Multimed Tools Appl
DOI 10.1007/s11042-015-2515-7

A robust image encryption algorithm resistant to attacks using DNA and chaotic logistic maps

Anchal Jain · Navin Rajpal

Received: 28 July 2014 / Revised: 1 February 2015 / Accepted: 13 February 2015

Abstract An image encryption technique using DNA (Deoxyribonucleic acid) operations and chaotic maps has been proposed in this paper. Firstly, the input image is DNA encoded and a mask is generated by using 1D chaotic map. This mask is added with the DNA encoded image using DNA addition. Intermediate result is DNA complemented with the help of a complement matrix produced by two 1D chaotic maps. Finally, the resultant matrix is permuted using 2D chaotic map followed by DNA decoding to get the cipher image. Proposed technique is totally invertible and it can resist known plain text attack, statistical attacks and differential attacks.

Keywords Image encryption · Chaotic map · DNA operation · Statistical attack · Differential attack

1 Introduction

Security of image data transmitted over public network is a major challenge due to the risk of eavesdropping. Since image is large in size and it has special storage format, traditional encryption techniques like RSA, DES, AES and IDEA are not suitable options for image data encryption due to their increased computation. Encryption of image should be faster to meet the real time constraint. Various encryption techniques have been introduced by the research community for image data which are faster and secure in nature.

Chaotic maps are nonlinear and deterministic. They are very sensitive to the input parameters and a slight change in these parameters results an entirely different output in the

A. Jain (✉)
Inderprastha Engineering College, Ghaziabad, India
e-mail: anchalresearch10@gmail.com

N. Rajpal
University School of Information and Communication Technology, GGSIP University, Delhi, India
e-mail: navin_rajpal@yahoo.com

Published online: 27 February 2015

chaotic maps [10]. A number of image encryption algorithms using chaotic maps have been proposed in literature. Bourbaki's et al.[2] suggested an encryption technique using SCAN patterns that encrypts as well as compresses an image simultaneously. N. Pareek et al [16] used a one dimensional chaotic logistic map for image encryption. They used 8 different encryption/decryption operations depending on the output of the logistic map whereas Schwinger [18] proposed a chaotic Kolmogorov flow based encryption technique where entire image is treated as a single block and it is permuted with the help of key based chaotic system. To increase the key space size, multidimensional chaotic maps have also been used. Fridrich [5] used two dimensional standard baker map to generate symmetric block encryption of image. Authors in [15][3] generalized a two dimensional chaotic maps to three dimensional for real time secure image encryption. They utilized this three dimensional chaotic map to scramble the position of pixels and to confuse the correlation among the encrypted and original image. A. Jain et al. [8] suggested a two layered chaotic image encryption in which the first layer creates diffusion and the second layer creates substitution and authors claimed the proposed technique to be more secure in terms of known plain text attack. Y. Wang et al. [21] proposed a faster image encryption technique in which an image is partitioned in to a number of block and these blocks are shuffled though a chaotic sequence. At the same time, content of the block is also modified.

With the introduction of DNA computing which supports high parallelism and huge information density, DNA cryptography is born which is used for image data encryption. Gehani at el. [6] used DNA strands to introduce one time pad image cryptography which is effective but incorporates complex biological operations. Authors in [11] suggested an image encryption approach in which the secret key is generated using input image and a common key. Pixels in an image are DNA encoded and each nucleotide in DNA encoded image is transformed to its base pair for random number of times with the help of DNA complementary rule. Q. Zhang at el. [24][23] used the concept of pseudo DNA sequence to transform the image and suggested image encryption techniques using hyper chaotic map and DNA addition operation. A. Jain at el. [9] used replication property of DNA alongwith DNA addition and complement rules for image encryption. Initial conditions of chaotic map are generated using 72-bit external key and authors claimed for larger key space. Authors in [19] proposed an image encryption technique using block shuffling and chaotic map. They randomly permuted overlapping image blocks as first level of encryption and used a secret key generated with the help of a chaotic map to perform xor operations with the image blocks and randomly selected secret key blocks for second level of encryption. Authors claimed for a larger Kay space and low computation cost.

Q. Zhang at el.[25] proposed an image encryption algorithm based on DNA encoding and two chaotic maps. They first transformed the image pixels using DNA sequence and divided the transformed image into some equal blocks. These blocks are DNA added with the help of a two dimensional chaotic map and finally the complement process is applied on the result with the help of a binary matrix generated by a one dimensional chaotic map. However, this technique has severe drawbacks as analyzed by H. Hermia's at el. [7]. They found this encryption approach as non-invertible and prone to known plain text attack. Later, authors in [25] modified their work and proposed an invertible technique for GB image encryption by modifying the DNA addition in [12] which was again cryptanalyst by A. Belazi at el. [1] and subsequently by Y. Liu at el. [14] and though it was found to be invertible but still vulnerable to known plain text attack. Authors in [14] also analyzed it for the encryption results in terms of key and image sensitivity and found that it is not sensitive to changes in them. Further, it can be observed that this technique is not suitable for gray image cryptography.

In this paper, the image encryption technique proposed in [25] (referred as original encryption scheme hereafter) has been modified and a new encryption scheme is proposed which is not only invertible but it can resist the plain text attack also. The paper is organized as follows. In Section 2, a brief introduction to chaotic maps, DNA encoding and DNA operations used in this paper are given. Section 3 describes the proposed image encryption technique which is followed by its advantages over the original encryption scheme in Section 4. Experimental results and analysis is performed in Section 5 and paper is concluded in Section 6.

2 Chaotic maps, DNA encoding and DNA operations

2.1 Chaotic maps

The proposed image encryption technique uses two logistic maps - 1D logistic map and 2D logistic map as given in the following (1) and (2) respectively [13].

$$X_{n+1} = rX_n(1 - X_n) \tag{1}$$

It has been found that for the values of parameter r in the range 3.057 to 4, outcome of (1) is highly chaotic in nature.

$$\begin{aligned} X_{n+1} &= r_1 X_n(1 - X_n) + s_1 Y_n^2 \\ Y_{n+1} &= r_2 Y_n(1 - Y_n) + s_2(X_n^2 + X_n Y_n) \end{aligned} \tag{2}$$

where r_1, r_2, s_1 and s_2 are parameters to control the chaotic behavior of logistic map in (2). When $2.75 < r_1 \leq 3.4$, $2.75 < r_2 \leq 3.45$, $0.15 < s_1 \leq 0.21$ and $0.13 < s_2 \leq 0.15$, (2) generates chaotic sequence in the range of [0,1]. Values of parameters $s_1$ands_2 have been set as 0.17 and 0.14 for the experimental results in this paper.

2.2 DNA encoding and decoding

There are four nucleic acid bases A, C, G and T used in a DNA sequence in which A is complement to T and C is complement to G. Two bit binary sequences 00, 01, 10 and 11 are used to represent them. As bit 0 is complement to bit 1, binary sequence 00 is treated as complement to 11 and 01 to 10. Therefore, one way to encode 00, 01, 10 and 11 using DNA bases is C, A, T and G respectively. Using this encoding, if the grey scale value of an image pixel is 135, its binary value 10000111 can be DNA encoded as TCAG from which pixel value can be obtained back through DNA decoding by replacing DNA bases with their binary sequences. Figure 1 shows the flowcharts of DNA encoding and decoding processes used in this paper.

2.3 DNA addition, subtraction and complement operations

Addition and subtraction operations of DNA bases are shown in Table 1 and 2 respectively.

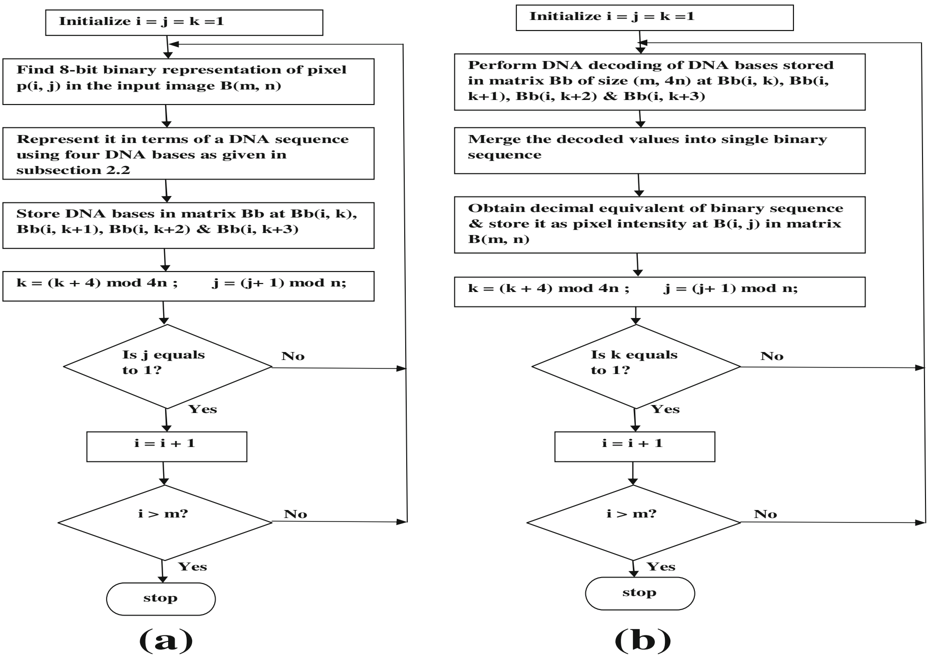

Initialize i = j = k =1
Find 8-bit binary representation of pixel p(i, j) in the input image B(m, n)
Represent it in terms of a DNA sequence using four DNA bases as given in subsection 2.2
Store DNA bases in matrix Bb at Bb(i, k), Bb(i, k+1), Bb(i, k+2) & Bb(i, k+3)
k = (k + 4) mod 4n ; j = (j+ 1) mod n;
Is j equals to 1?
No
Yes
i = i + 1
i > m?
No
Yes
stop
(a)
Initialize i = j = k =1
Perform DNA decoding of DNA bases stored in matrix Bb of size (m, 4n) at Bb(i, k), Bb(i, k+1), Bb(i, k+2) & Bb(i, k+3)
Merge the decoded values into single binary sequence
Obtain decimal equivalent of binary sequence & store it as pixel intensity at B(i, j) in matrix B(m, n)
k = (k + 4) mod 4n ; j = (j+ 1) mod n;
Is k equals to 1?
No
Yes
i = i + 1
i > m?
No
Yes
stop
(b)

Fig. 1 Flowchart of (a) DNA encoding (b) DNA decoding

Each DNA sequence base $m_i \in (A, C, G, T)$ satisfies the following rule for complementary operation as suggested in [11] -

$$\begin{aligned} m_i \neq BP(m_i) \neq BP(BP(m_i)) \neq BP(BP(BP(m_i))) \\ m_i = BP(BP(BP(BP(m_i)))) \end{aligned} \tag{3}$$

where $BP(m_i)$ is the base pair of m_i which is different from m_i in at least one bit position. There are total 6 group complementary rules (AT)(TC)(CG)(GA), (AT)(TG)(GC)(CA), (AC)(CT)(TG)(GA), (AC)(CG)(GT)(TA),(AG)(GT)(TC)(CA) and (AG)(GC)(CT)(TA). Any one of them can be applied though group complementary rule (AT)(TC)(CG)(GA) has been chosen in the proposed technique. In this complementary rule, T is the base pair of A, C is the base pair of T, G is the base pair of C and A is the base pair of G.

3 Proposed image encryption technique

The block diagram of the proposed image encryption algorithm in shown in Fig. 2.

Table 1 Addition operation of DNA sequences

+	A	C	T	G
A	C	A	G	T
C	A	C	T	G
T	G	T	C	A
G	T	G	A	C

Table 2 Subtraction operation of DNA sequences

-	A	C	T	G
A	C	G	A	T
C	A	C	T	G
T	G	T	C	A
G	T	A	G	C

3.1 Generation of secret key

Secret key for an image is generated using the method suggested in [4]. Two initial values $x_1$andy_1 have been chosen to compute $x_0$andy_0 as follows -

$$
\begin{aligned}
x_0 &= \left(\frac{mod(\sum_{i=1}^{m/2} p_{ij}, 256)}{256} + x_1\right) mod\ 1 \\
y_0 &= \left(\frac{mod(\sum_{i=m/2+1}^{m} p_{ij}, 256)}{256} + y_1\right) mod\ 1
\end{aligned}
\tag{4}
$$

where p_{ij} is the pixel value of 8-bit input image of size $m \times n$. These values of x_0, and y_0 along with two different values of system parameter r (r_3 and r_4) are used to generate chaotic sequences with the help of 1D logistic map given in (1) whereas x_0, y_0, r_1 and r_2 are chosen as parameters for 2D logistic map given in (2). Therefore, these parameters $(x_1, y_1, r_1, r_2, r_3, r_4)$ work as secret key for the proposed image encryption method.

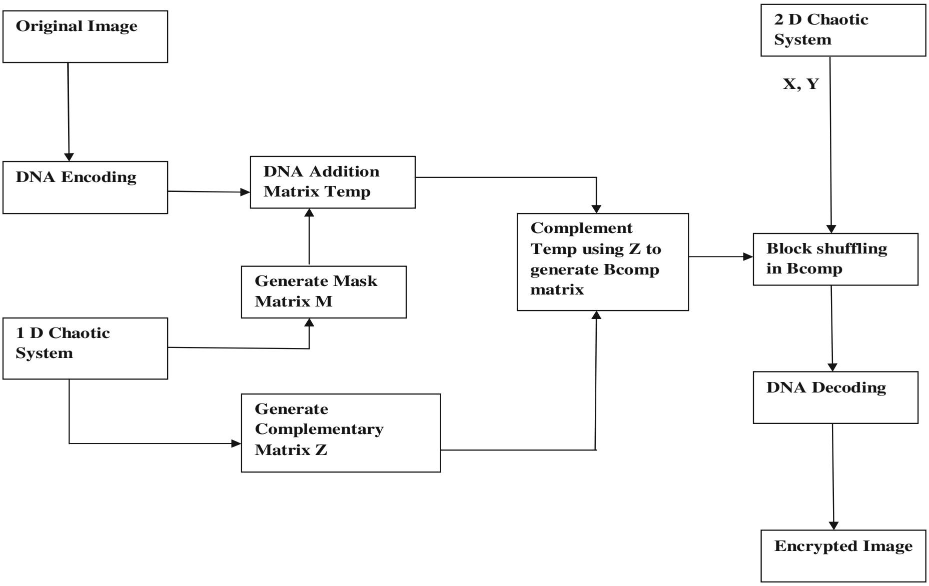


Fig. 2 Block diagram of the proposed image encryption algorithm

3.2 Image encryption using DNA sequence

1. In the input image B of size $m \times n$, perform DNA encoding on the binary representation of each pixel as shown in Fig. 1 to obtain DNA encoded matrix Bb of size $m \times 4n$.
2. Generate a chaotic sequence $C_{seq} = (c_1, c_2, ...c_{mn})$ of length $m \times n$ under initial condition x_0 and r_3 using (1). Convert each value of c_i, $1 \leq i \leq mn$ into its binary form, extract 8 most significant bits and encode them using DNA encoding. Arrange all the values in a matrix mask form M of size $m \times 4n$. A flowchart for the generation of mask matrix M is shown in Fig. 3.
3. Perform DNA addition using mask matrix M and matrix Bb to generate matrix Temp of size $m \times 4n$.
4. Two chaotic sequences z_1 and z_2 are produced by using two 1D logistic maps given in (1) with initial conditions (x_0, r_3) and (y_0, r_4) whose lengths are m and 4n respectively. Reconstruct z_1 and z_2 as two matrices $Z_1(m, 1)$ and $Z_2(1, 4n)$. Multiply Z_1 and Z_2 to obtain matrix Z of size $m \times 4n$. Map the values z of Z into (0,1,2,3) using the following function $f(x)$ to get a complementary matrix -

$$\text{f(x)} = \begin{cases} 0, & 0 < z \leq 0.25 \\ 1, & 0.25 < z \leq 0.50 \\ 2, & 0.5 < z \leq 0.75 \\ 3, & 0.75 < z \leq 1 \end{cases} \tag{5}$$

5. Use DNA complementary rule as suggested in Section 2.3 to complement matrix Temp obtained in step 3. Therefore, coefficient Temp(i,j) of matrix Temp is complemented as in (6) based on the value of z(i,j) of matrix Z.

$$Temp(i, j) = \begin{cases} comp_0(Temp(i, j)) \ if \ z(i, j) = 0 \\ comp_1(Temp(i, j)) \ if \ z(i, j) = 1 \\ comp_2(Temp(i, j)) \ if \ z(i, j) = 2 \\ comp_3(Temp(i, j)) \ if \ z(i, j) = 3 \end{cases} \tag{6}$$

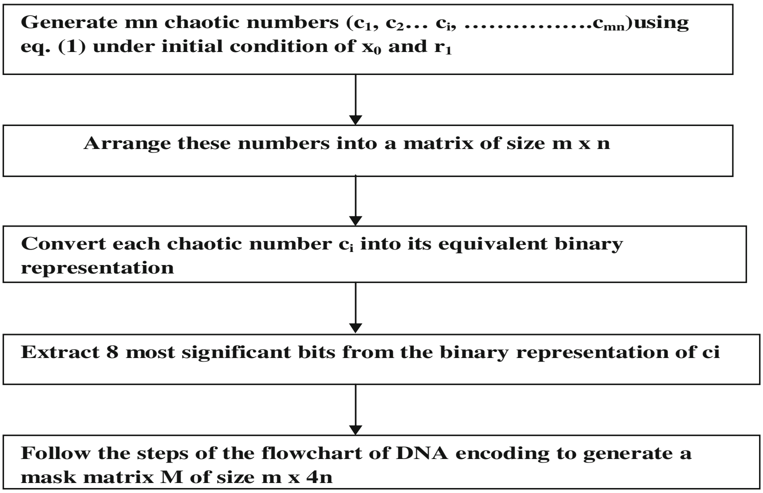

Generate mn chaotic numbers (c₁, c₂... cᵢ, ...c_mn)using eq. (1) under initial condition of x₀ and r₁
Arrange these numbers into a matrix of size m x n
Convert each chaotic number cᵢ into its equivalent binary representation
Extract 8 most significant bits from the binary representation of ci
Follow the steps of the flowchart of DNA encoding to generate a mask matrix M of size m x 4n

Fig. 3 Flowchart for generation of mask matrix

where
$comp_0(m) = m$
$comp_1(m) = B(m)$
$comp_2(m) = B(B(m))$
$comp_3(m) = B(B(B(m)))$.
Complemented Temp matrix is represented as Bcomp of size (m, 4n).

6. Divide Bcomp into equal size blocks Bcomp(i,j) of size 4 × 4 where $1 \leq i \leq \frac{m}{4}$ and $1 \leq j \leq n$.

7. Generate two chaotic sequences $X = (x_1, x_2, \ldots\ldots\ldots\ldots x_{m/4})$ and $Y = (y_1, y_2, \ldots\ldots\ldots y_n)$ through 2D logistic map given in (2) under the initial values x_0 and y_0 and system parameters r_1, r_2, s_1 and s_2.

8. Sort sequences X and Y in the ascending order and the location of values in sort(X) and sort(Y) in original X and Y are used to generate new sequences X' and Y'.

9. Let the values in X' and Y' represent the row and column coordinates used for shuffling the blocks of Bcomp. In other words, it can be expressed as Bcomp($x_i^{'}, y_j^{'}$) where $(x_1^{'}, x_2^{'}, \ldots\ldots x_{m/4}^{'})$ and $(y_1^{'}, y_2^{'}, \ldots\ldots\ldots y_n^{'})$ are location values in X' and Y'. Shuffling of blocks is performed by interchanging blocks Bcomp(i,j) with Bcomp($x_i^{'}, y_j^{'}$).

10. Combine the new blocks to generate DNA cipher matrix D.

11. Convert matrix D back into decimal number using DNA decoding to get the the final cipher image.

Decryption of the cipher image is parallel to the encryption process. At the receiver end, secret keys are obtained from sender and the coefficients of Bcomp matrix in step 5 are complemented using complement rules as suggested in [11]. DNA addition operation in step 3 is replaced by DNA subtraction operation with other steps kept unchanged.

4 Advantages of proposed image encryption

In Section 1, it has been pointed out that the original image encryption scheme suggested in [25] is found to be non-invertible and prone to known plain text attack by H.Hermassi at el. [25]. In this section, it is being proved that the proposed encryption technique is invertible and it can resist known plain text attack.

4.1 Invertibility

Following steps are used for decryption process -

1. Transform the cipher image of size $m \times n$ into DNA encoded matrix of size $m \times 4n$.
2. Divide the DNA encoded matrix into blocks of size 4×4 and perform the shuffle operation $Block(i, j) \rightleftharpoons Block(x_i^{'}, y_j^{'})$ where $x_i^{'}$ and $y_j^{'}$ are location values in sequences

X' and Y' obtained in step 8 in encryption process. After combining all blocks, Bcomp matrix of size (m, 4n) is obtained.
3. Perform complement rule on the coefficients of Bcomp to obtain Temp matrix using complementary matrix Z(m, 4n).
4. Generate mask matrix M(m, 4n) as in step 2 of encryption process and obtain matrix Bb(m, 4n) using DNA subtraction as follows - Bb = Temp - M
5. Perform DNA decoding operations on Bb as shown in Fig. 1 to get back the original plain image.

Therefore, the proposed image encryption scheme is fully invertible in nature.

4.2 Robustness to known plain text attack

In the original encryption scheme, the complementary matrix is found to be recoverable using chosen plain text in which the known input image with all pixel intensities as zero is taken. Such input image is chosen to make it invariant to the DNA addition operation in the original image encryption technique. In the proposed method of this paper, since the chaotic sequence based masked image M is used for addition operation, one will get the masked image M after addition for input with all zero pixel values and M is unknown to attacker.

To further explain, consider the following example in which DNA encoded blocks of size 4 × 4 of input image P with all zero pixel values as well as mask M is taken.

$$P_{Block} = \begin{bmatrix} C & C & C & C \\ C & C & C & C \\ C & C & C & C \\ C & C & C & C \end{bmatrix}$$

$$M_{Block} = \begin{bmatrix} A & T & C & G \\ G & C & A & A \\ T & A & C & G \\ G & T & A & C \end{bmatrix}$$

After addition operation, corresponding block of Temp matrix will be $Temp_{Block} = P_{Block} + M_{Block}$, which is same as M_{Block}. Similar is the case for remaining blocks of Temp matrix and therefore after DNA addition, Temp matrix will be same as mask M.

Since there are 6 group complementary rules and any of them can be used to generate complement matrix, total possible number of complement matrices can be six as well. Further, Bcomp matrix in encryption process is divided into 4×4 blocks, there are total $\frac{mn}{4}$ such blocks and these blocks are swapped using chaotic sequences. Hence, for a DNA encoded image of size $m \times 4n$, there are total $\frac{mn}{4}!$ permutation of swapping. So to retrieve Temp matrix of step 3 in encryption process, it requires $\frac{mn}{4}! \times 6$ operations provided we know the DNA encoding of image. In other words, for an input image of size 256 × 256, no. of operations required are (64 × 256)! × 6.

Further, since the generation of complement matrix Z and sequences X and Y in Section 3.2 is input image dependent, their values will be different for different input images. Therefore, it is very difficult to hack the encryption information for an attacker using known plain text attack.

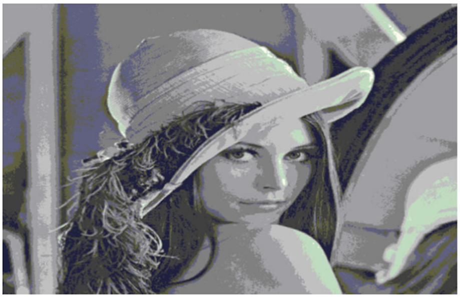


Fig. 4 Image before encryption

5 Experimental results and analysis

For experimental results, the image database of USC-SIPI (available at http://sipi/usc.edu/database/) maintained by University of South California has been used. Images of different content and size have been used in this paper. Values of parameters x_1, y_1, r_1, r_2, r_3 and r_4 have been chosen as 0.62, 0.12, 3.2, 3, 3.9 and 3.85 respectively as in the original paper. Performance evaluation of the proposed image encryption algorithm is done in terms of resistance to statistical attack using parameters like histogram analysis and correlation coefficient analysis and resistance to differential attack using parameters like number of pixel change rate (NPCR) and unified average changing intensity (UACI). Further, its performance has been compared with other encryption techniques with respect to the correlation coefficients between the plain and cipher images. Key sensitivity and key space analysis of the proposed technique has also been performed. In the last, proposed algorithm is analysed for entropy measurement and robustness in terms of attacks on encrypted images with other encryption techniques. Simulation work has been carried out on a computing device with intel core i3-370M processor with 2.4 GHz speed and 3MB L3 cache and 4GB RAM on MATLAB 7.4 using window-7 platform.

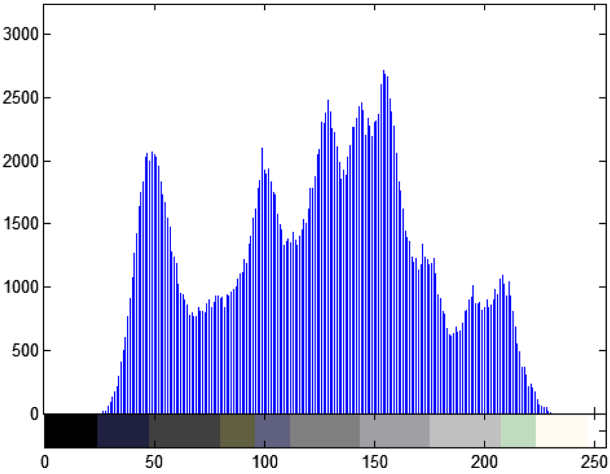
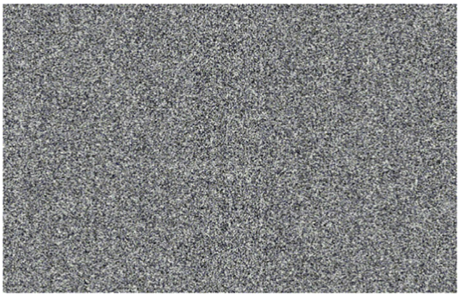


Fig. 5 Histogram of the plain image

Fig. 6 Image after encryption



5.1 Histogram analysis

Histogram of an image represents the intensity distribution of pixels and it is used to identify the no. of pixels with same intensity value. Histogram of an encrypted image must be different from the histogram of the original image and it should have almost uniform distribution of pixel values.

Figure 4 shows the standard Lena image followed by its histogram in Fig. 5 whereas Fig. 6 shows the corresponding encrypted image and its histogram in Fig. 7. It can be seen that the distribution of pixels in the histogram of encrypted image is almost uniform and it is very different from the pixels distribution in the histogram of the original image. Moreover, the image information in Fig. 4 is not at all relevant with that of its encrypted version in Fig. 6.

5.2 Correlation coefficient analysis

Correlation coefficient between two images has been calculated using (7).

$$Corr = \frac{N\sum_{i=1}^{N}(x_i \times y_i) - \sum_{i=1}^{N} x_i \times \sum_{i=1}^{N} y_i}{\sqrt{(N\sum_{i=1}^{N} x_i^2 - (\sum_{i=1}^{N} x_i)^2) \times (N\sum_{i=1}^{N} y_i^2 - (\sum_{i=1}^{N} y_i)^2)}} \tag{7}$$

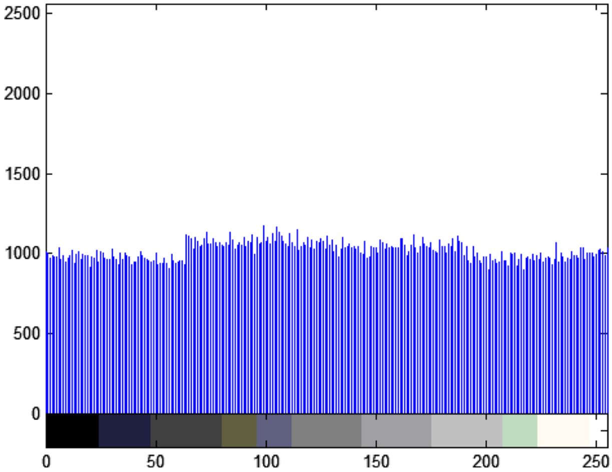


Fig. 7 Histogram of the encrypted image

Table 3 Average correlation coefficients of two adjacent pixels

Image Type	original correlation (plain image)	encryption [19]	encryption [25]	encryption (proposed technique)
5.1.11 (256 x 256)	0.9288	0.0032	0.0772	0.0017
5.1.13 (256 x 256)	0.8334	0.0074	0.0693	0.0032
5.2.09 (512 x 512)	0.8547	0.0033	0.0205	0.0014
Lena gray (512 x 512)	0.9433	0.0026	0.0033	0.0015
Numbers.512 (512 x 512)	0.6945	0.0000	0.0516	0.0000
Testpat.1k (1024 x 1024)	0.7521	0.0005	0.0342	0.0003
5.3.02 (1024 x 1024)	0.8998	0.0013	0.0347	0.0006

where x_i and y_i are the corresponding pixels in two images and N is the total number of pixels in the images used for computing correlation.

Figure 8 and Fig. 9 show the correlation of two horizontally adjacent pixels in original Lena image and its encrypted version. It can be seen that pixels are highly correlated in plain image whereas this correlation is drastically reduced in the encrypted image. Further, Fig. 10 and Fig. 11 show the correlation for the same image between vertically adjacent pixels and it can be observed that the correlation of pixel in Fig. 11 is very poor.

Table 3 shows the average correlation coefficient values (computed using horizontal, vertical and diagonal values) for input images, encrypted images using [19], [25] and the proposed encryption scheme. It can be seen that there is an average decrement of 54 % and 97 % in the value of correlation coefficients in the proposed technique than the suggested in [19] and [25] respectively.

5.3 Differential attack analysis

Proposed image encryption scheme has been analyzed for differential attack on Lena image using two parameters - number of pixel change rate (NPCR) and unified average chang-

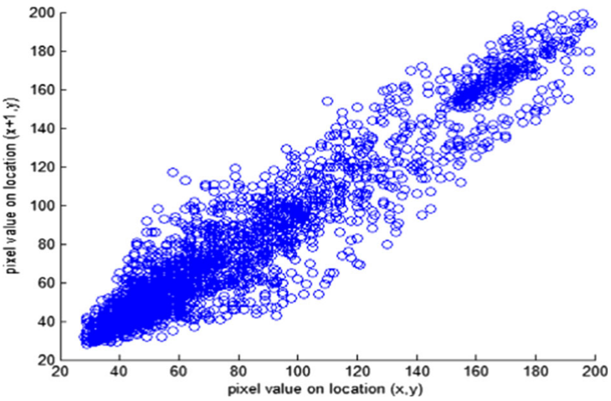


Fig. 8 Correlation of horizontally adjacent pixels in plain image

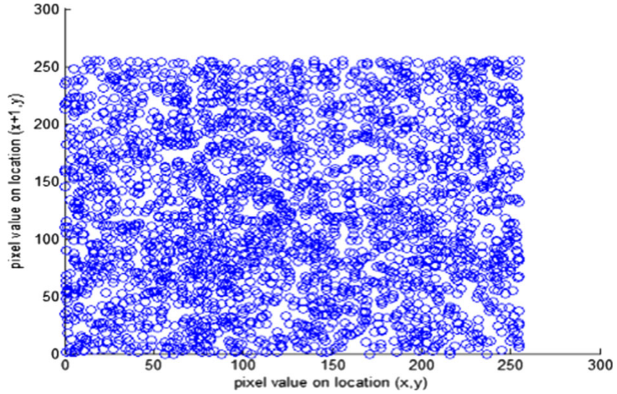


Fig. 9 Correlation of horizontally adjacent pixels in cipher image

ing intensity (UACI) which are defined for an image size of $m \times n$ using .(8) and (9) respectively.

$$NPCR = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} g(i,j)}{m \times n} \tag{8}$$

$$UACI = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} |I_1(i,j) - I_2(i,j)|}{255 \times m \times n} \tag{9}$$

where $g(i,j)$ in (8) is

$$g(i,j) = \begin{cases} 0, & if\ I_1(i,j) = I_2(i,j) \\ 1, & if\ I_1(i,j) \neq I_2(i,j) \end{cases} \tag{10}$$

and I_1 and I_2 denote the encrypted images of plain image before and after modification in plain image. For analysis, 1^{st} pixel of input image has been changed to get the encrypted

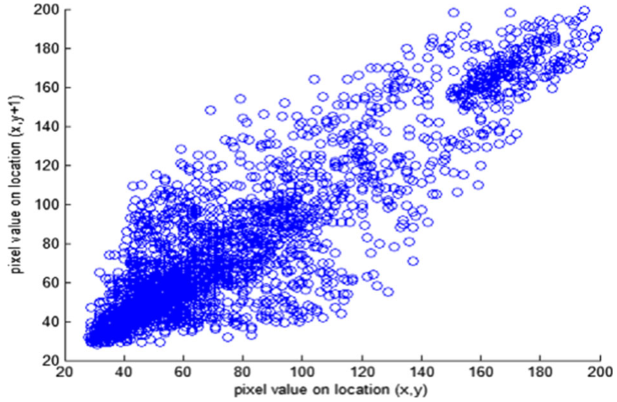


Fig. 10 Correlation of vertically adjacent pixels in plain image

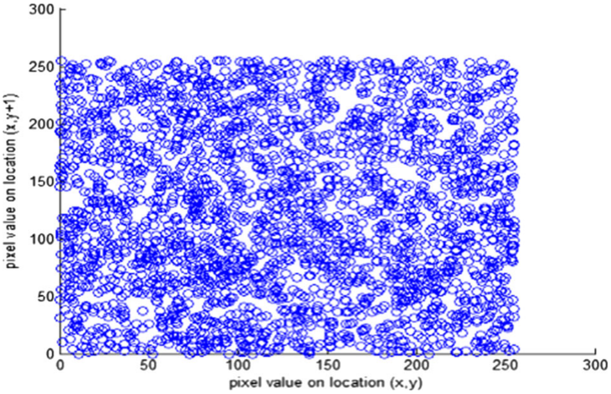


Fig. 11 Correlation of vertically adjacent pixels in cipher image

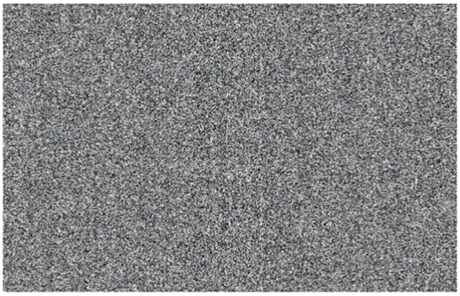
image I_2. In simulation, values of NPCR and UACI has been found to be 99.62% and 33.06% respectively.

5.4 Key sensitivity and key space analysis

The proposed image encryption scheme uses $(x_1, y_1, r_1, r_2, r_3, r_4)$ as secret key and as discussed in the beginning of this section values of these parameters have been chosen as (0.62,0.12,3.2,3,3.9,3.85). After slightly changing the value of parameter x_1 to 0.620000000001 and keeping remaining parameters same, Fig. 12 shows the decrypted image of Fig. 4 using modified secret key and Fig. 13 shows its histogram. It can be seen that the decrypted image is entirely different from the original image and its histogram is almost uniformly distributed. Similarly, by changing other parameters of secret key it has been found that decrypted image is very different from the original one. It proves that proposed encryption technique is robust for exhaustive attack.

Since there are six parameters used in secret key in the proposed image encryption scheme, if the precision of 10^{-12} is chosen for each parameter, the size of secret key space will be $10^{12} \times 10^{12} \times 10^{12} \times 10^{12} \times 10^{12} \times 10^{12} = 10^{72}$ which is large enough for exhaustive attack. Further, if parameters s_1 and s_2 in (2) are also kept secret and a precision of 10^{-12} is used for them, then the key space will expand upto 10^{96}.

Fig. 12 Image after decryption with secret key (0.620000000001,0.12,3.2,3,3.9,3.85)



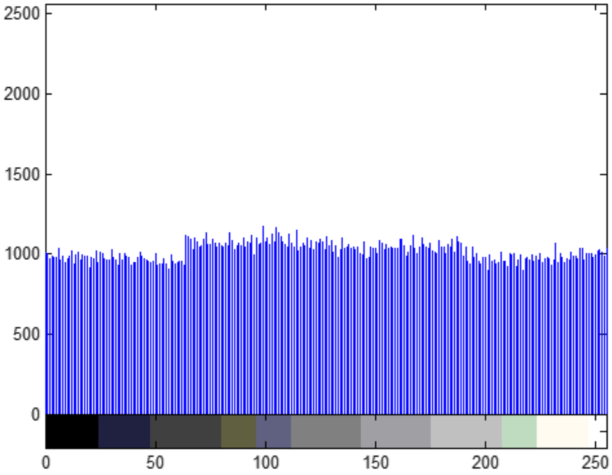


Fig. 13 Histogram of the decrypted image

5.5 Analysis of entropy

Entropy of a message is used to define the quantum of randomness present in that message [17]. For a message A, consisting of q symbols $(a_0, a_1, ..a_i, ...a_{q-1})$ with the probability of occurrence of symbol a_i as p_i is defined as

$$H(A) = -\sum_{i=1}^{q-1} p_i log_2 p_i \tag{11}$$

For an 8-bit gray image, q = 256 and the permitted values of each symbol a_i are 0 to 255. As can be seen from (11), maximum entropy of such an image will be 8 when all the pixels are equally distributed. Table 4 gives the comparative study of the entropy of encrypted images using [19], [25] and the proposed algorithm. It can be seen that values of entropies in the encrypted image using proposed technique is very close to the ideal entropy value 8.

Table 4 Entropy comparison of encrypted images among different encryption techniques

Image Type	encrypted image [19]	encrypted image [25]	encrypted image (proposed technique)
5.1.11 (256 x 256)	7.9912	6.4296	7.9940
5.1.13 (256 x 256)	7.8252	7.2896	7.9642
5.2.09 (512 x 512)	7.9971	6.8091	7.9976
Lena gray (512 x 512)	7.9992	7.9980	7.9994
Numbers.512 (512 x 512)	7.9987	7.0567	7.9988
Testpat.1k (1024 x 1024)	7.9764	7.0542	7.9952
5.3.02 (1024 x 1024)	7.9951	7.1788	7.9963

Table 5 PSNR (dB) and SSIM index comparison of different encryption techniques

Attack Type	Parameter	PSNR [19]	PSNR (proposed)	SSIM [19]	SSIM (proposed)
Gaussian noise	mean 0, variance 0.01	15.2	28.6	0.1316	0.3627
Gaussian noise	mean 0, variance 0.05	12.4	27.8	0.0707	0.2398
Gaussian noise	mean 0, variance 0.1	11.5	27.7	0.0513	0.1914
Salt and paper noise	intensity 0.05	22.2	40.3	0.5023	0.7019
Salt and paper noise	intensity 0.08	20.0	38.3	0.3634	0.6075
Salt and paper noise	intensity 0.1	19.1	37.3	0.3110	0.5623
Block loss	7 blocks	29.2	46.8	0.9676	0.9681
Block loss	15 blocks	25.6	43.7	0.9354	0.9368

5.6 Robustness Analysis

Robustness analysis of image encryption techniques has been performed by using the encrypted image and attacking it through various operations and reconstructing the original image using respective image decryption techniques. For operations to attack the encrypted image, Gaussian white noise contamination, salt and paper noise contamination and block loss have been used. Standard lena image (512x512) is taken as input and noise contaminations on encrypted image are performed for different parameters. For block loss operation, randomly located blocks of size 20x20 are removed from the encrypted image. Table 5 shows the quantitative analysis of encryption techniques in terms of peak signal to noise ratio (PSNR) and structural similarity (SSIM) index. MATLAB code of SSIM index from [20] designed by Z.Wang at el. [22] has been used with its default parameters. Since image encryption technique in [25] is non invertible, it has not been used for robustness analysis.

5.7 Computational cost

Computational cost of the proposed image encryption algorithm has been evaluated in terms of execution time measured in seconds. Table 6 shows the comparative study of execution time for the proposed technique with that of [19] and [25]. Though the time taken by the proposed algorithm is larger than those presented in [19] and [25] but it is very effective with reference to other parameters discussed above. Also with the development of DNA computers equipped with massive parallel computing, time taken for the proposed technique will be negligible.

Table 6 Performance comparison in terms of execution time (sec)

Image size	time [19]	time [25]	time (proposed)
256x256	0.035	0.063	0.078
512x512	0.065	0.156	0.195
1024x1024	0.110	0.398	0.510

6 Conclusion

In this paper, a new image encryption scheme using DNA operations and chaotic maps has been proposed which is a modification of encryption technique proposed in [25]. It generates a mask matrix using 1D chaotic map which is DNA added with the DNA encoded input image and the intermediate result is complemented with the help of a complement matrix generated through two 1D chaotic maps and finally the result is permuted using a 2D chaotic map. Proposed technique is not only invertible but it is robust to various attacks like known plain text attack, statistical attacks and differential attacks.

Acknowledgments Authors are very thankful and obliged for the valuable comments and suggestions of anonymous reviewers and to the editor in chief for his smooth coordination.

References

1. Belazi A, Hermassi H, Rhouma R, Belghith S Algebraic analysis of a RGB image encryption algorithm based on DNA encoding and chaotic map, Journal of Non Linear Dynamics, (online February 2014) doi:10.1007/s11071-014-1263-y
2. Bourbakis N, Alexopoulos C (1992) Picture data encryption using SCAN patterns. J. Pattern Recognit 25(6):567–581
3. Chen G, Mao Y, Chui CK (2004) A symmetric image encryption scheme based on 3D chaotic cat maps. Journal of Chaos Solitons and Fractals 21(3):749–761
4. Enzeng D, Zengqiang C, Zhuzhi Y, Zaiping C (2008) A chaotic image encryption algorithm with the key mixing proportion factor, International Conference on Information Management, Innovation Management and Industrial Engineering, Taipei, (19-21 December)
5. Fridrich J. (1998) Symmetric ciphers based on two dimensional chaotic maps. Journal of Bifurcation and Chaos 8(6):1259–1284
6. Gehani A, LaBean TH, Reif JH (2000) DNA based cryptography. Dimacs Series of Discrete Mathematics and Theoretical Computer Science 54:233–249
7. Hermassi H, Belazi A, Rhouma R, Belghith SM Security analysis of an image encryption algorithm based on a DNA addition combining with chaotic maps, Journal of Multimedia Tools and Applications, (online June 2013) doi:10.1007/s11042-013-1533-6
8. Jain A, Rajpal N (2012) A two layer chaotic network based image encryption technique, IEEE National Conference on Computing and Communication System, Durgapur, India, 21-22 November
9. Jain A, Rajpal N (2013) Adaptive key length based encryption algorithm using DNA approach, IEEE International Conference on Machine Intelligence Research and Advancement, Katra, Jammu, India, 21-23 December
10. Kocarev L (2002) Chaos based cryptography: a brief overview. IEEE Magazine on Circuits and Systems 1(3):6–21
11. Liu H, Wang X, Kadir A (2012) Image encryption using DNA complementary rule and chaotic maps. Journal of Applied Soft Computing 12(5):1457–1466
12. Liu L, Zhang Q, Wei X (2012) A RGB image encryption algorithm based on DNA encoding and chaos map. Journal of Computers and Electrical Engineering 38(5):1240–1248
13. Liu HJ, Zhu ZI, Ziang HY, Wang BI (2008) A novel image encryption algorithm based on improved 3D chaotic cat map, IEEE International Conference for Young Computer Scientists, Hunan, (18-21 November)
14. Liu Y, Tang J, Xie T (2014) Cryptanalyzing a RGB image encryption algorithm based on DNA encoding and chaos map. Journal of Optics and Laser Technology 60:111–115
15. Mao YB, Chen G, Lian SG (2004) A novel fast image encryption scheme based on 3D chaotic baker map. Journal of Bifurcation and Chaos 14(10):3613–3624
16. Pareek NK, Patidar V, Sud KK (2006) Image encryption using chaotic logistic map. J Image Vis Comp 24(9):926–934
17. Shannon CE (1949) Communication Theory of security systems. Bell System Technical Journal 28:656–715

18. Scharinger J (1998) Fast encryption of image data using chaotic Kolmogrov flow. Journal of Electronic Engineering 7(2):318–325
19. Tang Z, Zhang X, Lan W (2014) Efficient image encryption with block shuffling and chaotic map. Journal of Multimedia Tools and Applications. doi:10.1007/s11042-014-1861-1
20. The SSIM Index for image quality assessment [online] Available. http://www.cns.nyu.edu/lev/ssim/
21. Wang Y, Wong KW, Liao X, Chen G (2011) A new chaos-based fast image encryption algorithm. Journal of Applied Soft Computing 11(1):514–522
22. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing 13(4):600–612
23. 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. Journal of System Software 85:290–299
24. Zhang Q, Xue X, Wei X (2012) A novel image encryption algorithm based on DNA subsequence operation. The Scientific World Journal 2012:1–10. doi:10.1100/2012/286741
25. Zhang Q, Guo L, Wei X (2010) Image encryption using DNA addition combining with chaotic maps. Journal of Mathematical and Computer Modeling 52:2028–2035

Anchal Jain has received her B.Tech in Computer Science and Engineering from CCS University Meerut India and M.Tech in Computer Science from U.P. Technical University Lucknow, India. She is currently working as Associate Professor in the department of Computer Science & Engineering, Inderprastha Engineering College, Ghaziabad. She is now pursuing her Ph.D. at School of Information and Communication Technology, GGSIP University, Delhi, India. Her research interests are secure multimedia processing, video coding and soft computing approaches of cryptography.

Navin Rajpal is professor of USICT since September 2004 and he has been Dean of the school from 1st October 2011 to 30th September 2014. He did his B. Sc. (Engineering) in Electronics and Communication from R.E.C. Kurukshetra, now known as NIT, Kurukshetra. He did his M. Tech and Ph.D. from Computer

Science & Engineering Department, IIT, Delhi. He served in various capacities and has more than 24 years of experience in teaching and research. He has worked as Senior Scientific Officer for more than eight years at Centre for Applied Research in Electronics IIT Delhi on various sponsored and Consultancy projects. Before joining this university in July 2000 as Reader, he worked for more than 4 years as Assistant Professor at C.R. State College of Engineering, Murthal, where he was Incharge Computer Science and Engineering Department for about two years. In G.G.S. Indrapratha University he has worked as Reader, USICT and Incharge Computer Center from July 2000 to August 2004. He has also worked as Head CS&E at IGIT from Jan. 2005 to Dec. 2007. He has supervised several M. Tech. and three Ph.D. Students. He has published / presented more than 75 research papers in National and International Journals / Conferences. He is a life member of CSI and ISTE. His areas of interest are Computer Vision, Image Processing, Pattern Recognition, Artificial Neural Networks, Computer Graphics, Algorithms Design and Digital Hardware Design.