ls, fs] = sort(s); \\ [lt, ft] = sort(t); \end{cases} \tag{4}$$

As: [•,•] = sort(•) is a list of function sequences, Lx is the new sequence after rearranging x ascending and F_x is the index value of Lx. Other sequences are sorted out similarly. Sorting x is shown in figure 3.

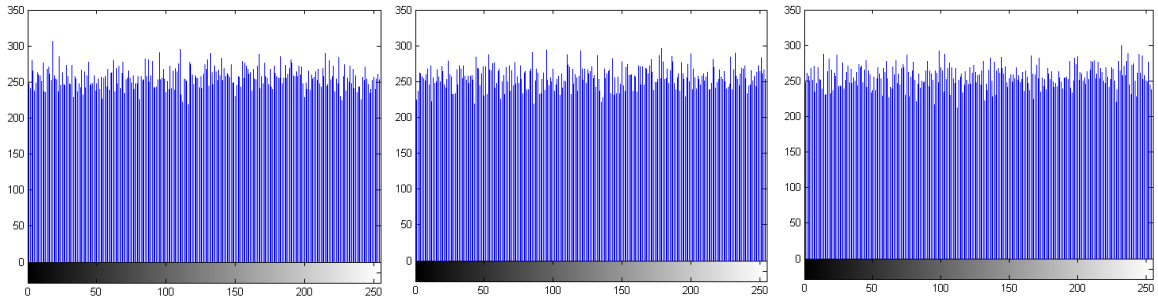


Figure 3. Chaos sequences` distribution

3.2. Producing Chaos Sequences with Chen System

To have a safe and secure encryption algorithm, we can consider Chen hyperchaotic function initial values as keys. Chen hyperchaotic function is used to create three chaos sequences. If the image size was M × N × 3, we can receive three chaos sequences of x_i, y_i, z_i, in each repetition, for each factor of Red, Green and Blue as x_1, y_1, z_1, i =1,2,... M × N and therefore, sequences of S_x, S_y,

$S_z \in [0, 255]$ which are extracted from relations 5 and 6.

$$\begin{cases} S_x = \{ \acute{x}_1, \acute{x}_2, \ldots, \acute{x}_{MxN} \} \\ S_y = \{ \acute{y}_1, \acute{y}_2, \ldots, \acute{y}_{MxN} \} \\ S_z = \{ \acute{z}_1, \acute{z}_2, \ldots, \acute{z}_{MxN} \} \end{cases} \quad (5)$$

As

$$\begin{cases} \acute{x}_i = (abs(x_i) - fix(abs(x_i))) \times 10^{14} \bmod 256 \\ \acute{y}_i = (abs(y_i) - fix(abs(y_i))) \times 10^{14} \bmod 256 \\ \acute{z}_i = (abs(z_i) - fix(abs(z_i))) \times 10^{14} \bmod 256 \end{cases} \quad (6)$$

We generate three sequential S_x, S_y, S_z sequels. The distribution of the mappings of these sequences using a histogram is shown in Fig. 4. It can be seen that chaos sequences have homogenous distribution in between [0,255].

X:	0.3000	3.6000	89.8000	207.7000	110.9000	234.2000	38.8000	135.1000	10.3000	111.1999
LX:	0.0186	0.0610	0.3000	2.2631	3.6000	4.1600	5.7564	5.9922	6.3018	6.4408
FX:	114	55	1	37	2	251	120	126	65	75

Figure 4. Sorting random sequences

3.3. Encryption

The overall structure of encryption is shown in fig. 5. The steps of the proposed method are as follows.

Step 1: The input is a color image called **P** (M, N, 3) so that M and N respectively represent the rows and columns of the image dimensions.

Step 2: The color image is divided into three components: red, green, and blue. And we can receive three components P_R, P_G, P_B as shown in relation 7.

$$\begin{cases} P_R = \{r_1, r_2, \ldots, r_{M \times N}\} \\ P_G = \{g_1, g_2, \ldots, g_{M \times N}\} \\ P_B = \{b_1, b_2, \ldots, b_{M \times N}\} \end{cases} \quad (7)$$

In order that $\mathbf{b_i}$, $\mathbf{g_i}$, $\mathbf{r_i}$ are the amounts of the $\boldsymbol{i}^{th}$ pixel of red, green and blue factors between 2 and 255.

Step 3: By selecting and combining sorted sequences in 3 and 4 relations, we use the place of main P image pixels in the order of relation number 8 for the rows and 9 for the columns of each red, green, blue factors, to have pixel permutations.

$$\begin{cases} P_R^R(i, j) = P_R(Fx(i), Fr(j)); \\ P_G^R(i, j) = P_G(Fy(i), Fs(j)); \\ P_B^R(i, j) = P_B(Fz(i), Ft(j)); \end{cases} \quad (8)$$

In an order in which: $\boldsymbol{i}$ = 1, 2,. . . ,M; $\boldsymbol{j}$ = 1,2,. . .,N

$$\begin{cases} P_R^{RC}(i, j) = P_R^R(Fr(i), Fx(j)); \\ P_G^{RC}(i, j) = P_G^R(Fs(i), Fy(j)); \\ P_B^{RC}(i, j) = P_B^R(Ft(i), Fz(j)); \end{cases} \quad (9)$$

In an order in which: $\boldsymbol{i}$ = 1, 2,. . . ,M; $\boldsymbol{j}$ = 1,2,. . .,N

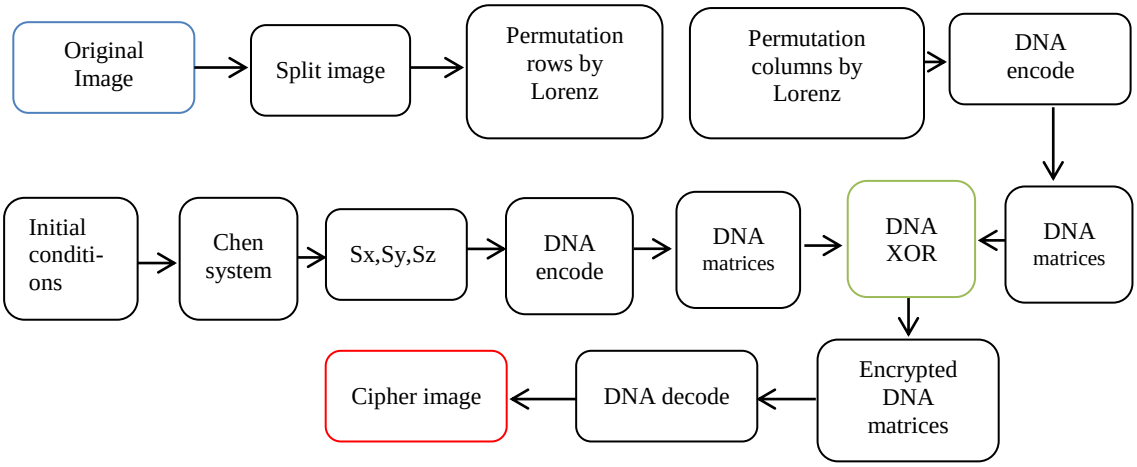


Figure 5. The overall structure of the proposed plan

Step 4: In order for the proposed concept to withstand the original text attack, the initial values of the Chen system X_0 , Y_0 , Z_0 are updated using relation number 10.

$$\begin{aligned} &\text{SummImg} = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} (P_R(i,j) + P_G(i,j) + P_B(i,j)) \right) \\ &X_0 = \text{mod}((X_0 + \text{SummImg}),1) \\ &Y_0 = \text{mod}((Y_0 + \text{SummImg}),1) \\ &Z_0 = \text{mod}((Z_0 + \text{SummImg}),1) \end{aligned} \quad (10)$$

So as to prevent irresistibility effect, the Chen permutation gets repeated 200 times as the produced numbers are not considered, therefore permutation repetition continues and three sequences of $\mathbf{S_x}$, $\mathbf{S_y}$, $\mathbf{S_z}$, are created with the exact size as each color image factor, as indicated in equations 2, 5, and 6.

Step 5: We transform the amount values present in the main image, $P_{R,G,B}$ and transform three sequences of $\mathbf{S_x}$, $\mathbf{S_y}$, $\mathbf{S_z}$, into binary format. We receive $\mathbf{P_R}$(m, n×8), $\mathbf{P_G}$(m, n×8), $\mathbf{P_B}$(m, n×8) و $\mathbf{S_x}$(m, n×8), $\mathbf{S_y}$(m, n×8), $\mathbf{S_z}$(m, n×8), therefore, based on the rules mentioned in table 1, we begin the transformation of binary sequence to DNA sequence for binary values of P and S and receive the matrixes of

$$P_{R,G,B}^{Encoding\_DNA}(m, n \times 4) \text{ و } S_{x,y,z}^{Encoding\_DNA}(m, n \times 4).$$

Step 6: We use XOR operator to encode and implement the XOR action on all of the DNA sequences like those in Table 2 as follows.

$S_{x,y,z}^{Encoding\_DNA}$ are used to encode the red, green, and blue factors of image $P_{R,G,B}^{Encoding\_DNA}$, and respectively, receive $c_R \in C_R$, $c_G \in C_G$, $c_B \in C_B$, using equation 11.

$$\begin{cases} c_{R(i,j)} = S_{x(i,j)}^{Encoded\_DNA} \oplus P_{R(i,j)}^{Encoded\_DNA} \\ c_{G(i,j)} = S_{y(i,j)}^{Encoded_{DNA}} \oplus P_{G(i,j)}^{Encoded_{DNA}} \\ c_{B(i,j)} = S_{z(i,j)}^{Encoded_{DNA}} \oplus P_{B(i,j)}^{Encoded_{DNA}} \end{cases} \quad (11)$$

$$i = 1, 2, ..., m, \quad j = 1, 2, ..., n \times 4$$

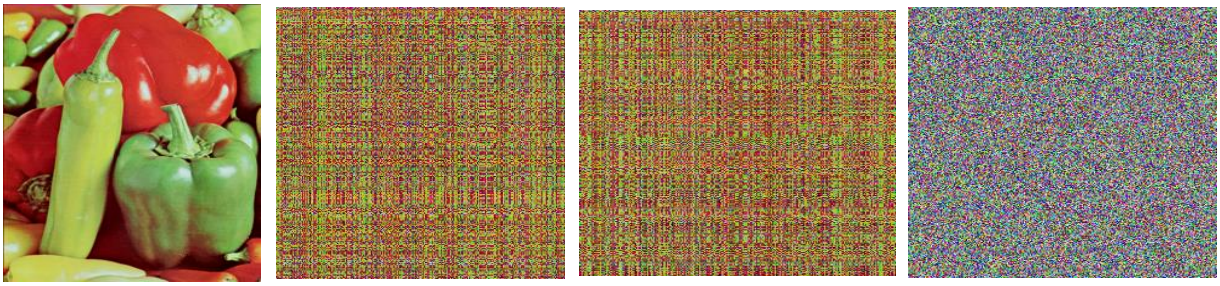
In an order in which, the sign $\oplus$, shows DNA XOR operation.

Step 7: After implementing the XOR operator on all of the DNA sequences in the above step, we turn the DNA sequences to binary sequences by using decoding rules of table.1 and then, we move on to decimal basis. We combine C_R, C_G, C_B together and receive the C coded image.

The above steps are presented to encrypt the image; it is obvious that for decoding, the above steps need to be performed vice versa.

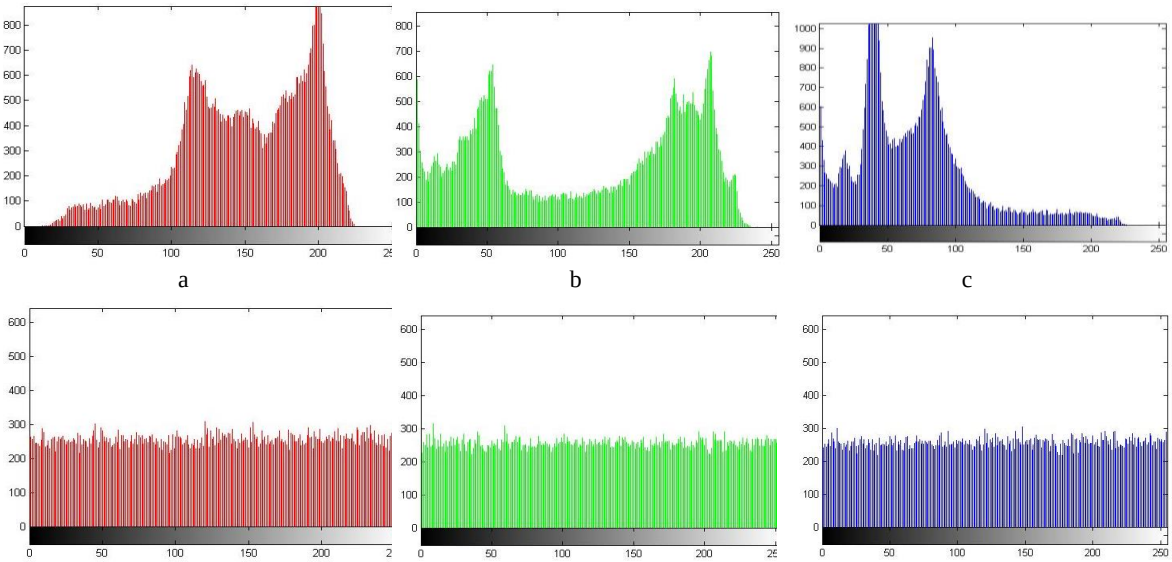
4. Experimental Results

In this section, different tests implemented to prove the proposed algorithm will be analyzed. We have used a “peppers” image for the test results with the size of 256×256×3. The results of permutations and encryption of the “peppers” image are shown in Figure 6.



a b c d

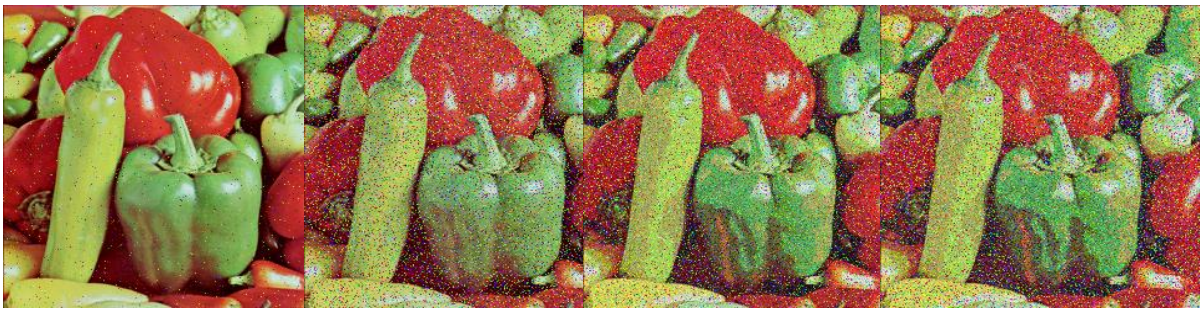
Figure 6. The main image and encryption results, (a): The “peppers” image, (b): Image permutations based on rows(c): Image permutations based on columns, (d): Encrypted image in proposed format



a b c

d e f

Figue 7. Histogram comparison of "peppers" before and after the encryption (a, b, c): In order, histogram of the red, green, and blue factors before image encryption (d, e, f): In order, histogram of the red, green, and blue factors after image encryption



a b c d

Figure 8. The results of decoded image by implementing differing noises. (a, b): by salt & pepper noise, (c, d): by Gaussian noise. (a) d=0.02, (b) d=0.2, (c) m=0, v=0.005, (d) m=0, v=0.01

4.1. Histogram Analysis

Histogram analysis describes how to distribute pixels in the image by drawing the number of observations of each intensity of light. Equal distribution of the image histogram can be a good indicator of the quality of the encryption method. In Figure 7, a, b, and c demonstrate the histogram of the red, green, and blue factors of the color image before encryption and the d, e, and f images are histograms of these factors after encryption. As it is clear in Figure 7, the histogram of the encrypted image is monotone. Therefore it does not give any information about gray value distribution to the attacker.

4.2. Resistant to Noise

There are always different types of dense noises transferring through physical channels, hence, a good encryption system must be designed in such a way so as to be resistant against noise to some extent (Liu & Kadir, 2015; Dong, 2013).

In the implemented test, we add “Salt & Pepper” noise with different ‘d’ intensities and the "Gaussian white" with an average of ‘m’ and ‘v’ variances accordingly to the encrypted image of “Peppers”. The encryption results in Figure 8 show that after adding different noises the encrypted image is still visible.

4.3. Checking the Correlation Between Adjacent Pixels

In an image data, each pixel strongly correlates with its neighboring pixels. An ideal algorithm must produce encrypted images with a low correlation of pixels with each other. The following equations are used to study the correlation between two neighboring pixels in horizontal, vertical and diagonal directions.

$$E(x)=\frac{1}{N}\sum_{i=1}^{N}x_i,\quad D(x)=\frac{1}{N}\sum_{i=1}^{N}(x_i-E(x))^2$$
$$Cov(x,y)=\frac{1}{N}\sum_{i=1}^{N}(x_i-E(x))(y_i-E(y)) \tag{12}$$
$$r_{xy}=\frac{Cov(x,y)}{\sqrt{D(x)}\times\sqrt{D(y)}}$$

Where, x and y, are brightness values of two neighboring pixels in the image and N is the quantity of selected neighbor pixels in the image, in order to calculate the correlation. In Table 3, the proposed method’s correlation coefficient is compared with different references. The results show the encrypted image’s data taking distance from each other. Therefore, the effect of encryption is ideal.

Table 3. Comparison of correlation coefficient and entropy with different references

Algorithm	Correlation coefficients			entropy		
	Horizontal	**Vertical**	**Diagonal**	**Red**	**Green**	**Blue**
Our algorithm (Peppers)	-0.00025	-0.00025	0.0008	7.9991	7.9991	7.9991
Our algorithm (Baboon)	-0.0008	0.0004	-0.0006	7.9976	7.9979	7.9978
Ref.[11]	0.0033	0.0042	0.0024	7.9971	7.9969	7.9962
Ref.[20]	0.0017	0.0016	0.0014	7.9926	7.9934	7.9923
Ref.[22]	-0.0084	0.0004	-0.0015	7.9891	7.9900	7.9897
Ref.[23]	0.0208	0.0279	0.0041	7.9973	7.9978	7.9975
Ref.[25]	0.0084	0.0102	-0.0014	7.9979	7.9979	7.9979

4.4. Entropy of Information

Information entropy can be used as a criterion to find out the degree of disturbance in gray pixels. Entropy of an image can be found from the following relation:

$$H(m) = \sum_{i=1}^{2^N-1} p(mi) \log_2 \frac{1}{p(mi)} \tag{13}$$

In this relation $p(m_i)$ is the probability of occurrence of gray surface m_i and the possible number of gray surfaces is 2^{N-1}. The entropy value will be at its highest, that is 8, which means the highest disorder among the pixels of the image. In table (3), the entropy of the proposed method has been compared with different references.

4.5. Key Space Analysis

In the cryptographic algorithm, the key space should be large enough to keep all types of attacks ineffective. The size of key space is the total number of different keys that can be used in cryptographic algorithm. Extreme sensitivity to initial conditions is one of the characteristics of chaotic systems. In this algorithm, initial conditions in Lorenz and Chen systems can be used as encryption keys in pictography and decryption. But three chaos sequences are used for encryption that are the size of M x N and will have a space of M x N x 224. As a result, key space is very wide to sustain against a variety of attacks.

4.6. Differential Attacks

The effect of a pixel shift in the original image on its corresponding encrypted image is used with two criteria, NPCR (number of changing pixel rate) and UACI (unified averaged changed intensity) that can be calculated using the following relationships (Enayatifar, Abdullah & Isnin, 2014).

$$NPCR_{R,G,B} = \left(\sum_{i=1}^{W} \sum_{j=1}^{H} D_{R,G,B}(i, j) / (W \times H) \times 100\% \right) \tag{14}$$

$$UACI_{R,G,B} = \left(\sum_{i=1}^{W} \sum_{j=1}^{H} \frac{\left| C_{R,G,B}(i, j) - C_{R,G,B}(i, j) \right|}{255} / (W \times H) \times 100\% \right) \tag{15}$$

So that H and W, are the length and width of the image respectively. $C_{R,G,B}$ and $C'_{R,G,B}$ are two encrypted images which are taken from two images with one pixel difference and $D_{R,G,B}(i,j)$ is calculated as follows.

$$D_{R,G,B} = \begin{cases} 1, & if\ C_{R,G,B} \neq C'_{R,G,B} \\ 0, & if\ C_{R,G,B} = C'_{R,G,B} \end{cases} \tag{16}$$

We test two original images with the same common initial parameters and values. Table (4) shows the results and mean NPCR and UACI values for the original image after ten repetitions. Clearly, encryption pattern has a high performance and is very sensitive to a minute change in the original image. And can well withstand differential attacks.

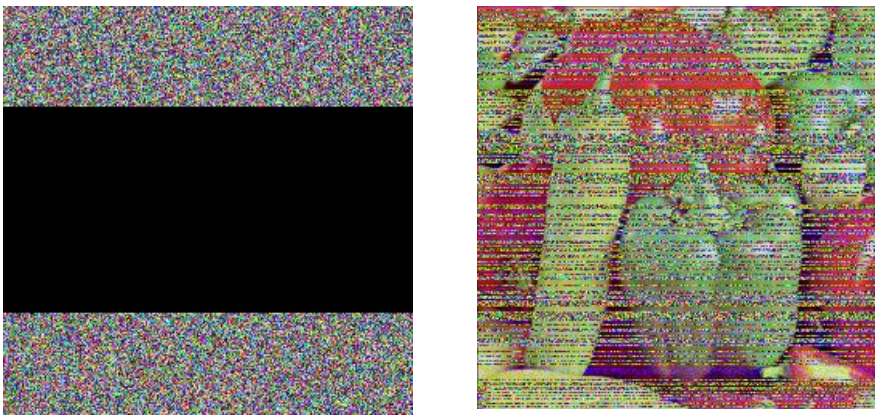
Table 4. Results and mean of NPCR and UACI for various colored images

NPCR R,G,B (%)				UACI R,G,B (%)		
Image	Red	Green	Blue	Red	Green	Blue
Peppers	99.6323	99.6231	99.6139	33.3767	33.3951	33.3977
Baboon	99.6185	99.6321	99.6185	33.5587	33.5246	33.4610
Lena	99.6704	99.5824	99.5949	33.6026	33.5661	33.5028
Average R,G,B	99.6404	99.6125	99.6091	33.5126	33.4952	33.4538
Average for all images	99.6206			33.4872		

4.7. Data Loss Attack

A good cryptographic system should be immune against the effect of data reduction. The proposed design is resistant to data loss attacks. A part of encrypted image may get lost or changed during transmission (Dong, 2013; Wang et al., 2016). Results of the lost data decryption are shown in Figure 9. Although 50% of total data of the image is lost, the decrypted image can still be recognized.

Figure 9. Missing data. (a) 50% of the data is lost, (b) is decryption of image (a) with the lost data.



a b

5. Conclusion

This paper presented a new method for encrypting images, using hyperchaotic functions along with DNA sequences. In this method, one has used Lorenz hyperchaotic functions to break down the relationships between pixels and the Chen hyperchaotic functions for encryption. Also, capabilities and features of hyperchaotic functions, that include sensitivity to initial values, random behavior, non-periodicity and certainty that result in production of semi-random numbers and DNA sequences that have complexities, have been utilized. Based on the test results and security analysis, it can be seen that the correlation between the encrypted image pixels is reduced, and that the histogram of encrypted image has a very uniform distribution. The entropy is 7.9991, which is very close to the ideal value of 8.The algorithm has a good effect and a bigger secret space key. Moreover, the proposed algorithm can also detect attacks, such as resistance against statistical analysis, comprehensive attacks, data loss attacks and noise immunity. All these features show that the proposed algorithm is very suitable for digital image encryption.

REFERENCES

1. Adleman, L. M. (1998). *Computing with DNA. Scientific american*, 279(8), 34-41.
2. Ailenberg, M., Rotstein, O.D. (2009). *An improved Huffman coding method for archiving text, images, and music characters in DNA.* Biotechniques 47(3), 747-789.
3. Annaby M. H., Rushdi M. A., Nehary E. A.. (2018). *"Color image encryption using random transforms, phase retrieval, chaotic maps, and diffusion",* Optics and Lasers in Engineering, 103, 9–23.
4. Dong, C. (2013). *Color image encryption using one-time keys and coupled chaotic sy*stems, Image Communication, http://dx.doi.org/10.1016/j.image.2013.09.006.
5. Enayatifar, R., Sadaei, H. J., Abdullah, A. H., Lee, M., Isnin, I.F. (2015). *A novel chaotic based image encryption using a hybrid model of deoxyribonucleic acid and cellular automata*. Opt. Lasers Eng. 71, 33–41.

6. Enayatifar, R., Abdullah, A.H., Isnin, I.F. (2014). *Chaos-based image encryption using a hybrid genetic algorithm and a DNA sequence*. Opt. Lasers Eng. 56, 83–93.
7. Liu, H., Kadir, A. (2015). *Asymmetric color image encryption scheme using 2D discrete-time map*, Signal Processing, 113, 104–112.
8. Liu, Y., Tang, J., Xie, T. (2014). *Cryptanalyzing a RGB image encryption algorithm based on DNA encoding and chaos map.* Opt. Laser Technol. 60, 111–115.
9. Liu, L., Zhang, Q.,Wei, X. (2012). *A RGB image encryption algorithm based on DNA encoding and chaos map.* Comput. Electr. Eng. 38(5), 1240–1248.
10. Pisarchik, A. N., Zanin, M. (2008). *Image encryption with chaotically coupled chaotic maps*, Physica D, 237, 2638–2648.
11. Shannon, C. (1949). *Communication theory of security systems*, The Bell System Technical Journal, 28, 656-715.
12. Shujun, Li, Chengqing, Li, Guanrong, Chen, Xuanqin, Mou. (2005). *On the dynamical degradation of digital piecewise linear chaotic maps.* International journal of bifurcation and chaosvol. 15(10), 3119-3151.
13. Singh, N, Sinha, A. (2008). *Optical image encryption using fractional Fourier transform and chaos*, Optics and Lasers in Engineering, 46, 117–123.
14. Valandara, M .Y., Jafari Barania, M., Ayubi, P. (2019). *A fast color image encryption technique based on three dimensional chaotic map*. Optik-international journal for light and electron optics, 193, 162-181.
15. Wang, L., Song, H., Liu, P. (2016). *A novel hybrid color image encryption algorithm using two complex chaotic systems*. Optics and Lasersin Engineering 77, 118–125.
16. Wan, X., Zhang, H. (2015). *A color image encryption with heterogeneous bit-permutation and correlated chao*s. Optics Communications, 342, 51–60.
17. Wang, X., Zhao, Y., Zhan,g H., Guo, K. (2016). *A novel color image encryption scheme using alternate chaotic mapping structure*. Optics and Lasers in Engineering, 82, 79–86.
18. Wei, X., Guo, L., Zhang, Q., Zhang, J., Lian, S. (2012). *A novel color image encryption algorithm based on DNA sequence operation and hyper-chaotic system*. J. Syst. Softw. 85(2), 290–299.
19. Wu, X., Kan, H., Kurths, J. (2015). *A new color image encryption scheme based on DNA sequences and multiple improved 1D chaotic maps*. Applied Soft Computing Journal, http://dx.doi.org/10.1016/j.asoc.2015.08.008.
20. Xiao, G., Mingxin, L., Qin, L., Lai, X. (2006). *New field of encryption: DNAencryption. Chin.* Sci. Bull. 51(12), 1413– 1420.
21. Xuejing, K, Zihui, G. (2020). *A new color image encryption scheme based on DNA encoding and spatiotemporal chaotic system.* Signal Processing: Image Communication, 80, 115-130.
22. Zhang, Y., Fu, L.H.B. (2012). *Research on dna encryption*. Applied encryption and network security, InTech Press, Rijeka, Croatia ,357–376.
23. Zhang, Q., Guo, L., Wei, X. (2013). *A novel image fusion encryption algorithm based onDNAsequence operation and hyperchaotic system*. Optik-Int. J. Light Electron Opt. 124(18), 3596–3600.
24. Zhang, Q.,Wei, X. (2013). *A novel couple images encryption algorithm based onDNAsubsequence operation and chaotic system.* Optik-Int. J. Light Electron Opt. 124(23), 6276–6281.
25. Zhang, Q., Guo, L., Xue, X., Wei, X. (2009). *An image encryption algorithm based on DNA sequence addition operation. In: Bio-Inspired Computing. BIC-TA'09*. Fourth International Conference on, pp 1–5. Ieee.

Mohammadreza ZARIF is MSc in Computer Systems Architecture, Department of Computer Engineering, Islamic Azad University, Ferdows Branch, Iran His general research interests are: Computer Systems Architecture, grid computing and cloud computing, peer-to-peer systems, Big Data management, data aggregation, and information retrieval and classification techniques.

Hamid reza GHAFFARY is Assistant professor at the Department of Computer Engineering, Islamic Azad University, Ferdows Branch, Iran. He also works as a scientific researcher in Iran. His general research interests are: distributed systems (design and performance), grid computing and cloud computing, peer-to-peer systems, Big Data management, data aggregation, and information retrieval and classification techniques.

Masoud DAVOUDI is MSc in Artificial Intelligence, Department of Computer Engineering, Islamic Azad University, Ferdows Branch, Iran. His general research interests are: Artificial Intelligence, grid computing and cloud computing, peer-to-peer systems, Big Data management, data aggregation, and information retrieval and classification techniques.