Secure And Verifiable Cryptographic Scheme Using Rubik's Cube Principle

B.Nagarajan
PG Scholar, Department of Computer Science and Engineering, RVS Faculty of Engineering, Coimbatore, TamilNadu - 641 402.

B.Manju
Assistant Professor, Department of Computer Science and Engineering, RVS Faculty of Engineering, Coimbatore, TamilNadu - 641 402.

Abstract – With the rapid development of the networked multimedia, communication and propagation techniques, the trend of sending or receiving the digital data, especially images has greatly increased. To protect the privacy of the authorized users and to guarantee the legal data access, security is an important issue in communication and storage of images. Encryption is one way to ensure the security. The original image is scrambled using the bit level permutation based on DNA encoding to confuse the relationship between original and encrypted images. Then genetic operators like crossover, mutation are applied to rows and columns of the scrambled image. Both this operations are done by using random numbers generated by pseudorandom number generators like LCG and BBS. This proposed image encryption scheme can resist exhaustive attack, statistical attack, and differential attack. For evaluating the performance of the algorithm a series of tests are performed. These tests include information entropy analysis, correlation analysis, and analysis of NPCR and UACI values.

Index Terms – Multimedia, Communication, Propagation, LCG, BBS, NPCR, UACI.

1. INTRODUCTION

Security of multimedia information is used to protect the multimedia content from unauthorized access. Traditionally developed encryption algorithm such as AES, IDEA, and DES Blowfish are suitable for text encryption but not suitable for image encryption directly because of two reasons. One is that the image size is larger than that of text, so the traditional cryptosystems take much time to directly encrypt the image data. The other reason is that the decrypted text must be equal to the original text. However, this requirement is not necessary for image, a decrypted image containing small distortion is acceptable due to human perception. The multimedia information including image data has some special characteristics like high capacity, redundancy and high correlation among pixels. In some cases image applications require to satisfy their own needs like real time transmission and processing. One of the main goals that must be achieved during the transmission of information over the network is security.

Cryptography is the technique that can be used for secure transmission of data. This technique will make the information to be transmitted into an unreadable form by encryption so that only authorized persons can correctly recover the information. The security of image can be achieved by various types of encryption schemes. Different chaos based and non-chaos based algorithms have been proposed. Among this the chaotic based methods are considered to be more promising. The chaotic image encryption method can be developed by using the properties of chaos including deterministic dynamics and unpredictable behaviour. There are three kinds of encryption techniques namely substitution, transposition and permutation and techniques that include both transposition and substitution. Substitution schemes change the pixel values while permutation schemes just shuffle the pixel positions based on the algorithm. In some cases both the methods are combined to improve the security level. Images are used in fields such as medical science, military, social networks, and satellite communication.

2. RELATED WORK

Z.L. Zhu [6] presented image encryption using bit level permutation. As the pixels at different locations are exchanged, this is equivalent to encrypting the coordinate pair. In the diffusion phase, these pixels are considered as a 1-D sequence, and each pixel value is modified one after another according to the chaotic sequence. When these two operations are repeated several times, the pixel vector set has been changed substantially, and the encryption is complete. Confusion operation is done by permute the pixels by the bits within the pixel. Image is divided into 8 binary images. Higher-order bits are more significant than lower-order bits. The binary images formed by the higher 4 bits, are permuted independently using the same chaotic map with different coefficients. On the other hand, the binary images of the lower 4 bits are permuted as a whole to reduce the execution time. This means that an MxN matrix is permuted, where each matrix element is a 4-bit sequence. Each element of the matrix can be considered as a kind of pixel, but it carries the lower 4-bit information only. It

employing the Arnold cat map for bit-level permutation. Liu et al. [7] presented an image encryption scheme based on iterative random phase encoding in gyrator transform domains. A two-dimensional chaotic mapping is employed to generate many random data for iterative random phase encoding.

Qiang Zhang [1] presented a novel image encryption algorithm based on DNA subsequence operation. Different from the traditional DNA encryption methods, the algorithm does not use complex biological operation but just uses the idea of DNA subsequence operations such as elongation operation, truncation operation and deletion operation. Then combining with the logistic chaotic map to scramble the location and the value of pixel points from the image. Chaotic sequences produced by chaotic maps are pseudorandom sequences; their structures are very complex and difficult to be analysed and predicted. S.S.Manickkam proposed image and video encryption using scan patterns [4]. This encryption methods are based on the SCAN methodology which is a formal language-based two-dimensional spatial accessing methodology which can generate very large number of scanning paths or space filling curves. The SCAN family of formal languages includes several versions, each of which can represent and generate a specific set of scanning paths. Each SCAN language is defined by a grammar and each language has a set of basic scan patterns, a set of transformations, and a set of rules to compose simple scan patterns to obtain complex scan patterns. The rules for building complex scan patterns from simple scan patterns are specified by the production rules of the grammar of each specific language. The image encryption method is based on permutation of the pixels of the image and replacement of the pixel values. The permutation is done by scan patterns (encryption keys) generated by the SCAN methodology and the pixel values are replaced using a simple substitution rule. The permutation and substitution operations are applied in intertwined manner and iteratively.

3. GENETIC ALGORITHM

A genetic algorithm (GA) is a search heuristic that mimics the process of natural selection. This heuristic is routinely used to generate useful solutions to optimization and search problems. Genetic algorithms belong to the larger class of evolutionary algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover. Genetic operators used in genetic algorithms maintain genetic diversity. Genetic diversity or variation is a necessity for the process of evolution. Genetic operators are analogous to those which occur in the natural world: Reproduction (or Selection); Crossover (or Recombination); and Mutation.

3.1 Cross Over

In genetic algorithms, crossover is a genetic operator used to vary the programming of a chromosome or chromosomes from one generation to the next. It is analogous to reproduction and biological crossover, upon which genetic algorithms are based. Cross over is a process of taking more than one parent solutions and producing a child solution from them.

One-point crossover: A single crossover point on both parents' organism strings is selected. All data beyond that point in either organism string is swapped between the two parent organisms. The resulting organisms are the children. Example for crossover:

parent1= **AC**CG & parent2=TA**TG**.

After crossover parent1= **TG**CG & parent2=TA**AC**.

Two-point crossover: Two-point crossover calls for two points to be selected on the parent organism strings. Everything between the two points is swapped between the parent organisms, rendering two child organisms. Consider the two parents selected for crossover:

Parent 1

1 1 0 1 1 | **0 0 1 0 0 1 1** | 0 1 1 0

Parent 2

1 1 0 1 1 | **1 1 0 0 0 0 1** | 1 1 1 0

Interchanging the parents chromosomes between the crossovers points - The Offspring produced are:

Offspring 1

1 1 0 1 1 | **0 0 1 0 0 1 1** | 0 1 1 0

Offspring 2

1 1 0 1 1 | **0 0 1 0 0 1 1** | 0 1 1 0

3.2 Mutation

Mutation is a genetic operator used to maintain genetic diversity from one generation of a population of genetic algorithm chromosomes to the next. It is analogous to biological mutation. Mutation alters one or more gene values in a chromosome from its initial state. In mutation, the solution may change entirely from the previous solution. Hence GA can come to better solution by using mutation. A common method of implementing the mutation operator involves generating a random variable for each bit in a sequence. This random variable tells whether or not a particular bit will be modified. This mutation procedure, based on the biological point mutation, is called single point mutation. Other types are inversion and floating point mutation. When the gene encoding is restrictive as in permutation problems, mutations are swaps, inversions and scrambles.

Example:

1 0 1 0 **0** 1 0

↓

1 0 1 0 **1** 1 0

3.3 DNA encoding

A single DNA sequence is made up of four nucleic acid bases: A (Adenine), C (Cytosine), G (Guanine), and T (Thymine), where A and T are complements, and C and G are complements. Let binary number 0 and 1 be complements, then 00 and 11 are complements, and 01 and 10 are complements. Thus use these four bases: A, T, G, and C to encode 01, 10, 00, and 11, respectively. The encoding method still satisfies the Watson-Crick complement rule. Usually, each pixel value of the 8 bit grey image can be expressed to 8 bits binary stream. The binary stream can be encoded to a DNA sequence whose length is 4. For example: if the first pixel value of the original image is 75, convert it into a binary stream [01001011]. By using the above DNA encoding rule to encode the stream, get a DNA sequence [AGTC], whereas we use A, T, G, and C to express 01, 10, 00, and 11, respectively.

3.4 Logic for addition and subtraction of DNA sequences

Addition and subtraction operation for DNA sequences are performed according to traditional addition and subtraction. For example: 11+10=01, 01-11=10. We use 00, 01, 10, 11 to denote A, C, G, and T respectively. That is G+T=A, A-C=G. The details of addition and subtraction rule are shown in figure 1.

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

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

Fig.1 Addition and Subtraction Rule for DNA Sequence

4. THE PROPOSED METHOD

In the proposed method, first, the bit level permutation is applied to the original image to shift the pixel positions. Then the genetic operators are used to diffuse the pixel values. Figure 2 shows the flow diagram of proposed method.

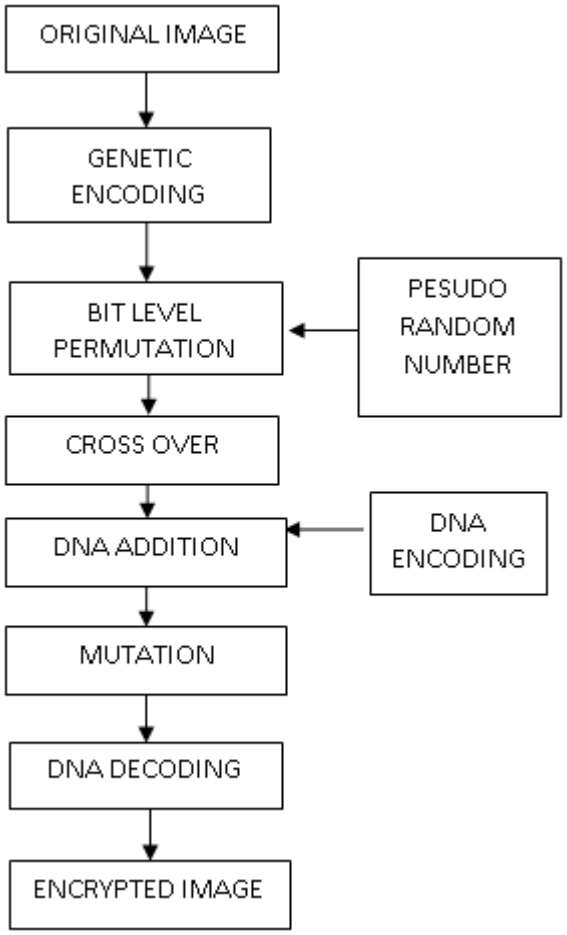


Fig 2. Block Diagram of Proposed Method

4.1 Example for the Proposed Method

The pixel values of original image is encoded into genetic form. Then by using the circular shift, the original image pixel positions are transformed which results in a scrambled image which is entirely different from the original image. The values of key vector Kc= [3 0 1 2] & Kr= [2 3 0 1]. Figure 3 illustrates the proposed method to permute the pixel positions of a digital image. (b) Shows the genetic encoding value for original matrix. (d) Shows value after circular shifting in row wise by K_r position.

40	27	113	241
182	39	124	98
15	254	7	175
147	201	83	255

(a) Original Matrix

GAAG	CATG	TGCT	TGCC
ATCA	CTAG	GGCT	AGAT
CCGG	ACCC	CTGG	CCAA
CGTA	TAGC	CGTT	CCCC

(b) Matrix after DNA Encoding

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

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

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

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

(c) Before Circular Shifting Row wise.

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

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

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

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

(d) After Circular Shifting Row wise.

CAAC	TGCC	TTGG	CCAA
CGGT	GTCA	GTAG	CGTA
CGCT	ACCC	ACAC	CATG
TCCG	CGTT	CAGG	ACCT

(e) After Permutation

235	241	5	175
67	180	36	147
115	254	238	27
61	83	11	126

(f) Scrambled Matrix

Fig 3 Bit level Permutation

4.2 Random Number Generation

Random numbers are useful for various purposes, such as generating data encryption keys, simulating and modelling complex phenomena and for selecting random samples from larger data set. Generation of random numbers consists of two main approaches using computers. They are the Pseudo-Random Number Generators (PRNGs) and True Random Number Generators (TRNGs) [9]. Many computers are built with inputs that digitize some real world analog sources, such as sound from microphone, audio, etc. If the system has enough gain to detect anything, such input can provide reasonable high quality random bits.

4.2.1 Linear Congruential Generator (LCG)

The Security of many cryptographic systems depends upon the generation of unpredictable quantities. A cryptographic algorithm's security rests with its key. Symmetric key technique requires the sender and receiver to use same key

For any data transmissions. The Linear Congruential Generator (LCG) yields a sequence

Of randomized numbers are calculated using the linear equation (1), where Xn is the sequence of pseudorandom values [11].

$X_{n+1} = (aX_n+c) \pmod m$ ⟶ (1)

Where, a is the initial seed value, c is a constant value and m is the modulo value.

4.2.2 Blum Blum Shub

Blum Blum Shub (BBS) is an unpredictable pseudorandom number generator. It provides security to the computational difficulty of the quadratic residuosity problem. The generator is not appropriate for use in simulations, only for cryptography, because it is very slow. The randomized numbers are calculated by using equation (2) [11].

$X_{n+1} = (X_n)^2 \bmod M \longrightarrow (2)$

Where M is the product of two large primes p and q.

4.3 Proposed Encryption Algorithm

The following algorithm is used to encrypt image using genetic operators.

Input: Plain image P of size m x n, seed for random number generator

Output: Cipher image B of size m x n

Step 1: Let the image to be encrypted is Original image [m, n]. Generate randomly two vectors K_R and K_C of length M and N, respectively.

Step2: [P1 P2 P3 P4] ⟵ DNA sequences obtained from image A by DNA encoding.

Step3: for i varies from 1 to M

[R1 R2 R3 R4] ⟵Permute [P1 P2 P3 P4] using circular-shifted by K_R (i) positions

Step4: for j varies from 1 to N

[C1 C2 C3 C4] ⟵ Permute [R1 R2 R3 R4] using circular-shifted by K_C (j) positions

Step5: [S1 S2 S3 S4] ⟵ Crossover operation [C1 C2 C3 C4]

Step6: [K1 K2 K3 K4] ⟵ DNA sequences obtained from key vector K_R by DNA encoding.

Step7: [A1 A2 A3 A4] ⟵ [S1 S2 S3 S4] + [K1 K2 K3 K4] (DNA Addition)

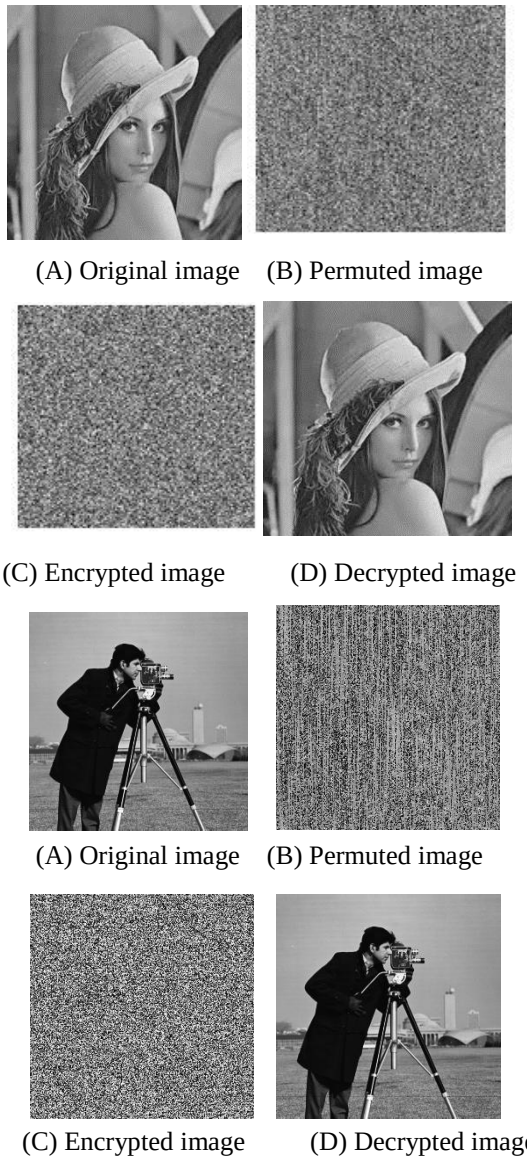
Step8: [A1' A2' A3' A4'] ⟵ Complement (A1 A2 A3 A4)

Step9: B ⟵ carry out DNA decoding and recombining binary bit planes from A1', A2', A3', A4'

Step10: Display the cipher image B.

5. IMPLEMENTATION RESULTS AND ANALYSIS

The algorithm is implemented using Mat lab with Intel(R) core to dual as processor, clock speed of 2.40GHZ, 2GB RAM, 250GB hard disk and Windows 7(32-bit) operating systems. Figures 4 shows the results of image encryption method based on genetic operator using Lena as original image.



(A) Original image (B) Permuted image

(C) Encrypted image (D) Decrypted image

(A) Original image (B) Permuted image

(C) Encrypted image (D) Decrypted image

Fig. 4 Results for Proposed method

5.1 Visual Testing

There is no perceptual similarity between original images and their encrypted counterparts. The encrypted image should greatly differ from its original form. In general, two difference measures such as NPCR and UACI are used to quantify this requirement.

5.1.1 Number of pixel change rate (NPCR)

The first measure is the Number of Pixels Change Rate (NPCR), which indicates the percentage of different pixels between two images. For the plaintext image I_o (*i, j*) and encrypted image I_{ENC} (*i, j*) the equation (3) gives the mathematical expression to compute the NPCR value [9].

$$NPCR = \frac{\sum_{i,j} D(i,j)}{W*H} * 100\% \longrightarrow (3)$$

Where, W and H are the width and height of the images. If $I_o(i, j) = I_{ENC}(i, j)$, then D(i,j) = 0 Otherwise, D(i,j)=1.

5.1.2 Unified Average Changing Intensity (UACI)

A small change in plain image must cause some significant change in cipher image. UACI is helpful to identify the average intensity of difference in pixels between the two images. For the plain image I_o (i, j) and encrypted image I_{ENC} (i, j) the equation (4) gives the mathematical expression to compute the UACI value [9].

$$UACI = \frac{1}{W*H}\left[\sum_{i,j} \frac{Io(i,j) - Ienc(i,j)}{255}\right] * 100\% \quad (4)$$

Where, W and H are the width and height of the images. Table 1 gives NPCR and UACI values of images encrypted using genetic operators. Table 2 gives NPCR and UACI values of images encrypted.

Table 1 NPCR and UACI Values

Images	**NPCR (in %)**		**UACI (in %)**	
	LCG	**BBS**	**LCG**	**BBS**
Lena	99.5072	99.5152	30.2477	30.1124
Cameraman	99.4720	99.5376	30.3288	30.6764
Baboon	99.6824	99.6520	30.3128	30.9327
Coin	99.5802	99.5424	30.9832	31.2373
Rice	99.5691	99.5104	30.1095	30.2129

Table 2 Existing Method Values

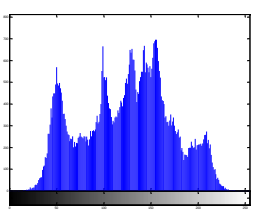
Images	**Ref. [2]**		**Ref. [3]**	
	UACI	**NPCR**	**NPCR**	**UACI**
Lena	29.6201	99.5850	99.6078	30.5903
Cameraman	27.4092	99.6094	-	-
Baboon	29.781	99.4824	-	-

5.2 Statistical Analysis

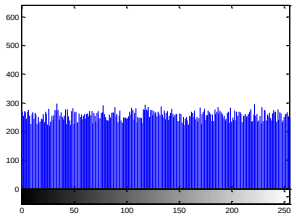
It is possible to break the ciphers by statistical analysis. This is done by analysis the histogram of the encrypted images and the correlation between the adjacent pixels in the encrypted image. In order to check whether the suggested method is safe against statistical attacks, the evaluation parameter such as histogram and correlation are analysed.

5.2.1 Histogram Analysis

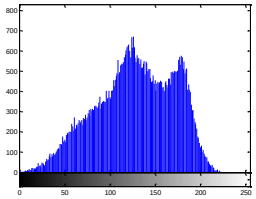
To prevent the access of information to attackers, it is important to ensure that encrypted and original images do not have any statistical similarities. The histogram analysis clarifies that, how the pixel values of image are distributed. The histogram of original image contains great rises followed by sharp declines and the histogram of the encrypted image has uniform distribution which is different from the original image. Figure 5 shows the histogram of the Lena and baboon images before and after encryption.



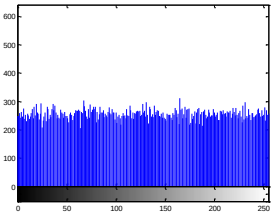
(a) Lena



(b) Encrypted Image



(c) Baboon



(d) Encrypted Image

5.2.2 Correlation Coefficient

Correlation computes the degree of similarity between two objects. This parameter is useful for calculating the quality of the cryptosystem. An arbitrarily chosen pixel in an image is generally strongly correlated with adjacent pixels, and it's in either horizontal, vertical or diagonal directions. A secure image encryption algorithm must produce an encrypted image having low correlation between adjacent pixels.

$$\text{Correlation co-efficient} = \frac{Cov(x,y)}{\sigma x * \sigma y} \longrightarrow (5)$$

$$\sigma_x = \sqrt{VAR(x)}$$

$$\sigma_y = \sqrt{VAR(y)}$$

$$VAR(x) = \frac{1}{N}\sum_{i=1}^{N}(xi - E(x))^2$$

$$Cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}(xi - E(x))(yi - E(y))$$

Where, Cov(x,y) is the covariance between x and y; x,y are values of adjacent pixels in image, N is the number of pixel pairs (xi, yi), and E(x) and E(y), are the mean values of xi and yi respectively. Table 3 gives the adjacent pixel correlation

values obtained by the proposed method and existing methods. The cross correlation value shows that there is no exact relationship between the original and encrypted images. Table4 gives the correlation value between original and encrypted images.

5.3 Entropy Analysis

For grey-scale images of 256 levels, if each level of grey is assumed to be equiprobable, then the entropy of this image will be theoretically equal to 8 Sh (or bits). Ideally, an algorithm for encryption of images should give an encrypted image having equiprobable grey levels. The entropy values of original images are far from ideal value of entropy since information sources are highly redundant and thus rarely generate uniformly distributed random messages. On the other hand, the entropy values of the encrypted images are very close to the ideal value of 8 Sh, which means that the encryption algorithm is highly robust against entropy attacks.

$$H(m) = \sum_{i=0}^{m-1} p(mi) log\ (1/p(mi)) \quad (6)$$

Where, m is the total number of symbols mi € m; p (mi) represents the probability of occurrence of symbol mi and log denotes the base 2 logarithm. Table 5 gives the entropy values of the original images and those of their encrypted versions.

Table 3 Adjacent Pixel Correlation Values

Images	**Horizontal**	**Vertical**	**Diagonal**
Original image: Lena	0.9846	0.9756	0.9096
Encrypted image [LCG]	0.0059	0.0031	0.0056
Encrypted image [BBS]	0.0023	-0.0137	0.0035
Ref. [2]	0.0068	0.0091	0.0063
Ref.[3]	0.0006	0.0002	0.0043
Original image : Baboon	0.4265	0.5022	0.4340
Encrypted image: [LCG]	0.0086	0.0015	0.0047
Encrypted image: [BBS]	-0.0549	-0.1032	0.0021
Ref. [2]	0.0055	0.0078	0.0042
Original image : Cameraman	0.9875	0.9949	0.9845
Encrypt image [LCG]	0.0053	0.0040	0.0098
Encrypted image [BBS]	0.0019	-0.0083	0.0048

Table 4 Cross Correlation Values

Images	**Original image (Sh)**	**Encrypted image (LCG)**	**Encrypted image (BBS)**	**Ref. [2]**
Lena	7.4579	7.9968	7.9891	7.9968
Baboon	7.2279	7.9972	7.9977	7.9974
Cameraman	7.2993	7.9988	7.9998	7.9966
Coin	7.1531	7.9385	7.9982	-
Rice	7.3806	7.9723	7.9993	-

Images	**LCG**	**BBS**
Lena	0.0028	0.0025
Baboon	0.0019	-0.0040
Cameraman	0.0048	-0.0019
Coin	0.0036	-0.0249
Rice	0.0021	-0.0067

Table 5 Entropy values

6. CONCLUSION

In this paper, an image encryption algorithm based on the genetic operators is implemented and results are analysed. The original image is scrambled using the bit level permutation based on DNA encoding to confuse the relationship between original and encrypted images. Then genetic operators like crossover, mutation are applied to the scrambled image. Both this operations are done by using random numbers generated by pseudorandom number generators like LCG and BBS. The NPCR and UACI values show that encrypted image is completely different from the original image by high pixel change rate. The entropy values of the encrypted images are very close to the ideal value of 8 Sh, which means that the encryption algorithm is highly robust against entropy attacks.

The histogram of the encrypted image is flat and different from the original image. The performance assessment tests demonstrate that the image encryption algorithm is highly secure and suitable for real time applications.

REFERENCES

[1] Young-Chang Hou , Shih-Chieh Wei, Chia-Yin-Lin, "Random-Grid-Based Cryptography Schemes" ,vol.24, Issue:5, May 2014
[2] Qiang Zhang, Xianglian Xue and Xiaopeng Wei, "A Novel Image Encryption Algorithm Based on DNA Subsequence Operation'', the Scientific World Journal, Volume 2012, Article ID 286741, 2012.
[3] Khaled Loukhaoukha, Jean-Yves Chouinard and Abdellah Berdai, "A Secure Image Encryption Algorithm Based on Rubik's Cube Principle", Journal of Electrical and Computer Engineering, Article ID 173931, 2012.
[4] Adrian Viorel Diaconu and Khaled Loukhaoukha, "An Improved Secure Image Encryption Algorithm Based on Rubik's Cube Principle and Digital Chaotic Cipher", Hindawi Publishing Corporation, Mathematical Problems in Engineering, 2013.
[5] S.S. Maniccam and N.G. Bourbakis, "Image and Video Encryption using SCAN Patterns", Pattern Recognition Journal, pp 725-737, 2003.
[6] Mitra Y. V. Subba Rao and S. R. M. Prasanna, "A New Image Encryption Approach using Combinational Permutation Techniques", International Journal of Electrical and Computer Engineering, 2006.
[7] Z.L. Zhu, W. Zhang, K.W. Wong, and H. Yu, "A Chaos Based Symmetric Image Encryption Scheme using a Bit-level Permutation", Information Sciences, vol. 181, no. 6, pp. 1171–1186, 2011.
[8] Mohammad Ali Bani and Aman Jantan, "Image Encryption Using Block-Based Transformation Algorithm", International Journal of Computer Science, 2008.
[9] Zhengjun Liu, LieXu, ChuangLin, JingminDai and ShutianLiu, "Image Encryption Scheme by using Iterative Random Phase Encoding in Gyrator Transform Domains", Optics and Lasers in Engineering, pp 542–546, 2011.
[10] Yue Wu, Josep P. Noonan and Sos Again, "NPCR and UACI Randomness Tests for Image Encryption", Journal of Selected Areas in Telecommunication, pp 31-38, 2011.
[11] Qing Guo, ZhengjunLiu and ShutianLiu, "Color Image Encryption by using Arnold and Discrete Fractional Random Transforms in IHS Space", Optics and Lasers in Engineering, pp 1174–1181, 2010.
[12]William Stalling, "Cryptography and Network Security – Principles and Practices", Pearson Education, New Delhi, 2013.
[13] V.Srikanth, Udit Asati, Teja Mullapudi and N. Iyengar, "Bit_Level Encryption of Images using Genetic Algorithm", International Journal of Computing Science and Communication Technologies, Vol.3, pp 546-550, July 2010.
[14]W.C.Chen, Z.Y.Chen, Z.H.Chenetal, "Operational Rules of the Digital Coding of DNA Sequences in High Dimension Space," Acta Biophysica Sinica, vol.17, pp.542–549, 2001