Multimed Tools Appl (2019) 78:2105–2133
https://doi.org/10.1007/s11042-018-6346-1

A novel robust dual diffusion/confusion encryption technique for color image based on Chaos, DNA and SHA-2

Aqeel ur Rehman[1] · **Xiaofeng Liao**[2]

Received: 17 November 2017 / Revised: 2 May 2018 / Accepted: 29 June 2018 /
Published online: 5 July 2018

Abstract In the proposed article, a novel way of confusion is designed by introducing intra-permutation and Exclusive-OR operation with complementary DNA rules that brings randomness in the image. A SHA-256 hash function is used for modification of the initial conditions for 2-Dimensional Logistic map. In the 1st phase of diffusion, the rows of the three colored channels are exchanged by selecting DC-Boxes chaotically and then same operation is applied on the columns. In 2nd diffusion phase, each color channel is permuted independently using chaotic sequence. Before confusion, DNA encoding is applied at pixel level chaotically and transformed each color channel into a linear array. These three arrays are combined into a matrix of three rows and multiple columns. This matrix is divided into blocks; each of size of three DNA bases; one from each color channel and substituted by Intra-channel diffusion using DC-Boxes. In 2nd phase of confusion, matrix is transformed into a large 1D array representing DNA bases of a color image. This large array is split into groups of size of four DNA bases; representing a pixel. These groups are substituted by Exclusive-OR operation with DNA complementary rules that selected chaotically. The proposed algorithm requires only single round of confusion/diffusion operation to achieve high quality of encryption results. This scheme is quite different for color image encryption based on DNA and has better results for different tests like NPCR, UACI, information entropy etc. Besides the larger key space, resistance against common transmission noise is another significant advantage of proposed scheme over some existing systems.

Keywords 2D logistic map · Chaos theory · DNA rules · SHA-256 · Color image encryption Dual diffusion/confusion

✉ Aqeel ur Rehman
rehmancqu@gmail.com

[1] Department of Computer Sciences, COMSATS University, Vehari campus, Pakistan

[2] College of Electronics and Information Engineering, Southwest University, Chongqing, People's Republic of China

1 Introduction

This era is the most beneficent for the human beings in the sense of information. Few decades back, humans were not very involved in sharing their information, in such a fast and efficient way. However, security of this sharing is very important phase, as now a days, a lots of information travel through networks. As the evolvement of internet increase the use of computer network, the security of data became the essential focus of the user. Image data is one of the important data types exchanged in bulk by the user and, in some situations, it requires much focus to exchange securely. In era of current 21st, cryptographic algorithms for securing digital using chaotic systems have fascinated a lot of researchers [4, 7, 13, 15, 18, 21, 22, 27, 30, 33, 34, 43, 45, 46, 48] due to numerous imperative features, such as extremely sensitive dependence on initial conditions, pseudo randomness and ergodicity. The above discussed properties made chaotic system highly suitable candidate to build secure encryption schemes [15, 22, 27, 45].

Chaos based image encryption ciphers are proposed generally on Fridrich's architecture proposed in 1998 [12] which consists of two phases; diffusion and confusion where each phase may have several rounds. The former phase also known as permutation which is used for disruption of the correlation among contiguous pixels of the digital images by exchanging pixel's positions within the image. But this disruptions does not affect the frequencies of gray levels means histograms remains same. The 2nd phase of Fridrich's architecture also called substitution that modifies the pixel intensities, which is a crucial procedure and cannot omitted but permutation can be omitted. This is the reason that security of chaos-based cryptosystems depend both on permutation and diffusion phases. Some of the chaos based image cryptosystems have been cryptanalyzed [20, 32] and shown that they are not secure enough to resist against some known attacks. Yen et al. [44] proposed HCIE scheme and Li et al. [19] cryptanalyzed this technique, and found a weakness that cipher-text only attack can be launched on HCIE. In 2017 Eric Y. Xie et al. [42] cracked Fridrich's chaotic scheme for images. Chanil Pak's introduced new Logistic-Logistic System, Sine-Sine System and proposed an image encryption scheme [30] which soon cryptanalyzed using chosen plaintext attack [37]. But these numbers are limited.

An efficient cipher to encrypt color images is introduced by B. Norouzi et al. [28] based on two hyper-chaotic systems. These chaotic systems were used to generate pseudo-random numbers to perform diffusion and confusion of R, G and B channels of the image simultaneously at pixel level. But permutation phase takes more time than the substitution phase, so the security of cipher does not compromise to run it faster without permutation. In Ref. [29] authors proposed a novel image cipher which utilized hash function and encryption solely depends only on substitution phase without permutation. The substitution phase consists of two-round; horizontal division of plain image and vertical division of plain image. In horizontal division, an array is generated consisting of 1024 segments sized 8×8, and, in vertical division transpose is applied on the array obtained in horizontal division.

Due to the properties of massive parallelism and astonishing material compactness, Deoxyribonucleic Acid (DNA) computing has arrived in the field of encryption for digital images [2, 8, 10, 11, 16, 17, 25, 31, 35, 36, 39, 40, 47]. To achieve encryption using concept of DNA, pseudo random numbers are transformed into DNA bases by employing DNA complementary rules. After that, DNA computations are achieved through biological operations for substitution. Usually, diffusion is followed to increase the safety of proposed cipher. There are eight different DNA rules for transforming digital data into DNA bases and back to digital format. The algebraic operations like additions, subtraction and Exclusive-OR are works on these

DNA bases. X. Wei et al. [39] designed a cipher technique for color images using Deoxyribonucleic acid, addition operation on DNA bases by employing hyperchaotic system. J. Kalpana et al. [16] suggested three modifications in [39] to improve the randomness and better results. In Ref. [47], a new image encryption technique was proposed using DNA sequence operation and hyperchaotic system known to image fusion scheme. Soon it was cryptanalyzed by applying chosen plaintext attack and known plaintext attack [41]. Rehman et al. proposed modified dual fusion technique using SHA-256 to avoid chosen plaintext attack [2]. In 2014, Enayatifar et al. [10] proposed an encryption technique for images using hybrid model of Deoxyribonucleic acid (DNA) masking, based on a genetic algorithm and Logistic map. The same author Enayatifar et al. in [11], designed another novel image encryption technique based on hybrid model of the Tinkerbell chaotic map. X. Wu et al. [40] generated DNA matrices from 1D chaotic sequence and perform XOR operations between encoded plain image and DNA matrices. These DNA matrices are divided into blocks, permute these blocks randomly, and at last Exclusive-OR and addition operations are applied on DNA bases to achieve image encryption. The motivation of the proposed work is to overcome the limited use of DNA complementary rules which were focused only for transformation from digital to DNA [35, 40, 47] and as well as to resist plaintext attacks.

The proposed system has adopted a novel diffusion/confusion approach called Dual Diffusion/Confusion to achieve encryption for three channels of a color image. A SHA-256 hash function is used to modify the initial conditions and to avoid chosen-plaintext attack. The permutation phase consists of two sub phases; called Inter-channel and Intra-channel permutation that applied on the rows and the columns of each channel respectively using 2-Dimensional chaotic maps. The recently proposed [2] DNA encoding and decoding mechanism is adopted to encode every pixel with different DNA rules selected chaotically. The novel Dual confusion phase is accomplished by simple DNA diffusion and Exclusive OR operation on each block/group of the color image. The simple diffusion during confusion phase is processed by randomly selecting DS-boxes and in 2nd confusion, DNA complementary rules are selected chaotically to perform exclusive-OR operation on groups of color image. The proposed scheme has a clear advantage over other encryption techniques that it has nice feature to resist transmission noise, occlusion attack and other impairments.

This paper is organized as follows; in the first two sections, preliminary work is discussed and in section 3 encryption scheme will be introduced. Simulation results and security analysis are presented in section 4 and conclusion is left to the final section of the article.

2 Related work

2.1 2D logistic map

In encryption technique, 2-Dimensional coupled Logistic map [24] is utilized for diffusion and confusion which is as follows,

$$\begin{aligned} u_{n+1} &= \mu_1 u_n(1-u_n) + \gamma_1 u_n \\ v_{n+1} &= \mu_2 v_n(1-v_n) + \gamma_2\left(u_n^2 + u_n v_n\right) \end{aligned} \tag{2.1}$$

The above system of equation is chaotic when $2.75 < \mu_1 < 3.4$, $2.7 < \mu_2 < 3.45$, $0.15 < \gamma_1 < 0.21$ and $0.13 < \gamma_2 < 0.15$ to generate pseudo-random numbers w and x in the interval (0, 1).

The Eq. (2.1) is iterated $4MN$ times using seed and parameters as: $\mu_0 = 0.256789$, $v_0 = 0.854321$, $\mu_1 = \mu_2 = 3.12$ and $\gamma_1 < 0.19$, $\gamma_2 < 0.14$. Both sequences u and v have poor distribution, auto-correlation and cross-correlation features [1]. The average values of two sequences are 0.6534 and 0.6554 but Ref. [1] suggested following processing to incorporate better uniform distribution,

$$\begin{aligned} u_i &= 10^6 \times floor\left(10^6(u_i)\right) \\ v_i &= 10^6 \times floor\left(10^6(v_i)\right) \end{aligned} \tag{2.2}$$

The mean value of u is 0.5013 and mean of v is 0.5006, which are almost close to the ideal mean of 0.5 and can be employed in a cryptographic algorithm.

2.2 DNA coding and complementary rules and algebraic operations

Deoxyribonucleic Acid (DNA) is a biological material found in almost all living beings. It is responsible to transmit parental features into child and also known. In humans, it exists in twisted ladder like structure where DNA bases A, G, C and T are paired. These bases are Adenine (A), Cytosine (C), Guanine (G) and Thymine (T). In 1953, Watson and Crick published article in Nature which defines principle of complementary base pairing [38] and also known as complementary rule. According to complementary rule, DNA bases in each pair must complement to each other like A and T are complementary, C and G are complementary. The binary number system consists of two numbers only; 0 and 1 which are opposite to each other or complementary. In the same way, 00 and 11 are complementary, while 01 and 10 are also complementary. The above discussed four bases A, C, G and T are used to symbolize the binary sequence 00, 11, 10 and 01. There are total twenty-four kinds of DNA transforming rules, however, only eight kinds of them meet the Watson–Crick complementary rule, shown in Table 1. A digital image has pixels of intensities ranging from 0 to 255, so to represent 8-bit pixel intensity value into DNA domain that requires only four DNA bases. For example, if a pixel intensity has a value of 93, its binary value will be "01 01 11 01". The DNA transformed value of 93 depends on the selection of a DNA rule, it will become "AACA" if DNA encoding rule 5 is used as presented in Table 1. To obtain the same intensity value of 93 by using same DNA rule 5 to transform "AACA" into digital format. However, if we select another DNA rule like Rule 1 then "AACA" will be '00 00 01 00' and pixel's intensity value will be 2. This is DNA encoding/decoding method.

A. Rehman[2016] [2] proposed a new way where eight DNA rules which are separated into encoding and decoding groups at pixel level and has better results. There are some algebraic operations are introduced for DNA bases like addition (+), subtraction (−) and Exclusive or (XOR). Only XOR is highlighted in Table 2 that would be applied in proposed scheme.

Table 1 Eight kinds of DNA mapping rules

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

Table 2 XOR operations for DNA Bases

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

3 Proposed scheme

3.1 Initial conditions for chaotic maps

SHA-2 family of hash algorithms has more complex structure than MD5 and MD4, hence more secured. A SHA-256 generates a digest of 256-bits regardless of the size of the input. If there is one bit difference between two inputs, their message digest will be completely different [14]. So, this is used to generate digest of the color image to which encryption is to be done. The message digest is splits into blocks of bits and then convert it into a floating decimal number $key_j \epsilon (0, 0.0156)$ by Eq. (3.1):

$$key_j = hex2dec(k_i \ldots k_n)/2^{50} \tag{3.1}$$

Where $\{j = 1,\ldots,4\}$, $\{i = 1,2,\ldots,44\}$ and k represents the bits of message digest and the last two keys are generated using 40 bits each as follows

$$\begin{aligned} key_5 &= hex2dec(k_{177} \ldots k_{216})/2^{46} \\ key_6 &= hex2dec(k_{217} \ldots k_{256})/2^{46} \end{aligned} \tag{3.2}$$

Now, let denote the six common secret input as a_0, b_0, e_0, d_0, x_0, y_0 then on another key is calculated as

$$key = u_0 + v_0 + e_0 + d_0 + x_0 + y_0 \mod 1 \tag{3.3}$$

Suppose, u_0, v_0 are initial conditions for 2D logistic map to generate two U and V sequences for intra-channel permutations for diffusion and confusion phases then new seeds are as follows;

$$\begin{cases} u_0^{'} = u_0 + key_1 + key \\ v_0^{'} = v_0 + key_2 + key \end{cases} \mod 1 \tag{3.4}$$

There is requirement of two more pseudo-random arrays for encoding and decoding of Red, Green and Blue channels of an image. The initial conditions e_0 and d_0 are used to generate pseudo -random arrays called E and D Then new seed can be designed as follows;

Table 3 DNA complementary rules for plain image and key image [2]

S#	*Chaotic intervals*	*Encoding*	*Decoding*
1	0.001–0.05, 0.20–0.25, 0.40–0.45, 0.50–0.55, 0.95–0.99	"AGCT"	"GTAC"
2	0.05–0.10, 0.30–0.35, 0.60–0.65, 0.070–0.75, 0.85–0.90	"ACGT"	"TGCA"
3	0.10–0.015, 0.35–0.40, 0.55–0.60, 0.65–0.70, 0.80–0.85	"GATC"	"CTAG"
4	0.15–0.20, 0.25–0.30, 0.45–0.50, 0.75–0.80, 0.90–0.95	"CATG"	"TCGA"

$$\begin{cases} e_0^{'} = e_0 + key_3 + key \\ d_0^{'} = d_0 + key_4 + key \end{cases} \mod 1 \tag{3.5}$$

Two more arrays of deterministic pseudo-random numbers are required; one for inter-channel permutation of pixels of Red, Green and Blue channels and other to select DNA Rules for Exclusive-OR operation during confusion phase which can be symbolized as x_0, y_0 and new seed are as follows:

$$\begin{cases} x_0^{'} = x_0 + key_5 + key \\ y_0^{'} = y_0 + key_6 + key \end{cases} \mod 1 \tag{3.6}$$

The dependence of keys on plain image make sure to change for every input, hence more secure [32].

3.2 Diffusion

The color image consists of three matrices called Red, Blue and Green channel. To encrypt all the three channels, 2-dimensional Logistic map is used to generate pseudo-random numbers for permutation. For encryption, 24-bit gray image $I(M \times N \times 3)$ is used as an input, compute hash value and modify common initial conditions as described in section 3.1. The permutation can be explained as:

Step 1: For permutation, iterate 2D Logistic map using seed $u_0^{'}$ and $v_0^{'}$ up to $4MN$ times to produce vector U and V. To normalize U with respect to the set $\{0, 1, 2, 3, 4, 5\}$, we use the following operation,

$$U = {}_{i=1}^{4MN}\left[\left(U \times 10^{14}\right) \bmod 6^{i=1}\right. \tag{3.7}$$

Now, construct a vector λ using U of dimension $1 \times (M+N)$ such that

$$\lambda = {}_{k=4MN-(M+N)+1}^{4MN}[U(k)] \tag{3.8}$$

Split λ into U_1 and U_2 sub arrays of size $1\times$ M and $1 \times$ N respectively. These arrays are used for inter-permutation of the rows and columns of matrices R, G and B on which image I is based.

$$U_1 = \{\lambda_1, \lambda_2, \cdots, \lambda_M\} \quad U_2 = \{\lambda_{M+1}, \lambda_{M+2}, \cdots, \lambda_{MN}\} \tag{3.9}$$

Table 4 Combination for Inter-diffusion/Confusion for encryption

S#	Selection Value	Inter-Channel or ***DC_SETS*** or ***DC_Boxes***	DNA_RULES
1	0	[1 2 3] = ()	"AGCT"
2	1	[1 3 2] = (3 2)	"ACGT"
3	2	[2 3 1] = (231)	"GATC"
4	3	[2 1 3] = (213)	"CATG"
5	4	[3 1 2] = (312)	"GTAC"
6	5	[3 2 1] = (321)	"TGCA"
7	6	–	"CTAG"
8	7	–	"TCGA"

However, U_1 and U_2 can be possible representation of sub-arrays and are used to select permutations set called ***DC_SETS*** in Table 4. These randomly selected ***DC_SETS*** will decide which rows/columns of matrices are interchanged. As an example, if permutation (213) is selected in the beginning of diffusion then 1st row of Green matrix is exchanged with 1st row of Red channel while 1st row of Blue matrix will remain unchanged.

Step 2: Now, U_1 and is used to select random inter-channel permutations called ***DC_SETS*** as according to Table 4 to exchange the rows of RED, GREEN and BLUE channels of an image I using **Algorithm 1**. After row permutation, U_2 is used to exchange columns of $I^{'}$ using **Algorithm 2** as described below.

Algorithm 1: Inter-Channel Row Permutation (U_1, *DC_SETS*, I)

Inputs: U_1; // Transformed Chaotic sequence;
DC_SETS; // Diffusion/Confusion sets,
I, // RGB colored Image of size MN;
Output: Permuted Rows of colored Image I'
For each *i*th row of colored image *I* in M
S = ***DC_SETS*** ($U_1(i)$); //Select a Diffusion/Confusion set (0 - 5) using Table 4
I' (*i*, :, 1) = I (:, :, S (1)); // *i*th row of 1st channel obtained according to S(1) value
I' (*i*, :, 2) = I (:, :, S (2)); // *i*th row of 2nd channel obtained according to S(2) value
I' (*i*, :, 3) = I (:, :, S (3)); // *i*th row of 3rd channel obtained according to S(3) value

End

Algorithm 2: Inter-Channel Column Permutation (U_2, *DC_SETS*, I')

Inputs: U_2; // Transformed Chaotic sequence;
DC_SETS; // Diffusion/Confusion sets,
I' , // Row permuted RGB colored Image of size MN;
Output: Permuted Rows of colored Image I'
For each *i*th column of colored image *I* in N
S = ***DC_SETS*** ($U_2(i)$); //Select a Diffusion/Confusion set (0 - 5) using Table 4
I'' (:, *i*, 1) = I' (:, *i*, S (1)); // *i*th column of 1st channel obtained according to S(1) value
I'' (:, *i*, 2) = I' (:, *i*, S (2)); // *i*th column of 2nd channel obtained according to S(2) value
I'' (:, *i*, 3) = I' (:, *i*, S (3)); // *i*th column of 3rd channel obtained according to S(3) value

End

Step 3: For 2nd phase of diffusion called intra-permutation, above said 2D Logistic map iterated $3MN$ times using seed $x_0^{'}$ and $y_0^{'}$ to get two vectors X and Y. The X is split into three non-overlapped equal sized sub-vectors called X_1, X_2 and X_3. The permuted image $I^{''}$ is split into three channel of one dimensional array called *red, green and blue,* sort X_1, X_2, X_3 and record their index as,

$$\begin{aligned}[V_1, Idx_1] &= \text{sort}(X_1)\\ [V_2, Idx_2] &= \text{sort}(X_2)\\ [V_3, Idx_3] &= \text{sort}(X_3)\end{aligned} \tag{3.10}$$

where $[\bullet,\bullet] = \text{sort}(\bullet)$ is the sequencing index function, V_1 is the new sequence after sorting in ascending order, $Idx_1$is the index value of V_1. After this, re-arrange/diffuse the elements of *red*, *green* and *blue* channels as given;

$$\begin{aligned} red^{'} &= red\ (Idx_1) \\ green^{'} &= green\ (Idx_2) \\ blue^{'} &= blue\ (Idx_3) \end{aligned} \tag{3.11}$$

3.3 DNA encoding

Encoding of an image into DNA bases is simple as discussed in [2] using Table 3 but with little difference. In A. Rehman[2016] [2], a new DNA model is used for encoding two key images while decoding is done in conventional way. The encoding of Red, Green and Blue channels is completed using 1st four DNA rules and other four rules are reserved for decoding. For DNA encoding, two more chaotic sequences E and D of size $3MN$ are generated using new secret keys shown in Eq. (3.5). The elements of E are divided into three equal sized sub-vectors called E_1, E_2 and E_3 where each having size of MN. Now, DNA encoding is applied on each channel independently using chaotic sequence for the selection of DNA rules as given in Table 3.

$$\begin{aligned} R\_DNA &= Encode\left(red^{'}, E_1\right) \\ G\_DNA &= Encode\left(green^{'}, E_2\right) \\ B\_DNA &= Encode\left(blue^{'}, E_3\right) \end{aligned} \tag{3.12}$$

In this way, each color channel is transformed into $1 \times 4MN$ array and then matrix of size $3 \times 4MN$ is built called $IMAGE$ in which each row represents a colored channel of DNA encoded RGB image.

$$IMAGE = cat(R\_DNA, G\_DNA, B\_DNA) \tag{3.13}$$

3.4 Novel confusion

Before going into detail of novel confusion phase, algorithm requires two pseudo-random sequences to accomplish dual confusion,

Step 1: Now, two pseudo-random sequences V and Y of sizes $4MN$ and $3MN$, which were generated in step 1 and 3 at diffusion phase, remain unused. The vector V is processed into floating point values set {0, 1, 2, 3, 4, 5} and Y into another set {0, 1, 2, 3, 4, 5, 6, 7} called $V^{'}$ and $Y^{'}$ shown in Eq. (3.14).

$$\begin{aligned} V^{'} &= \left(V(i) \times 10^{12}\right) \bmod 6 \\ Y^{'} &= \left(Y(i) \times 10^{12}\right) \bmod 8 \end{aligned} \tag{3.14}$$

Step 2: The matrix called $IMAGE$ is divided into sequential blocks b_i of size 3×1 as shown in Eq. (3.15). Each block b_i consists of three DNA bases, one from each channel, hence total b_{4MN} blocks will be formed.

$$b = \left(b_i, b_{i+1}, \ldots, b_{i+\frac{IMAGE}{3\times1}}\right) \tag{3.15}$$

Step 3: The 1st confusion phase is accomplished by applying permutation on each block of image *IMAGE*. For this, chaotic sequence V' is used to select ***DC_SETS*** according to Table 4 for each block b_i of size 3 × 1 and exchange DNA bases as follows,

$$\begin{aligned} S &= DCSETS(V'(i)) \\ b'_i(1,1) &= b_i(S(1),1) \\ b'_i(2,1) &= b_i(S(2),1) \\ b'_i(3,1) &= b_i(S(3),1) \end{aligned} \tag{3.16}$$

The Eq. (3.16) is used to permute the DNA bases of all 4MN blocks where $i = 1, 2, \ldots, 4MN$. This permutation is called the Intra-Channel permutation and it will modify intensities at 2-bits level in each color channel. This is first phase of confusion.

Step 4: In the next phase of confusion, matrix of size 3 × 4MN in which each row represents a colored channel is transformed into one vector of size 1 × 12MN. This 1D array is split into groups; each having four DNA bases representing a pixel as shown in Fig. 1. These groups are then substituted using Exclusive – OR operation with chaotically selected DNA rules using random sequence Y' according to Table 4,

$$\begin{aligned} &Rule = DNA\_RULES\left(Y'(i)\right) \\ &g'(i) = g(i) \oplus Rule \end{aligned} \quad \text{where } i = 1, 2, 3 \ldots 3MN \tag{3.17}$$

3.5 DNA decoding

The substituted groups g'_i for each colored channel are decoded back to digital format by selecting DNA rules randomly according to Table 4. The pseudo-random array D of size $3MN$ is used for selecting DNA Rules randomly shown in Eq. (3.18). This array produced during encoding process was still free to use. After this, encrypted image is split into three equal size arrays; representing a color channel,

$$Cipher(i) = Decode\left(g'_i(i), D(i)\right) \tag{3.18}$$

At last Cipher image is split into three 2D-matrices representing Red, Green and Blue channels of encrypted image.

3.6 Illustration of confusion

A 3 × 4MN matrix is achieved after permutation shown in Fig. 2 in which each row is representing color channel and it is divided into 3 × 1 sized blocks b_{i+1}, b_{i+2} and so on. Each block consists of three DNA bases, one from each channel.

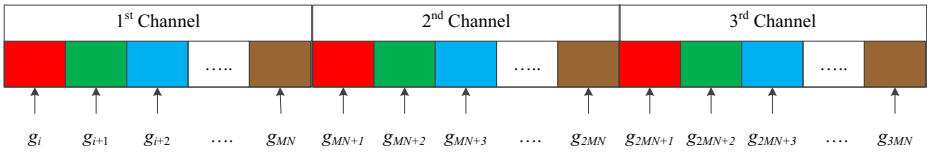


Fig. 1 Splitting of large array into groups of 4 DNA bases each

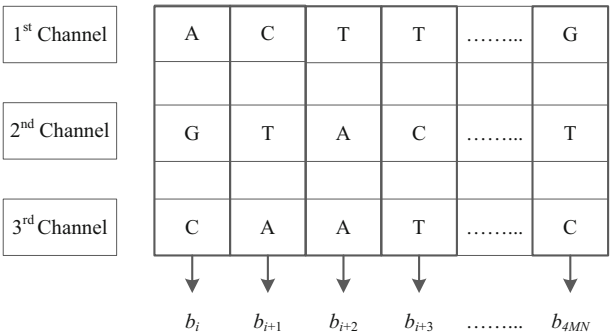


Fig. 2 Blocks b_i of color image after DNA encoding

The first block b_1 can be presented as {A, G, C}, b_2 is {C, T, A}, b_3 is {T, A, A} and fourth block b_4 is {T, C, T} as shown in Fig. 2. Suppose $\acute{Y}$ = {0, 3, 5, 2} then ***DC_SETS*** {[1 2 3], [2 1 3], [3 2 1], [2 3 1]} will be selected for inter-channel permutation. The DNA bases of blocks b_1, b_2, b_3 and b_4 are scrambled according to ***DC_SETS*** into new orders {A, G, C}, {T, C, A}, {A, A, T} and {C, T, T} which is actually confusion and new blocks look like as shown in Fig. 3. In next phase of confusion, these blocks are combined as $3 \times 4MN$ matrix and then transformed into $1 \times 12MN$. This vector is divided into groups of four DNA bases each. Suppose, $\acute{V}$ = {7, 1, 0, 4} then DNA rules {'TCGA', ACGT' and 'AGCT', 'GTAC'} are selected to Exclusive-ORed with groups g_1, g_2, g_3, g_4 shown in Fig. 4. All the phases of proposed scheme are explained in Fig. 5 and encryption and decryption is applied on different images named Lena, Panda and Vegetables as shown in Fig. 6.

4 Security analysis

In proposed algorithm, SHA-2 hash is employed to adapt the initial conditions for chaotic maps. The simulation of the proposed system is done on Intel(R) Core*i*3 processor 3 GHz with 4GB of RAM. The algorithm is coded in Matlab R2015a and compiled by 5.5 on Windows 7 Home Premium. The parameters and initial values set for chaotic maps $u_0 = 0.12345678901234$, $v_0 = 0.23456789012345$, $e_0 = 0.34567890123456$, $d_0 =$

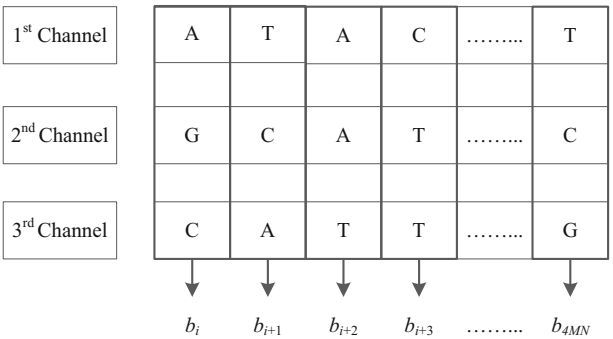


Fig. 3 block b_i of Color image after 1st Confusion

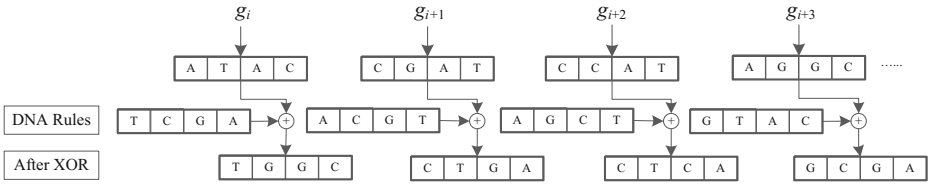


Fig. 4 Blocks g_1 to g_4 of a color image after 2nd Confusion

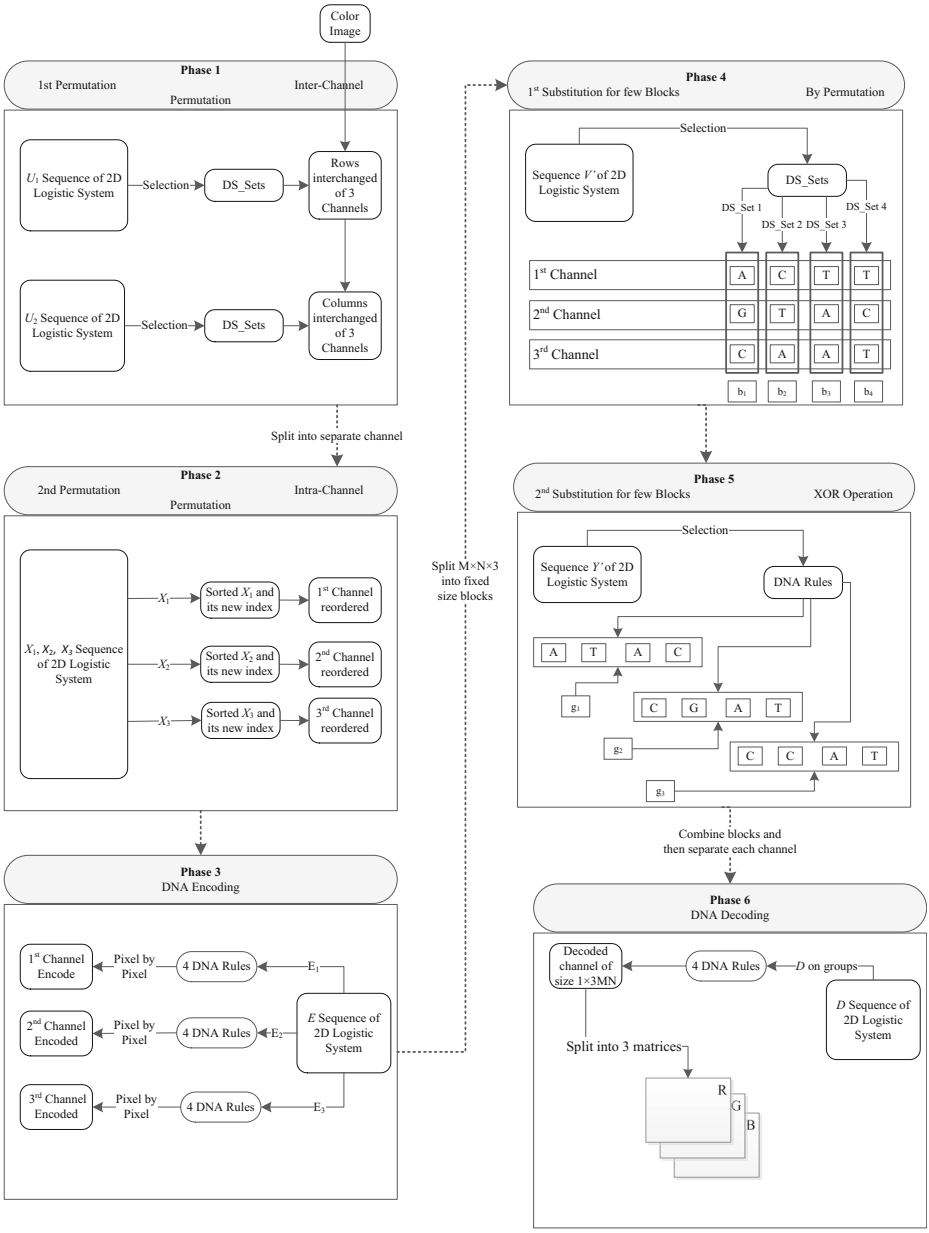


Fig. 5 Block Diagram of Phases of Dual Diffusion/Confusion Image encryption technique

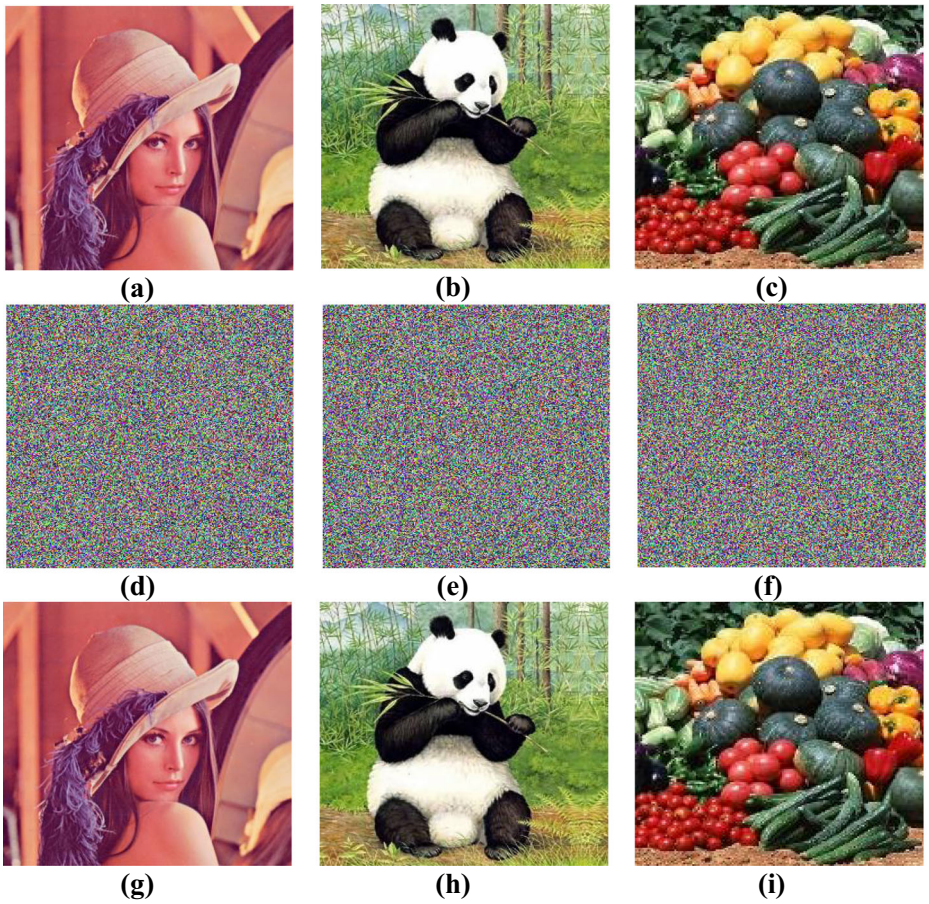


Fig. 6 Experimental results: (**a**) plain-image of Lena; (**b**) plain-image of Vegetables; (**c**) plain-image of Panda; (**d**) cipher-image of Lena; (**e**) cipher-image of Vegetables; (**f**) cipher-image of Panda; (**g**) deciphered image of Lena; (**h**) deciphered image of Vegetables; (**i**) deciphered image of Panda

0.44567890123456, $x_0 = 0.55678901234567$ and $y_0 = 0.68901234562567$ used. The control parameters for 2D logistic map remain fixed which are $\mu_1 = \mu_2 = 3.12$, $\gamma_1 < 0.19$ and $\gamma_2 < 0.14$.

4.1 Key space analysis

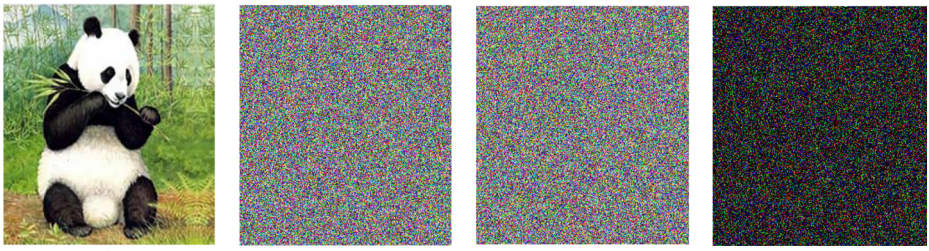
To resist brute force attack, the system must have large key space [12]. The proposed system has high security; because the large key space. The proposed algorithm used ten different common initial conditions to accomplish the cryptographic task and sensitivity to the initial values for coupled logistic map is considered as 10^{-14}. So key space for initial conditions is $S = u_0' u_0' v_0' e_0' d_0' x_0' y_0' = 10^{14\times 6} \cong 10^{84}$ and SHA-2 has space of 2^{256}. Consequently, the total key space is 10^{161}. Therefore, the total key space 2^{535} is much higher than required key space that is 2^{128} [23] to make brute force attack ineffective. The key space is larger than some of the existing cipher techniques [10, 11, 25, 36, 39, 40] as shown in Table 5.

Table 5 Comparison of Key space

Algorithm	Key Space
Proposed	10^{161}
A. Rehman[Jan-2018] [31]	2.4×10^{112}
X. Wei [39]	10^{70}
J. Kalpana [16]	10^{230}
R. Enayatifar[2014] [10]	2^{120}
R. Enayatifar[2015] [11]	2^{129}
X. Wu [40]	10^{90}
L. Liu [25]	10^{56}
A. Rehman[April-2018] [36]	10^{94}

4.2 Key sensitivity analysis

Chaotic maps are good random number generators which have high sensitivity to initial value and control parameter. The 2-dimensional chaotic system shown in Eq. (2.1) is extremely sensitive to its initial conditions and control parameter with precision of 10^{-14}. The plain image of Panda which is shown in Fig. 7(a) and its encrypted image with $u_0^{'}$=0.12345678901234 is displayed in Fig. 7(b) and its decrypted image using key $u_0^{'} = u_0^{'} + 0.00000000000001$ of only one bit difference that results in failure is shown in Fig. 7(c) and (d) is the difference of 7(b) and 7(c). A swift change in one of the secret key; results a total failure in deciphering into plain image. For further investigation of key sensitivity, comparison would be made between different key-sets starting from $\alpha_0 = \left[u_0^{'}, v_0^{'}, e_0^{'}, d_0^{'}, x_0^{'}, y_0^{'}\right]$ as an initial key-set. After a tiny modification in key set α_0 we obtained another key set α_1 by changing only value $u_0^{'} = u_0^{'} + 10^{-14}$, keeping other values unchanged. The key-set α_0 is used to encrypt Panda, Vegetables, Lena, Goat and Pepper images of size 256×256 and then same images are encrypted using new key set called α_1. The pixel-by-pixel difference of two encrypted images that are generated using α_0 and α_1 is measured and the rate of difference is 99.6053% as illustrated in Table 6 for Lena image. Similarly, to analyze the sensitivity of the all secret keys, six different key sets are generatedα_1, α_2, α_3, α_4, $\alpha_5$and$\alpha_6$as shown in Table 6. Using these key sets, the rate of difference is calculated for Lena, Vegetables, Panda and Goat as shown in



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

Fig. 7 Key sensitivity test: **(a)** the original image of Panda; **(b)** the encrypted image of Panda with u_0 = 0.1234567890134; **(c)** Encrypted image of Panda $u_0^{'} + 10^{-14}$; **(d)** the pixel-by-pixel difference between (**b**) and (**c**)

Table 6 Difference rates between two cipher-images generated by slightly different keys

Secret Keys	Proposed			
	Lena	Vegetables	Panda	Goat
$\alpha_1\left(u_0^{'}+10^{-12}\right)$	99.6053	99.5997	99.6206	99.6277
$\alpha_2\left(v_0^{'}+10^{-12}\right)$	99.6343	99.5890	99.6134	99.5895
$\alpha_3\left(e_0^{'}+10^{-12}\right)$	99.6119	99.6022	99.6195	99.6094
$\alpha_4\left(d_0^{'}+10^{-12}\right)$	99.6068	99.6384	99.6028	99.6048
$\alpha_5\left(x_0^{'}+10^{-12}\right)$	99.6104	99.6021	99.6180	99.5977
$\alpha_6\left(y_0^{'}+10^{-12}\right)$	99.6195	99.6078	99.6083	99.6089
Average	99.6147	99.60653	99.61377	99.60633
Avg. of all for Proposed	99.61033			
A. Rehman[Jan-2018] [31]	99.61			
X. Wu [40]	99.6062			
A. Rehman[April-018] [36]	99.6087			

Table 6. The results of key sensitivity are also compared to [31, 36, 40] and the proposed cipher has clear advantage over others.

4.3 Statistical analysis

Claud E. Shannon said, "It is possible to solve many kinds of ciphers by statistical analysis", and therefore he suggested two methods of diffusion and confusion for frustrating the powerful statistical analysis. In the coming sections, it will be validated that the above defined image cipher has good diffusion and confusion properties.

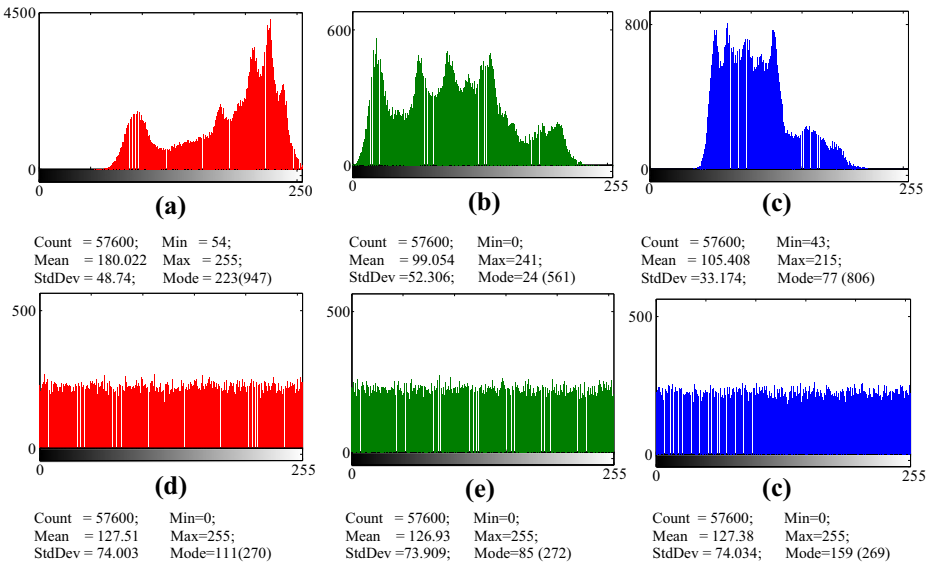


Fig. 8 Histogram of plain and encrypted Lena image for Red, Green and Blue channels

Table 7 Image histogram's variation under different keys

Test Images	α_0	α_1	α_2	α_3	α_4	α_5	α_6	α_7	α_8	α_9
Lena	5457.5482	5463.9947	5452.6895	5462.2752	5.456.9252	5464.2232	5465.9295	–	–	–
Vegetables	5464.9411	5459.9632	5452.6560	5465.7318	5.459.7647	5457.4357	5448.7123	–	–	–
Panda	5456.5095	5456.5859	5462.9812	5462.4565	5446.1422	5466.6434	5453.9827	–	–	–
Average	5459.6660	5460.1810	5456.1090	5463.4880	5454.2770	5462.7670	5456.2080	–	–	–
X Wu [40]	5464.4630	5473.1795	5465.8971	5458.3007	5458.3007	5458.3934	5465.781	5466.2373	5468.3108	5458.8814
A. Rehman[April-2018] [36]	5464.4630	5473.1795	5465.8971	5458.3007	5458.3007	5458.3934	5465.781	5466.2373	5468.3108	5458.8814

Table 8 Percentage of histogram's variance difference of the cipher-images

Test Images	$\alpha_1(\%)$	$\alpha_2(\%)$	$\alpha_3(\%)$	$\alpha_4(\%)$	$\alpha_5(\%)$	$\alpha_6(\%)$	$\alpha_7(\%)$	$\alpha_8(\%)$	$\alpha_9(\%)$
Lena	0.12	0.09	0.09	0.01	0.12	0.15	–	–	–
Vegetables	0.09	0.22	0.01	0.09	0.14	0.30	–	–	–
Panda	0.001	0.12	0.11	0.19	0.18	0.05	–	–	–
Average	0.07	0.14	0.07	0.10	0.15	0.17	–	–	–
X. Wu [40]	0.16	0.14	0.26	0.12	0.18	0.16	0.17	0.15	0.18
A. Rehman$^{\text{April-018}}$ [36]	0.10	0.30	0.30	0.23	0.22	0.26	0.22	0.20	0.15

4.3.1 Histogram analysis

Histogram displays the statistical information of an image. To resist the statistical attack, histogram of encrypted image must be uniform and completely different from histogram of plain image. One typical example LENA image is used to show histogram analysis in Fig. 8. The Fig. 8(a) to (c) are histograms for three color channels and its corresponding encrypted channel shown in Fig. 8(d) to (f). It is obvious that histograms for all channels of encrypted image have almost equal distribution of pixels for each gray levels and completely different from the original image's histogram.

According to the quantitative analysis method in Refs. [36, 40], the uniformity of encrypted image is measured by calculating the variance of histograms. The uniform distribution of encrypted image is revealed by the variance value, lowers the variance score, and hence better is the uniformity. Table 7 provides the variances of histograms of the encrypted images of Lena, Vegetables and Panda, where each column represents variance calculated by each keyset from α_0 to α_6. These key sets are obtained by tiny changing in one of secret keys. The differences of the variances scores in percentage between histograms of two encrypted images obtained separately by the initial key set α_0 and the key set α_i ($i = 1, 2,.. ., 6$); where α_i are defined as those in

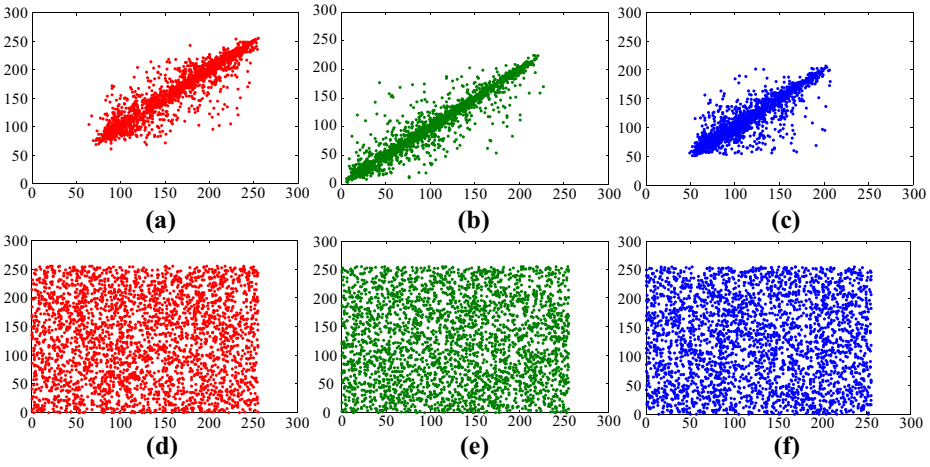


Fig. 9 Correlation of Horizontally, Vertically and Diagonally adjacent pixels of Plain and Encrypted Lena: Original Lena **(a)** Red Channel **(b)** Green Channel **(c)** Blue Channel, Encrypted Lena: **(d)** Red Channel **(e)** Green Channel **(c)** Blue Channel

Section 4.2. The corresponding results are listed in Table 8. In addition, it can be observed from Table 8 that the average variance values change with different plain-images ranging 0.07 to 0.17% which is lower than X. Wu [40] ranging from 0.01 to 0.26%. The results are better than A. Rehman[April-2018] [36] which has average variance fluctuation range 0.10 to 0.30%.

4.3.2 Correlation coefficient

The adjacent pixels have similar intensities in digital images; hence have strong correlation. These strong correlations must be broken in order to avoid statistical attack. The correlation coefficient can be measured within an encrypted image in; horizontal, diagonal and vertical directions by randomly selecting different number of pairs of adjacent pixels. The correlation coefficient values can be measured by Eq. (4.1) which scores between −1 to +1. If the result is close to +1 then it means strong correlation exist or vice versa. The strong and weak correlations are shown in Fig. 9(a) to (c) for plain image and (d) to (f) for cipher image.

$$\begin{aligned} r_{xy} &= \frac{|(Cov)|}{\sqrt{D(x) \times D(y)}} \\ Cov(x,y) &= \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \\ E(x) &= \frac{1}{N}\sum_{i=1}^{N} x_i \\ D(x) &= \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2 \end{aligned} \tag{4.1}$$

In (4.1); x and y represent gray values of two adjacent pixels; $E(x)$ and $D(x)$ are mean and variance values, respectively while $cov(x, y)$ represent the covariance. The Table 9 depicts the results of adjacent pixels in all three directions of plain and corresponding encrypted images of Lena.

Table 9 Correlation analysis and comparison of plain and encrypted image of Lena

Image	Channel	Horizontal	Vertical	Diagonal
Plain	Red	0.9624	0.9920	0.8946
	Green	0.9642	0.9834	0.8187
	Blue	0.9249	0.9649	0.8118
Proposed	Red	−0.0047	0.0028	−0.0043
	Green	−0.0023	−0.0060	−0.0069
	Blue	−0.0038	−0.0057	−0.0112
X. Wei [39]	Red	0.0356	0.1275	0.0783
	Green	0.0763	0.0067	0.0562
	Blue	0.0012	0.0098	0.0058
J. Kalpana [16]	Red	0.0144	0.0083	−0.0468
	Green	0.0163	−0.0180	0.0427
	Blue	−0.0837	−0.0187	−0.0084
A. Rehman[April-2018] [36]	Red	−0.0073	0.0010	−0.0013
	Green	0.0011	−0.0020	0.0078
	Blue	−0.0061	0.0058	−0.0003
X. Wu [40]	–	−0.0084	0.0004	−0.0015

Table 10 Comparison of entropy for different images

Encryption Technique	Test Image	Original Image			Encrypted image		
		Red	Green	Blue	Red	Green	Blue
Proposed	Lena	7.2417	7.5767	6.9170	7.9973	7.9965	7.9969
	Vegetables	7.8277	7.8245	7.3598	7.9962	7.9955	7.9956
	Panda	7.7335	7.6452	7.7969	7.9950	7.9951	7.9946
	Goat	7.2804	7.2596	7.1626	7.9963	7.9963	7.9961
	Peppers	7.3388	7.4962	7.0583	7.9965	7.9963	7.9970
	Average	**7.4844**	**7.5604**	**7.2589**	**7.9963**	**7.9959**	**7.9960**
X. Wu [40]	Lena	7.2933	7.5812	7.0856	7.9893	7.9896	7.9803
	Vegetables	7.7971	7.8215	7.3539	7.9895	7.9895	7.9894
	Panda	7.7118	7.6278	7.7939	7.9894	7.9898	7.9897
	Goat	7.2804	7.2596	7.1627	7.9886	7.9894	7.9894
	Peppers	7.3319	7.5242	7.0793	7.9891	7.9890	7.9897
	Average	**7.4829**	**7.5629**	**7.2962**	**7.9892**	**7.9897**	**7.9897**
A. Rehman[April-2018] [36]	Lena	7.2417	7.5767	6.9170	7.9966	7.9972	7.9967
	Vegetables	7.8277	7.8245	7.3598	7.9848	7.9846	7.9835
	Panda	7.7335	7.6452	7.7969	7.9903	7.9902	7.9903
	Goat	7.2804	7.2596	7.1626	7.9970	7.9970	7.9970
	Peppers	7.3388	7.4962	7.0583	7.9910	7.9918	7.9905
	Average	**7.4844**	**7.5604**	**7.2589**	**7.9919**	**7.9922**	**7.9916**
X. Wei [39]	Lena	–	–	–	7.9928	7.9912	7.9932
	Baboon	–	–	–	7.9945	7.9920	7.9932
	Average	**–**	**–**	**–**	**7.9936**	**7.9916**	**7.9932**
J. Kalpana [16]	Lena	–	–	–	7.9962	7.9993	7.9995
	Baboon	–	–	–	7.9968	7.9968	7.9960
	Average	**–**	**–**	**–**	**7.9980**	**7.9977**	**7.9965**

It is obvious that encrypted images have significantly lower correlation (close to zero) and also better than [16, 36, 39, 40].

4.3.3 Information entropy analysis

Entropy is a measure of randomness of a message. It computes the spread of the pixels for each gray levels of each color channel. If an image has good uniform distribution, it will be stronger against statistical attacks. For color images, Red, Green and Blue channels having intensities between 0 and 255, the ideal entropy score of encrypted message is 8, or in other words; higher the entropy score, higher will be the uniform distribution. It can be defined as,

$$H(X) = \sum_{i=1}^{L} P(x_i) log_2 \Pr(x_i) \tag{4.2}$$

Table 11 PSNR comparison

Image	Proposed			X. Wei [39]			J. Kalpana [16]			A. Rehman[April-2018] [36]		
	R	G	B	R	G	B	R	G	B	R	G	B
Lena	7.85	8.59	9.64	11.57	11.79	11.65	10.37	10.95	11.92	7.87	8.57	9.67
Baboon	8.90	9.50	8.57	12.00	12.44	12.28	11.37	11.72	11.99	8.93	9.50	8.56

Table 12 Comparison of Gray Value Difference

Image	Proposed			J. Kalpana [16]			X. Wei [39]			A. Rehman[April-2018] [36]		
	R	G	B	R	G	B	R	G	B	R	G	B
Lena	0.99	0.97	0.98	0.98	0.98	0.99	0.97	0.97	0.97	0.99	0.99	0.99
Baboon	0.96	0.96	0.95	0.99	0.99	0.99	0.98	0.99	0.99	0.96	0.97	0.95

where $x(i)$ is *ith* grey value, $\Pr(x_i)$ is the probability of grey level $x(i)$. Information entropy is a kind of quantitative measurement of how random a signal source is. In other words, the information entropy can be used to measure the randomness of the image. The information entropy is measured for different images of size 256×256 and their average is also listed in Table 10 and has better than Ref. [36, 39, 40] and comparable to [16].

4.3.4 Peak signal to noise ratio (PSNR)

The Peak Signal to Noise Ratio (PSNR) can be calculated using following equation [16].

$$PSNR(E,I) = 10\log_{10}\frac{255^2(M,N)}{\sum_{\forall i,j}E(M,N)-I(M,N)}dB \tag{4.3}$$

where M and N give the size of the two images for three different channels, E and I represent encrypted and plain images respectively. The PSNR results of the proposed system are shown and compared and far better than Refs. [16, 36, 39] given in Table 11.

4.3.5 Gray difference degree

The Gray difference degree is another statistical measure of randomness by comparing original and ciphered image. It can be computed by using the following equation that is defined as [16].

$$GN(x,y) = \frac{\sum\left[G(x,y)-G\left(x',y'\right)\right]^2}{4} \text{ here } \left(x',y'\right) = \begin{cases} (x-1,y) \\ (x+1,y) \\ (x,y+1) \\ (x,y-1) \end{cases} \tag{4.4}$$

Table 13 Comparison of NPCR and UACI of Lena

Image	Channel	Proposed		J. Kalpana [16]		X. Wei [39]		A. Rehman[April-2018] [36]	
		NPCR	UACI	NPCR	UACI	NPCR	UACI	NPCR	UACI
Lena	Red	99.6551	33.4927	99.6586	33.1154	99.3218	31.2189	99.6001	33.3575
	Green	99.5909	33.5522	99.5409	33.9966	99.2945	31.4183	99.5998	33.4287
	Blue	99.6301	33.5292	99.6697	33.8975	89.3027	31.3621	99.5997	33.3683
Baboon	Red	99.6109	33.5852	99.7350	33.3521	99.1689	31.3478	99.6099	33.3743
	Green	99.6414	33.4235	99.5940	33.6231	99.2749	31.2473	99.6058	33.3829
	Blue	99.6155	33.5400	99.6541	33.9012	99.2321	31.2956	99.5956	33.5604
Average		99.6240	33.5205	99.6420	33.6477	99.2658	31.3150	99.6108	33.4120

where as G(x, y) symbolizes the gray score at position (x, y). The average neighborhood Gray difference of the whole image can be computed by,

$$\mathrm{GVD} = \frac{AN'[GN(x,y)] - AN[GN(x,y)]}{AN'[GN(x,y)] + AN[GN(x,y)]} \tag{4.5}$$

$$AN[GN(x,y)] = \frac{\sum_{x=2}^{M-1}\sum_{y=2}^{N-1} GN(x,y)}{(M-2)(N-2)} \tag{4.6}$$

In Eqs. (4.5) and (4.6), AN and AN′ represent Average Neighborhood gray image; but former represent before encrypting and later used to represent after encryption. The final result of the above equations is called GVD score and it will be 0 if two images are completely same or else 1. The GVD score of encrypted and plain images for Lena and Baboon are computed displayed in Table 12 against each 2-diemnsional matrix of color image independently. The GVD score approaches to 1 for each channel which indicates that plain and encrypted are completely different. The listed results are comparable to Refs. [16, 36, 39].

4.4 Differential analysis

Two known tests called Number of Changing Pixel Rate (NPCR) and Unified Averaged Changed Intensity (UACI) are devised by Eli and Biham [5, 6] to evaluate ciphers against differential attacks. The $N(C^1, C^2)$ and $U(C^1, C^2)$ are defined for NPCR and UACI scores in which C^1 and C^2 are cipher-text images generated from the inputs that differ in one-bit only. The tests for NPCR and UACI are defined in Eqs. (4.7) and (4.8) where L symbolize the largest intensity in the image, M and N are height and width of cipher-text image.

$$N\left(C^1, C^2\right) = \sum_{i=1}^{M}\sum_{j=1}^{N}\frac{D(i,j)}{M \times N} \times 100\% \tag{4.7}$$

$$U\left(C^1, C^2\right) = \sum_{i=1}^{M}\sum_{j=1}^{N}\frac{\left|C^1(i,j) - C^2\left(i,j\right)\right|}{L \bullet M \times N} 100\% \tag{4.8}$$

Table 14 The results of average NPCR $_{R, G, B}$ and UACI $_{R, G, B}$ for different color images for 150 images

Image	Average NPCR (%)			Average UACI (%)		
	Red	Green	Blue	Red	Green	Blue
Lena	99.6251	99.6199	99.6231	33.4877	33.5122	33.5091
Vegetables	99.6100	99.6131	99.6110	33.4609	33.4455	33.4521
Panda	99.6073	99.6051	99.6085	33.4484	33.4584	33.4560
Goat	99.6122	99.6101	99.6113	33.4758	33.4776	33.4553
Pepper	99.6076	99.6062	99.6297	33.4442	33.4385	33.4508
Average (Proposed)	99.6124	99.6109	99.6167	33.4634	33.4664	33.4647
Average of all (Proposed)	99.6115			33.4648		
Average [40]	99.6106	99.6085	99.6107	33.4784	33.4582	33.4673
Average of all [40]	99.6099			33.4680		

Table 15 Comparison of robustness against Gaussian noise

Gaussian	Proposed				H Liu[2012] [26]			A. Kulsoom[2015] [17]		
Variance	Corr.	NPCR	UACI	PSNR	Corr.	NPCR	UACI	Corr.	NPCR	UACI
0.0001	0.90	86.91	3.34	22.14	0.96	99.21	28.44	0.65	69.25	11.85
0.0003	0.84	92.39	4.99	19.84	0.92	99.61	28.64	0.56	77.66	15.04
0.0005	0.81	94.06	5.91	18.82	0.91	99.61	28.80	0.49	81.82	17.21

$$D(i,j) = \begin{cases} 0. & \text{if } C^1(i,j) = C^2(i,j) \\ 1, & \text{if } C^1(i,j) \neq C^2(i,j) \end{cases}. \tag{4.9}$$

The results of NPCR, UACI for three channels of color images Lena and Baboon are shown in Table 13. The scores for NPCR>0.9962 and UACI>0.3352 prove that proposed system has strong resistance against differential attacks and has high quality of encryption than Ref. [36, 39] and comparable to Ref. [16]. In order to address the sensitivity of the plaintext, 150 ciphered images are generated from each of the plain image of Lena, Vegetables, Panda, Goat and Pepper by changing 1-bit of a pixel at a time. These 150 encrypted images are generated using secret key setα_0.At last, average NPCR and UACI scores of three channels are measured and listed in Table 14 which clearly show that proposed system is better than [36, 40].

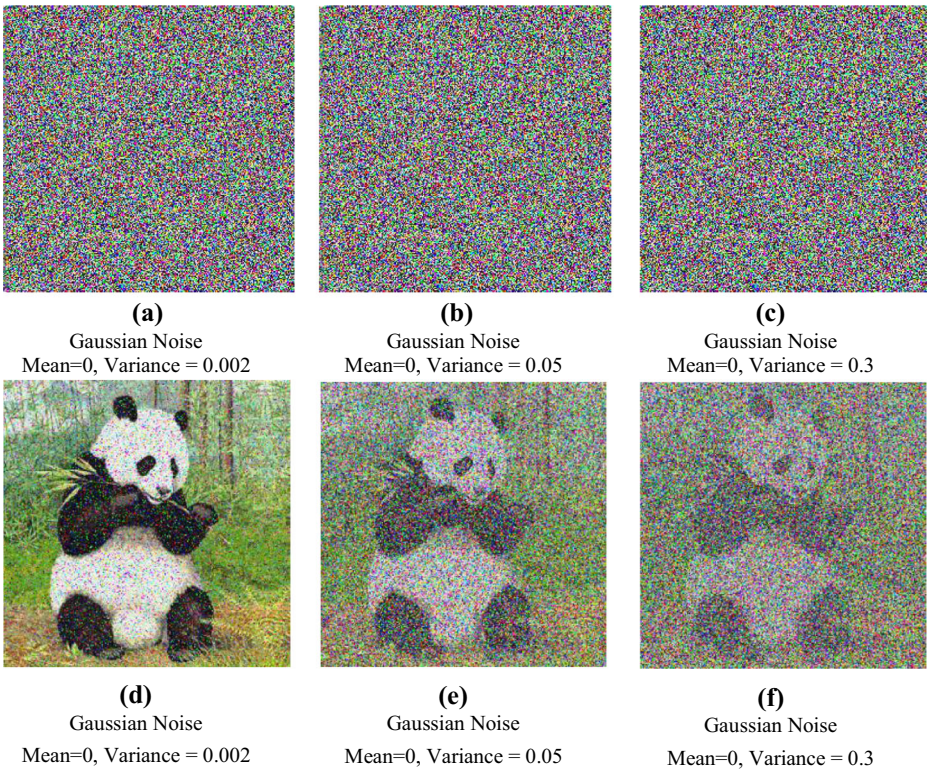


Fig. 10 Gaussian Noise with different variances and corresponding decrypted images

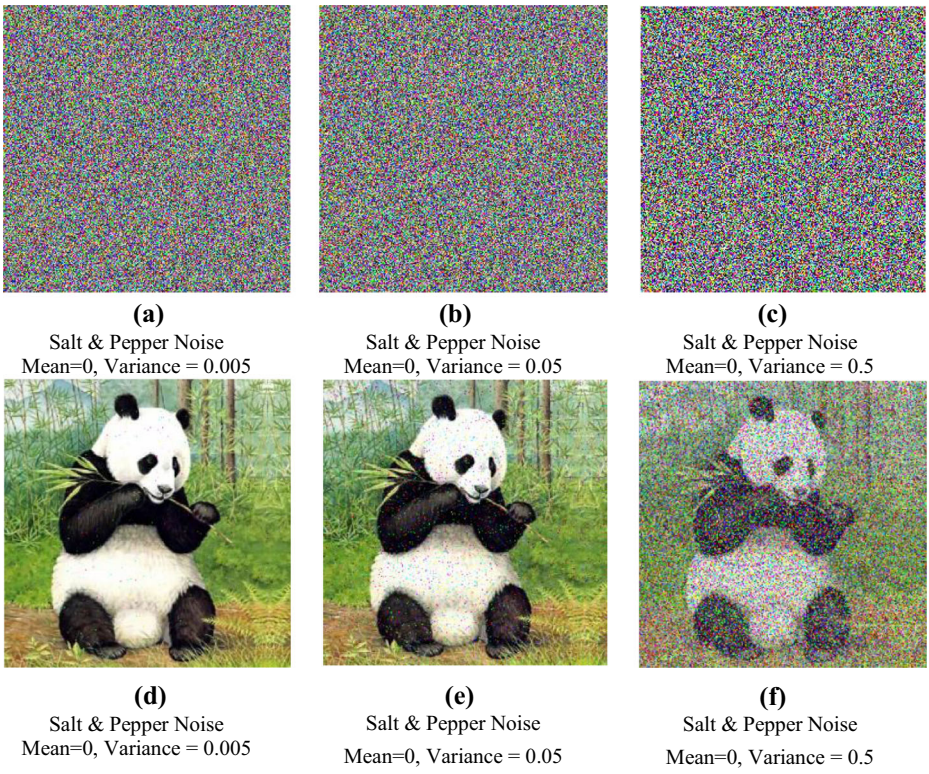


Fig. 11 Salt & Pepper Noise with different variances and its decrypted images

4.5 Robustness test

The real issue is that errors often occur in the data while being transferred over the internet using physical communication system. A trivial alteration in the ciphered image may carry a strong distortion in the decryption process which results in a failure to recover the original image as in Refs. [2, 3]; an error in one pixel; one loses the original image completely. A decent system should be designed in a way that it does not have domino effect in decryption process. The prefer usage to test the robustness is to use Gaussian noise which represents the

Table 16 The PSNR results between the decrypted images and the original Panda image under noise addition

Noise Attack	Parameters	Proposed PSNR	X. Wu [40] PSNR	A. Rehman[April-2018] [36] PSNR
Salt & Pepper	0.005	30.94	30.50	30.87
	0.05	20.87	20.73	20.77
	0.5	10.89	10.75	10.79
Gaussian	0, 0.002	15.15	16.36	16.15
	0, 0.05	10.07	10.98	10.81
	0, 0.3	8.83	9.19	9.57
Contrast Enhancement	70%	13.72	14.31	13.97
	30%	11.29	11.51	11.25
Histogram Equalization	–	33.19	32.42	33.23

inevitable randomness of the real physical medium. Besides this, other impairments are also used to verify the strength of the proposed system.

4.5.1 Noise addition

In the real world environment, noise accumulation is being occurred frequently. This noise lowers and bends the visual information of digital images. For this purpose, Gaussian noise with densities 0.0001, 0.0003 and 0.0005 are added to Fig. 6(d) and different results are measured to compare with Refs. [17, 26]. The proposed system has better robustness than Refs. [17, 26] shown in Table 15. In the next step, ciphered Panda image in Fig. 6(e) is polluted by Gaussian noise with different densities of 0.002, 0.05 and 0.3 as presented in Fig. 10(a) to (c) and corresponding deciphered images are displayed in Fig. 10(d) to (f). Similarly, the encrypted Panda images is contaminated with Salt & Pepper noise with different densities as shown in Fig. 11(a)–(c) and corresponding decrypted images displayed in Fig. 11(d) to (f). The comparison of PSNR for Gaussian and Salt & Pepper noise are measured shown in Table 16. The PSNR value of the proposed system for Salt & Pepper noise is better than [36, 40] but lower in Gaussian noise.

4.5.2 Occlusion attack

During communication over the internet, part of an image can be lost and proposed cipher must be capable to handle deciphering of lossy image in an appropriate manner. To demonstrate the strength of proposed cipher against such a situation, 1/16, 1/8, 1/4 and 1/2 part of encrypted image of Fig. 6(d) is removed which can be seen in Fig. 12(a) to (d) and resultant decrypted images are shown in Fig. 12(e) to (h). The PSNR, Mean Squared Error, NPCR and UCI scores are measured for lossy decrypted images and listed in Table 17. The NPCR and

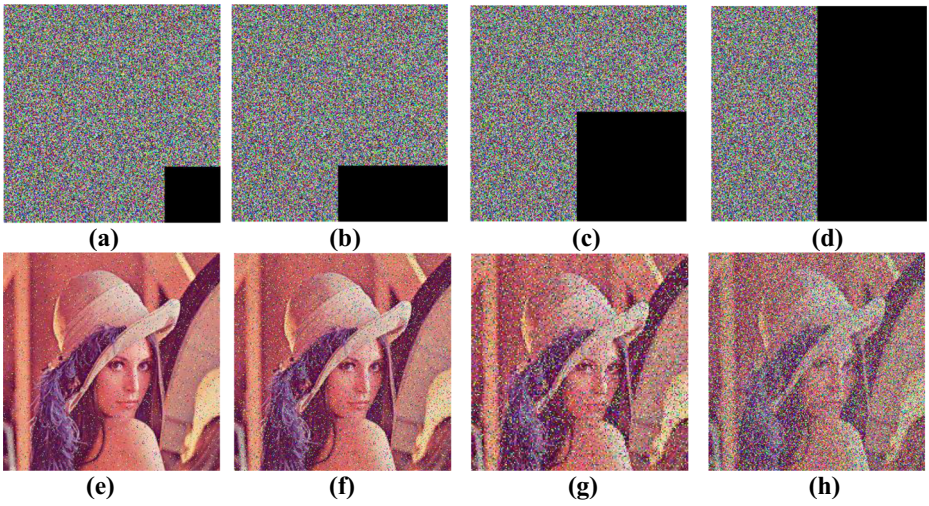
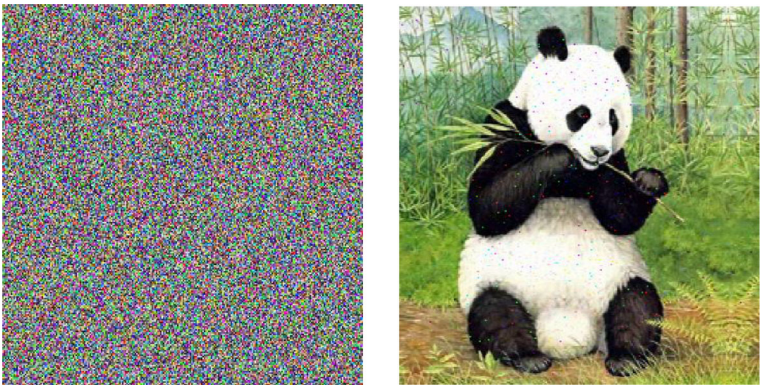


Fig. 12 Occlusion attack analysis by removing a part of **(a)** 1/16 of encrypted Lena; **(b)** 1/8 of encrypted Lena; **(c)** 1/4 of encrypted Lena; **(d)** 1/2 of encrypted Lena; **(e)** Decrypted image of (a); **(f)** Decrypted image of (b); **(f)** Decrypted image of (c); **(g)** Decrypted image of (d)

Table 17 MSE, PSNR, NPCR and UACI between plain and decrypted image under different clipping size

Clipping Size	Proposed				A. Rehman[April-2018] [36]				X. Chai [9]			
	MSE	PSNR	NPCR	UACI	MSE	PSNR	NPCR	UACI	MSE	PSNR	NPCR	UACI
1/16	550.55	20.73	6.23	1.87	579.98	20.57	6.23	1.93	11.21	37.63	14.1	2.2
1/8	1097.90	17.70	12.42	3.78	1155.32	17.57	12.45	3.85	22.63	34.58	28.1	4.4
1/4	2190.94	14.72	24.90	7.53	2289.90	14.59	24.91	7.68	38.94	32.22	42.2	8.3
1/2	4397.14	11.70	49.81	15.13	4578.34	11.58	49.82	15.37	78.04	29.20	84.7	16.6



(a) Histogram Equalized **(b)** Decrypted image from (a)

Fig. 13 Histogram Equalization analysis: **(a)** encrypted histogram equalized Panda image; (**b**) Decrypted Panda image from (a)

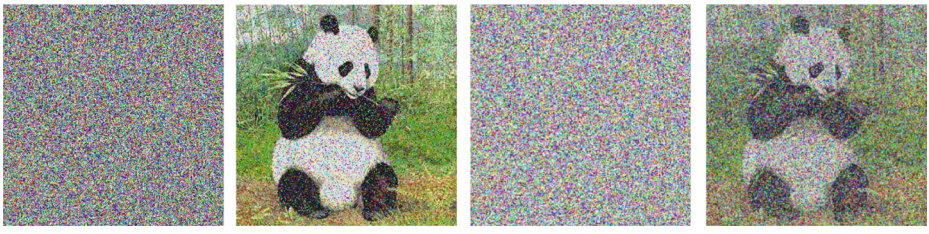
UACI score are better than A. Rehman[April-2018] [36] and X. Chai [9] but PSNR has lower values than [9].

4.5.3 Histogram equalization

Histogram of an image contains range of gray levels as well as occurrences of each gray level. It may be possible that some of gray levels exist with 'zero' frequency. In statistical terminology, Histogram Equalization is a method to stretch the range of gray level from 0 to 255 or in other words increases the contrast. So this technique is applied on the encrypted image of Panda that revealed in Fig. 6(e) on each channel and shown in Fig. 13(a). This histogram equalized image is decrypted and shown in Fig. 13(b). The PSNR value between the decrypted and original Panda images is 33.19 dB as shown in Table 16. It is obvious that the proposed cipher can counterattack the histogram equalization and has better value than Ref. [40] and comparable to Ref. [36].

4.5.4 Contrast Adjustment

A suitable level of brightness and contrast must exist in an image for comfortable viewing where former represent lightness or darkness of a whole image while later defines the



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

Fig. 14 Contrast Adjustment attack for Panda Image: **(a)** contrast adjustment increased by 70%; **(b)** Decrypted image from (a); **(c)** contrast adjustment increased by 30%; **(d)** Decrypted image from (c)

difference of brightness to identify clear separation of different regions. Therefore, contrast adjustment is an image processing mechanism in which the input intensities are mapped to a desired level to enhance the area of interest or region. This contrast enhancement is applied on encrypted image of Panda shown in Fig. 6(e) at two different levels, 70% and 30% where lower value means higher the contrast. These enhanced encrypted images are shown in Fig. 14(a) and (c) and corresponding decrypted images are shown in Fig. 14(b) and (d). The Peak Signal to Noise (PSNR) is computed between the decrypted and original Panda images which are 13.72 dB and 11.29 dB, for 70% and 30% contrast enhancement. The PSNR value is comparable to Ref. [36, 40]. Therefore, our proposed scheme is robust against contrast adjustment attacks.

5 Conclusion

A novel color image encryption algorithm is proposed in this paper based on chaotic maps and SHA-2 hash function. The three channels of a color image are permuted in two phases called Inter-channel and Intra-channel permutation using 2-Dimensional Logistic chaotic map. The recently proposed DNA encoding and decoding mechanism is adopted for each pixel to achieve better randomness. Likewise diffusion, confusion are also performed in two phases; by permutation of DNA bases and using Exclusive-OR operation with DNA complementary rules. After permutation and DNA encoding, color image is transformed into three linear rows; each representing a color channel. This image is divided into blocks; each having three DNA bases, one from each channel. The 1st confusion is performed on each block by inter-DNA permutations using randomly selected permutation sets. For 2nd phase of confusion, three linear arrays are split into groups of four DNA bases, each group representing a pixel and is Exclusive-ORed with DNA complementary rules. The permutation sets for inter-DNA permutation and DNA rules for Exclusive-OR operation are selected randomly. The extensive simulated experiments results prove that proposed algorithm has excellent encryption results in single round and suitable for real time applications due to high efficiency. This scheme is advantageous over other encryption techniques that it has nice feature to resist transmission noise.

Acknowledgements This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFB 0800601, in part by the National Natural Science Foundation of China under Grant 61472331, in part by the Talents of Science and Technology Promote Plan, Chongqing Science & Technology Commission and in part by the Fundamental Research Funds for the Central Universities under Grant XDJK2015C078.

References

1. Ahmad M, Alam MS (2009) A new algorithm of encryption and decryption of images using chaotic mapping. Int J Comput Sci Eng 2(1):46–50
2. Rehman A, Liao X, Kulsoom A, Ullah S (2016) A modified (dual) fusion technique for image encryption using SHA-256 hash and multiple chaotic maps. Multimed Tools Appl 75(18):11241–11266
3. Behnia S, Akhshani A, Mahmodi H, Akhavan A (2008) A novel algorithm for image encryption based on mixture of chaotic maps. Chaos, Solitons Fractals 35(2):408–419

4. Belazi A, Abd El-Latif AA, Belghith S (2016) A novel image encryption scheme based on substitution-permutation network and chaos. Signal Process 128:155–170
5. Biham E, Shamir A (1991) Differential cryptanalysis of DES-like cryptosystems. J Cryptol 4(1):3–72
6. E. Biham and A. Shamir (1993) Differential Cryptanalysis of the Full 16-round DES BT - Advances in Cryptology — CRYPTO' 92: 12th Annual International Cryptology Conference Santa Barbara, California, USA August 16–20, 1992 Proceedings. E. F. Brickell, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 487–496
7. Çavuşoğlu Ü, Kaçar S, Pehlivan I, Zengin A (2017) Secure image encryption algorithm design using a novel chaos based S-box. Chaos, Solitons Fractals 95:92–101
8. Chai X, Yang K, Gan Z (2017) A new chaos-based image encryption algorithm with dynamic key selection mechanisms. Multimed Tools Appl 76(7):9907–9927
9. Chai X, Chen Y, Broyde L (2017) A novel chaos-based image encryption algorithm using DNA sequence operations. Opt Lasers Eng 88:197–213
10. Enayatifar R, Abdullah AH, Isnin IF (2014) Chaos-based image encryption using a hybrid genetic algorithm and a DNA sequence. Opt Lasers Eng 56:83–93
11. Enayatifar R, Sadaei HJ, Abdullah AH, Lee M, Isnin IF (2015) A novel chaotic based image encryption using a hybrid model of deoxyribonucleic acid and cellular automata. Opt Lasers Eng 71:33–41
12. Fridrich J (1998) Symmetric ciphers based on two-dimensional chaotic maps. Int J Bifurc Chaos 8(6):1259–1284
13. Gan Z, Chai X, Yuan K, Lu Y (2017) A novel image encryption algorithm based on LFT based S-boxes and chaos. Multimed Tools Appl:1–25
14. Gotz M, Kelber K, Schwarz W (1997) Discrete-time chaotic encryption systems. I. Statistical design approach. IEEE Trans Circuits Syst I Fundam Theory Appl 44(10):963–970
15. Hua Z, Jin F, Xu B, Huang H (2018) 2D logistic-sine-coupling map for image encryption. Signal Process 149:148–161
16. Kalpana J, Murali P (2015) An improved color image encryption based on multiple DNA sequence operations with DNA synthetic image and chaos. Opt - Int J Light Electron Opt 126(24):5703–5709
17. Kulsoom A, Xiao D (2016) Aqeel-ur-Rehman, and S. A. Abbas, an efficient and noise resistive selective image encryption scheme for gray images based on chaotic maps and DNA complementary rules. Multimed Tools Appl 75(1):1–23
18. DS Laiphrakpam, MS Khumanthem (2017) A robust image encryption scheme based on chaotic system and elliptic curve over finite field. Multimed Tools Appl
19. Li C (2016) Cracking a hierarchical chaotic image encryption algorithm based on permutation. Signal Process 118:203–210
20. Li S, Zheng X (2002) Cryptanalysis of a chaotic image encryption method. Proc - IEEE Int Symp Circuits Syst 2:708–711
21. Li Y, Wang C, Chen H (2017) A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation. Opt Lasers Eng 90:238–246
22. Li B, Liao X, Jiang Y (2017) A novel image encryption scheme based on logistic map and dynatomic modular curve. Multimed Tools Appl:1–28
23. Liu H, Wang X (2010) Color image encryption based on one-time keys and robust chaotic maps. Comput Math with Appl 59(10):3320–3327
24. H Liu, Z Zhu, H Jiang, B Wang (2008) A Novel Image Encryption Algorithm Based on Improved 3D Chaotic Cat Map. 2008 The 9th International Conference for Young Computer Scientists 3016–3021
25. 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
26. Liu H, Wang X, Kadir A (2012) Image encryption using DNA complementary rule and chaotic maps. Appl Soft Comput J 12(5):1457–1466
27. Y Luo, R Zhou, J Liu, S Qiu, Y Cao (2018) An efficient and self-adapting colour-image encryption algorithm based on chaos and interactions among multiple layers. Multimed Tools Appl
28. Norouzi B, Mirzakuchaki S (Oct. 2014) A fast color image encryption algorithm based on hyper-chaotic systems. Nonlinear Dyn 78(2):995–1015
29. Norouzi B, Seyedzadeh SMSM, Mirzakuchaki S, Mosavi MRMR (Feb. 2014) A novel image encryption based on hash function with only two-round diffusion process. Multimedia Systems 20(1):45–64
30. Pak C, Huang L (2017) A new color image encryption using combination of the 1D chaotic map. Signal Process 138:9
31. Rehman A et al (2018) An efficient mixed inter-intra pixels substitution at 2bits-level for image encryption technique using DNA and chaos. Optik (Stuttg) 153:117–134
32. Rhouma R, Belghith S (2008) Cryptanalysis of a new image encryption algorithm based on hyper-chaos. Phys Lett A 372(38):5973–5978

33. S Sheela, KV Suresh, T Deepaknath (2018) Image encryption based on modified Henon map using hybrid chaotic shift transform
34. Silva-García VM, Flores-Carapia R, Rentería-Márquez C, Luna-Benoso B, Aldape-Pérez M (Sep. 2018) Substitution box generation using Chaos: an image encryption application. Appl Math Comput 332:123–135
35. Rehman A, Liao X, Kulsoom A, Abbas SA (2015) Selective encryption for gray images based on chaos and DNA complementary rules. Multimed Tools Appl 74(13):4655–4677
36. Rehman A, Liao X, Ashraf R, Ullah S, Wang H (2018) A color image encryption technique using exclusive-OR with DNA complementary rules based on Chaos theory and SHA-2. Optik (Stuttg) 159:348–367
37. Wang H, Xiao D, Chen X, Huang H (2017) Cryptanalysis and enhancements of image encryption using combination of the 1D chaotic map. Signal Process 144:444–452
38. Watson JD, Crick FHC (1953) A Structure for Deoxyribose Nucleic Acid. Nature 171:737–738
39. 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
40. Wu X, Kan H, Kurths J (2015) A new color image encryption scheme based on DNA sequences and multiple improved 1D chaotic maps. Appl Soft Comput J 37:24–39
41. Xie T, Liu Y, Jie T (2014) Breaking a novel image fusion encryption algorithm based on DNA sequence operation and hyper-chaotic system. Opt - Int J Light Electron Opt 125(24):7166–7169
42. Xie EY, Li C, Yu S, Lü J (2017) On the cryptanalysis of Fridrich's chaotic image encryption scheme. Signal Process 132:150–154
43. Yavuz E, Yazici R, Kasapba MC, Yamac E (2016) A chaos-based image encryption algorithm with simple logical functions. Comput Electr Eng 54:471–483
44. Yen J-I, Guo J-C (2000) Efficient hierarchical chaotic image encryption algorithm and its VLSI realisation. IEE Proc - Vision, Image, Sign Proc 147(2):167
45. Zhang Y (2018) The unified image encryption algorithm based on chaos and cubic S-box. Inf Sci (Ny) 450:361–377
46. Zhang X, Wang X (2017) Multiple-image encryption algorithm based on mixed image element and chaos. Comput Electr Eng 62:401–413
47. Zhang Q, Guo L, Wei X (2013) A novel image fusion encryption algorithm based on DNA sequence operation and hyper-chaotic system. Opt - Int J Light Electron Opt 124(18):3596–3600
48. Zhou G, Zhang D, Liu Y, Yuan Y, Liu Q (2015) A novel image encryption algorithm based on chaos and line map. Neurocomputing 169:150–157

Aqeel ur Rehman received his M.Sc degree in Computer Science from The Islamia University of Bahawalpur. He received his second Master degree in Computer Engineering from UET Taxila (CASE campus) Islamabad, Pakistan. Currently, he is pursuing Post Doctoral felloship in the department of Electrnoics and Information engineering, Southwest University Chongqing, PR. China. He was working as Assistant professor at COMSATS University Islamabad, Vehari Campus, Pakistan and now on study leave for Post Doc. His primary areas of research are Non-linear dynamics and cryptography.

Xiaofeng Liao (SM'11) received the B.S. and M.S. degrees in mathematics from Sichuan University, Chengdu, China, in 1986 and 1992, respectively, and the Ph.D. degree in circuits and systems from the University of Electronic Science and Technology of China in 1997. From 1999 to 2001, he was involved in postdoctoral research at Chongqing University, Chongqing, China, where he is currently a Professor. From November 1997 to April 1998, he was a Research Associate at the Chinese University of Hong Kong. From October 1999 to October 2000, he was a Research Associate at the City University of Hong Kong. From March 2001 to June 2001 and March 2002 to June 2002, he was a Senior Research Associate at the City University of Hong Kong. From March 2006 to April 2007, he was a research fellow at the City University of Hong Kong. At present, he is a Professor at Southwest University and the Dean of College of Electronics and Information Engineering. He has published more than 400 international journal and conference papers. His current research interests include neural networks, nonlinear dynamical systems, bifurcation and chaos, and cryptography.