IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 6, 2019, accepted November 17, 2019, date of publication November 27, 2019,
date of current version December 16, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2956389

An RGB Image Cipher Using Chaotic Systems, 15-Puzzle Problem and DNA Computing

NADEEM IQBAL[1,2], SAGHEER ABBAS[1], MUHAMMAD ADNAN KHAN[1,3], TAHIR ALYAS[3], AREEJ FATIMA[3], AND AIESHA AHMAD[4]

[1]Department of Computer Science, NCBA&E, Lahore 54660, Pakistan
[2]School of Computing and Information Sciences, Imperial College of Business Studies, Lahore 53720, Pakistan
[3]Department of Computer Science, Lahore Garrison University, Lahore 54810, Pakistan
[4]Department of Computer Science, NCBA&E, Multan 60700, Pakistan

Corresponding author: Nadeem Iqbal (nadeem.iqbal537@gmail.com)

This work was supported by the Higher Education Commission, Pakistan, under Grant 085-14317-Eg5-169.

ABSTRACT The application of confusion and diffusion processes on the three individual components of an RGB image is not secure and efficient, so this problem needs to be addressed. In this paper, a novel RGB image cipher is proposed using chaotic systems, 15-puzzle artificial intelligence problem and DNA computing. First of all the given color image is decomposed into its red, green and blue gray scale images. Then these gray scale images are concatenated to make a single gray scale image. This single gray scale image is further divided into different blocks. A block level permutation (BLP) is proposed on this gray scale image by using the 15-puzzle problem. A pixel level permutation is applied to further randomize the image pixels. This confused image is then DNA encoded. Afterwards, a diffusion process is applied on this DNA encoded image. Lastly this DNA diffused image is converted back into the decimal. Further, this single gray scale image is broken into three gray scale images. These three images are combined to get the final color cipher image. To create the plaintext sensitivity, SHA 256 hash function has been used. Both the simulation and a comprehensive security analyses suggest the robustness and the impregnability of the proposed scheme which in turn signals towards the real world applicability of the scheme.

INDEX TERMS Image processing, chaos, encryption, decryption, DNA computing.

I. INTRODUCTION

Current era is characterized by the ubiquity of images. These images are found in virtually the entire spectrum of human existence. For instance, there are biological images, medical images, satellite images, military images, commercial images, social images, advertising images to name a few. These digital images are frequently stored on diverse gadgets like hard disks, portable storage devices, cloud servers and PDAs etc. Apart from that, we frequently transmit them over the public networks like the Internet. Sometimes, these images are very sensitive and confidential. This frequent storage and transmission of images is not void of threat from the potential hackers and intruders. Therefore, their safety from the unauthorized access is a very essential and an urgent challenge. Traditionally, the integrity of these images has been maintained by turning them into some noisy version by using some encryption technique.

Digital images do have large data capacities, strong correlation between the adjacent pixels and a high redundancy, so the traditional cryptosystems like RSA, DES and AES are humble to do the job of encryption [1], [2]. Whereas, the chaotic systems/maps proved very helpful for the image encryption schemes due to their intrinsic properties of pseudo-randomness, unpredictability, high sensitivity to the initial conditions and system parameters, large key space, plaintext sensitivity, mixing and ergodicity etc. Using these chaotic maps/systems, hundreds of image ciphers have been developed during the last two decades [1]–[17]. Despite the afore-mentioned excellent properties of the chaotic systems, a host of image ciphers have been successfully cracked owing to having some weakness in them [18]–[20]. On the one hand, cryptographers are trying their level best to develop more and still more robust and strong ciphers whereas on the other hand, cryptanalysts happen to be very smart in breaking these

The associate editor coordinating the review of this manuscript and approving it for publication was Shiqi Wang.

ciphers by exploiting different vulnerabilities in them. So the battle between the cryptography and cryptanalysis seems to remain unabated. Therefore, the given circumstances dictate to develop still more robust, smarter and more foolproof image ciphers. Normally, chaotic maps are classified into low dimension or higher dimension. The former maps consist of one or two dimensions whereas the latter ones on more than two dimensions. Each has its pros and cons. Many cryptosystems were developed using low dimensional chaotic maps/systems [11], [21], [22]. Although it is very easy to implement low dimensional chaotic system but at the same time they can't deliver us the requisite security standards. As far as the image ciphers based upon the higher dimensional chaotic maps are concerned [7], [10], [23], it is difficult to implement them but they give us more random and chaotic data and therefore provide the higher security standards. So a suitable trade-off is required between the implementation of these two kinds of maps.

In 1998, Fridrich proposed a pioneering architecture of pixel level confusion and diffusion [24] that was adopted widely to build image encryption schemes [7]–[10], [14]–[16], [23], [25]–[33]. In these schemes, the confusion and diffusion operations were conducted multiple times to get the high security. For instance, in [9], a key stream was created by using the nonlinear Chebyshev function and then a multiple permutations of the pixels was done to decrease the strong correlation between the adjacent pixels. Further Hua *et al.* [25] developed a 2D Sine Logistic modulation map, a blend of the Logistic and Sine maps. In this particular study, a chaotic magic transform was proposed for modifying the image pixel positions in an efficient fashion. In [26], a novel confusion algorithm was presented by using the paired interpermuting planes in which an exchange and random access strategy was devised to supercede the classical confusion operations. In a yet another study conducted by Choi *et al.* [32], a new ARX model-based image encryption scheme was proposed that used addition, rotation and XOR operations as its principal modi operandi for confusion and diffusion in place of S-Box and permutation as in SP networks. Moreover, a novel 3D bit matrix permutation was proposed in [15] in which the Chen system [7] was employed to come up with a visiting mechanism of randomness at the bit level of the given plain image. Through the fusion of Chen system and a 3D Cat map [34] in permutation stage, they developed a new mapping rule which connects random positions in the 3D matrix as compared to the classical sequential treatment of the input image. Although the seasoned confusion-diffusion architecture provided a lot of image ciphers but many were cracked due to some lacunas in their design. For example, these ciphers [8], [9], [15], [23], [26]–[28], [33] based upon the confusion-diffusion architecture were cryptanalysed by [35]–[42]. Traditionally in the confusion-diffusion architecture, a color image is broken into its three color components and then the pixels of each component are separately confused and diffused. Later on, these are combined together. This approach has a loophole in the sense that the potential hacker can take advantage of the fact that the three components were confused and diffused independently. This is too much advantageous for him to reach to the secret key by sharply demarcating the cryptanalytic steps. The situation can be improved if all the three components are joined together before the processes of confusion and diffusion are launched. In this way, the confusion and diffusion process will be implemented on the single gray scale image. Resultantly, the pixels of each component will move freely in the other two components thus inter-blending them in a greater degree which will in turn enhance the security of the proposed images cryptosystem. In this study, after combining the three channels of an RGB image into a single entity, the 15-puzzle problem of artificial intelligence has been used to scramble the pixels at block level, each block consisting of 16 pixels of the original plain image.

DNA technology has been successfully employed in a lot of image encryption schemes due to its marvelous features of the massive parallelism, ultra-low energy consumption and extraordinary information density [11], [43]. In the last couple of years, many new image cryptosystems were developed [17], [44]–[49] through the amalgamation of the chaotic maps and DNA computing. In the encryption schemes [45], [46], the Logistic map along with the DNA sequence operations was used. But the problem with the Logistic map is that the random numbers generated through it do not fulfill the required standard of chaoticity [13]. In the DNA based encryption schemes, DNA encoding and DNA computing are employed. DNA encoding is concerned with the complementary rules of bases while DNA computing encompasses the DNA XOR operation, DNA addition and DNA subtraction. Some image ciphers contain some defects. For instance, Zhang *et al.* [45] developed an image cipher using the DNA encoding and chaotic map which was later on proved insecure by [50] against a chosen plaintext attack. Apart from that the secret key can be obtained by using the four plain images. Besides, some image encryption schemes [11], [12], [48], [51] have employed the fixed rules for both the encoding and decoding operations for the plain image and the mask image. In particular, the scheme in [12] used the third rule for the encoding operation and the fourth rule for the decoding operation. This fixation of the rules can be easily tapped by the potential hacker. These rules should be not only make fixed but they must change according to the chosen image for encryption. So corrective measures need to be adopted to thwart the known-plaintext and chosen-plaintext attacks.

Under the light of afore-mentioned discussion, a novel scheme using Intertwining logistic map (ILM), chaotic tent map, 15-puzzle artificial intelligence problem, DNA computing and SHA 256 hash function is proposed with the following salient features:

- 15-puzzle artificial intelligence problem is being used in the scrambling of the pixels. Although this problem has no direct relevance to the image encryption but this

TABLE 1. DNA encoding rules.

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

TABLE 2. XOR operation on DNA nucleotides.

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

serves as an added layer of complexity to enhance the security of the proposed image cipher.

- The three channels red, green and blue of the RGB image are separated and are concatenated into a single gray scale image. This act exercises a three fold positive impact on the proposed scheme. Firstly, the pixel data gets confused and diffused in a greater degree because all the three channels of red, green and blue have been inter blended into a single entity which enhances the randomization process. Secondly, the confusion and diffusion processes on a single image is more secured as compared to those on individual components since combined image will be more difficult for potential cryptanlyst to deal with. Thirdly, its time complexity will drop because the said act will obliterate to repeat the steps of confusion and diffusion on each red, green and blue components independently. Further, just two rounds of confusion and only one round of diffusion have been employed.
- A higher dimensional chaotic map has been used to generate the random data. Further, to add the plaintext sensitivity, SHA 256 hash function of the plain image has been used. Every new image will have its unique chaotic data which guarantees the high security standard. Even a change of single pixel of the plain image will produce radically different chaotic data.
- With some minor modification, the proposed scheme can be easily applied on the gray scale images. Further, it can also be applied on multiple images (both RGB and gray scale images). For instance, the proposed scheme concatenates the three gray scale images of the input color image after breaking it into its constituent channels and works on a single larger gray scale image. For gray scale images, it will directly start its working on the given input image because there is no need for decomposition and concatenation. For multiple images scenario, what the scheme again requires is the set of gray scale images which will be concatenated directly in case of multiple gray scale images or indirectly(in case of mutliple color images) after first decomposing color images into gray scale ones and then concatenating them.
- Fuller potetial of DNA encoding rules has been unleashed, i.e., all the 8 rules of DNA encoding are dynamically applied and every rule depends upon the random numbers which in turn depend on the plain image being used. This act, of course, helps in defeating the chosen plaintext/ciphertext attack and increases the security of the cipher. Further, only a single XOR operation has been conducted to make the cipher a time efficient without compromising the good quality of the validation metrics.

Given the above features, the proposed image cipher is highly secure and efficient. The simulation results and exhaustive security analyses done in the Sections 4 and 5 validate our thesis.

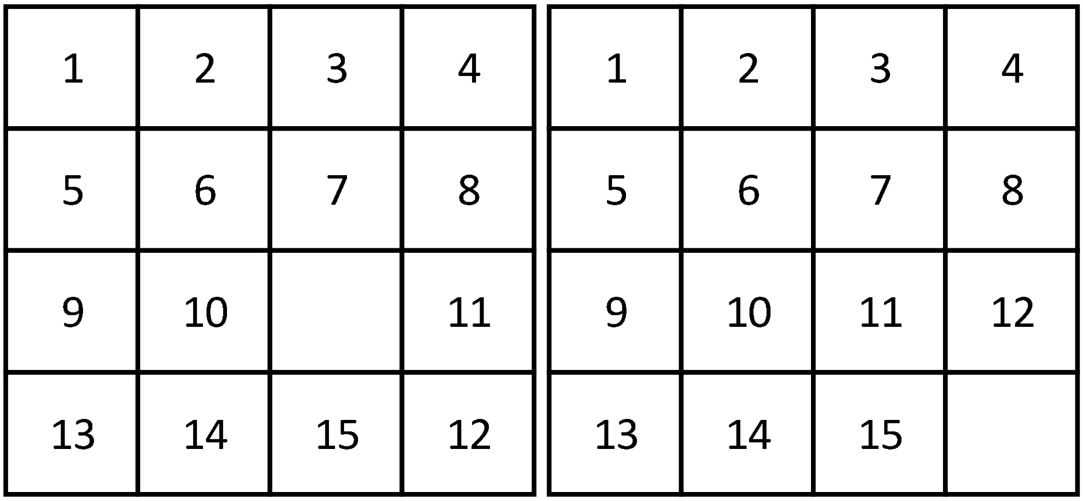
The paper has been fashioned as follows. Section 2 discusses the fundamental theories of DNA computing, chaotic maps and the 15-puzzle problem. Section 3 discusses the generation of chaotic data and the proposed RGB image encryption scheme. Sections 4 and 5 are for the simulation and security analyses. Lastly, Section 6 concludes the paper.

II. BASIC THEORIES

In this section, a brief overview will be given about the basic theories of DNA computing, chaotic systems and 15-Puzzle problem which are being used in the proposed scheme.

A. DNA COMPUTING

Each DNA comprises of four bases, i.e., A(Adenine), C(Cytosine), G(Guanine) and T(Thymine). These bases complement each other. For example, if '10' is assigned to A then '01' will be assigned to T. In the same fashion, if '00' is assigned to C then '11' will be assigned to G. In total 4! = 24 kinds of encoding, only 8 of them fulfill the Watson-Crick complementary rules which have been shown in the Table 1. In DNA computing, research studies [52] report the XOR, addition and subtraction binary operations over the DNA bases. The current study only uses the XOR operation which has been shown in the Table 2. Two conversion functions of *DECIMAL_TO_DNA* and *DNA_TO_DECIMAL* have been defined. As the name implies, *DECIMAL_TO_DNA* converts an 8-bit pixel value to its corresponding DNA sequence of length four and *DNA_TO_DECIMAL* converts the DNA sequence of length four to some decimal number. For example *DECIMAL_TO_DNA*(182, 1) = *GTCG*, *DECIMAL_TO_DNA*(182, 4) = *AGTA* and *DECIMAL_TO_DNA*(182, 7) = *GACG*. Further, *DNA_TO*



1	2	3	4
5	6	7	8
9	10		11
13	14	15	12

(a)

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

(b)

FIGURE 1. **(a) The initial state; (b); The goal state.**

$\_DECIMAL(CGTA, 1) = 108$, $DNA\_TO\_DECIMAL(CGTA, 4) = 54$ and $DNA\_TO\_DECIMAL(CGTA, 7) = 99$. Moreover, for DNA XOR operation, $\oplus(CATG, TCGA) = GCCG$.

B. CHAOTIC MAPS

Properties of theory of chaos like extreme sensitivity to the initial conditions, mixing, ergodicity and randomness potentially bore immense promise for its applications in cryptography. So a lot of image encryptions algorithms have been successfully developed using chaos with abundant complexity and security.

1) INTERTWINING LOGISTIC MAP

Chaotic maps lie at the heart of cryptography. These maps give us non-linear dynamical behavior which is very essential for encryption. A plethora of maps exist with varied characteristics. The name of one of these maps is a logistic map whose mathematical equation is

$$p_{n+1} = \mu p_n(1 - p_n) \tag{1}$$

where $0 < p_n < 1$ and $0 < \mu \leq 4$. Here μ is the system parameter and p_0 is the initial seed value which has to be given for the random numbers generation. The stream of random numbers produced by logistic map enjoys the properties of good auto-correlation and cross-correlation. But at the same time, this map is also plagued with some demerits like blank windows, uneven distribution of sequences, stable windows and a weak key [53]. Given the accompanying weaknesses of the logistic map, a new mixed map named as Intertwining Logistic Map has been developed which would address the reported weaknesses of the logistic map and would have the larger key space and is defined as follows:

$$\begin{cases} p_{n+1} = [\mu \times k_1 \times q_n \times (1 - p_n) + r_n] \bmod 1 \\ q_{n+1} = [\mu \times k_2 \times q_n + r_n \times 1/1 + p_{n+1}^2] \bmod 1 \\ r_{n+1} = [\mu \times (p_{n+1} + q_{n+1} + k_3) \times \sin r_n] \bmod 1 \end{cases} \tag{2}$$

where $0 < \mu \leq 3.999$, $|k_1| > 33.5$, $|k_2| > 37.9$, $|k_3| > 35.7$. This map gives much chaotic behavior as compared to the logistic map. Further, it contains no blank windows and has much even distribution [53].

2) CHAOTIC TENT MAP

Usually, the maps having the capability of preserving areas, like Baker map, logistic map and Lorenz map are deployed for the purpose of permutation of the pixels in the image. We have used the chaotic tent map [54] for this purpose. The discretized tent map can be defined as:

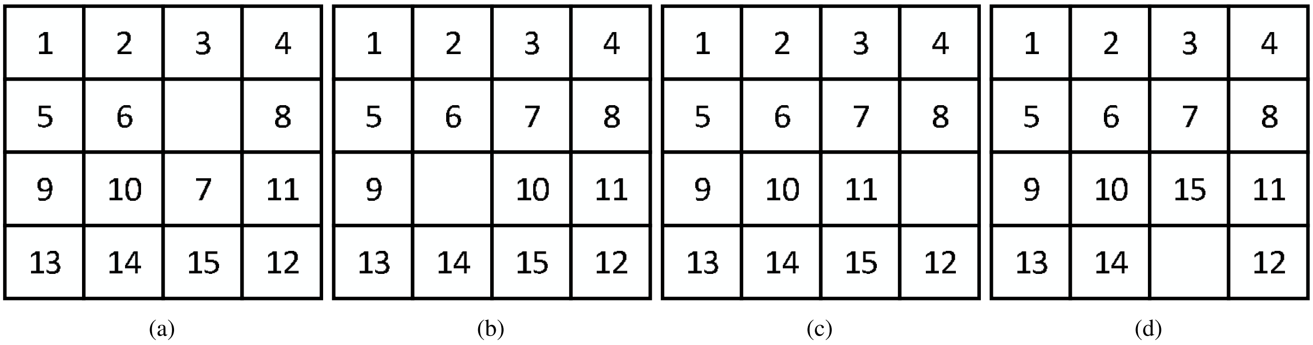
$$f(a, \rho, x) = \begin{cases} \lceil \frac{\rho}{a} x \rceil, & \text{if } 0 \leq x \leq a; \\ \lfloor \frac{\rho(\rho - x)}{\rho - a} x \rfloor + 1, & \text{if } a < x \leq \rho, \end{cases} \tag{3}$$

where $a \in (0, \rho)$ is an integer. This map will be used for shuffling the pixels in the proposed image cipher.

C. 15-PUZZLE AND HEURISTICS

The 15-puzzle problem [55] is a problem in which 15 square tiles are initially in some random order and one tile is missing. The goal of the game is to slide the square tiles by using the empty space in such a way that it makes an order. The initial state and goal state of the puzzle is given. By moving the tiles of the initial state one has to reach to the goal state. One instance of the initial state and the goal state has been given in the Figure 1.

The titles can be moved in upper, left, right and lower directions. After moving the tiles of the initial state of Figure 1a, the states in the upper, left, right and lower directions have been depicted in the Figure 2. Tiles can't be moved diagonally. Heuristics [56] has been applied on the puzzle states to select the puzzle which has the least heuristic value. The heuristic value is the number of unmatched tiles between the initial state and the goal state while no counting will be done when the initial state has an empty tile. The puzzle with the



1	2	3	4
5	6		8
9	10	7	11
13	14	15	12

(a)

1	2	3	4
5	6	7	8
9		10	11
13	14	15	12

(b)

1	2	3	4
5	6	7	8
9	10	11	
13	14	15	12

(c)

1	2	3	4
5	6	7	8
9	10	15	11
13	14		12

(d)

FIGURE 2. The four states of puzzle after moving tiles: (a) The state after moving tile in upper direction; (b); The state after moving tile in left direction; (c) The state after moving tile in right direction; (d) The state after moving tile in down direction.

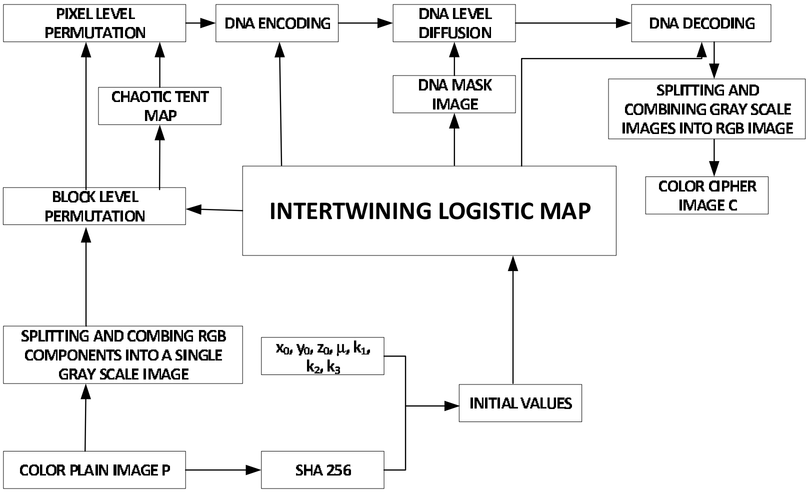


FIGURE 3. BLP-Based encryption scheme.

least heuristic value is selected for the next iteration until the heuristic with value of zero is found meaning that the goal state has been reached. The values of the heuristics for the four states depicted in the Figure 2 are 3, 3, 1 and 3 respectively. So the third state will be selected for the next iteration. In the proposed scheme, in case the empty tile is at one of the four edges and there is no space for moving in a particular direction, then the tile will move in a cyclic fashion, i.e., it will be placed in the opposite direction. This deviation from the rules makes the scheme more complex and hence more secure.

III. PROPOSED BLP-BASED IMAGE ENCRYPTION SCHEME

Let the size of the input color plain image is $M \times N \times 3$. The flowchart of the proposed BLP-Based encryption scheme has been given in the Figure 3.

In order to create the plaintext sensitivity, SHA 256 hash function has been used to generate the different hash key values for each given input image. These generated hash values in combination of the initial values and the parameters of the ILM are fed to the ILM to generate the streams of random numbers. The proposed encryption procedure consists of seven stages. In the first stage, the given color image is split into its constituent red, green and blue gray scale images each of size $M \times N$. Then these images are concatenated in the red, green and blue arrangement to get the combined single gray scale image of size $M \times 3N$. In the second stage, the single gray scale image is converted into the linear array of size $1 \times 3MN$. Then this array is divided into $\frac{3MN}{16}$ blocks of 16 pixels each, i.e., $B_1, B_2, \ldots, B_{\frac{3MN}{16}}$. Now for each block, initialize a scrambling image say D of size 4×4. We have three entities, i.e., the block of pixels, scrambling image and the 15-puzzle game. 15-puzzle game will act as a bridge to shift the pixels from the current block to the scrambling image. As the empty tile moves according to the heuristic values as described in the Section II-C in the 15-puzzle at some particular address, a pixel will be taken from the current block and it will be put in the scrambling image at the address where the empty tile is residing currently in the 15-puzzle. If empty tile reaches an address where it has already reached then no pixel will be shifted from the current block to the scrambling image. At the end, the remaining pixels of the current block will be shifted to the vacant positions of the scrambling image. In the same way, 15-puzzle game is played for each block. As the second stage ends, we get the block level scrambled image of size $M \times 3N$. In the third stage, to further increase the degree of randomness, the pixels of the block level scrambled image are further permutated by using the chaotic tent map. In the fourth stage, each pixel is DNA encoded. While DNA encoding, all the rules of conversion have been applied to make the process more and more random. The rules of conversion are used from the random numbers of the ILM which in turn uses the SHA 256 hash values of the given input image. In the fifth stage, DNA encoded pixel data is further diffused by using the DNA mask image which has been designed by the random numbers generated by the ILM. In the sixth stage, the DNA strands are converted back into the decimal to get the $M \times 3N$ gray scale image. In the seventh stage, this single gray scale image is split into three different gray scale images each of size $M \times N$. These images are then combined to get the final single color cipher image of size $M \times N \times 3$.

A. GENERATION OF THE INITIAL VALUES OF THE CHAOTIC SYSTEM

In the proposed image cipher as described earlier, a 256 bit key has been used which has been generated by the SHA 256 hash function of the plain image. Indeed, it increases the relationship of the scheme with the plain image which in turn upgrades the security of the scheme. Just due to the difference of only one bit between the plain images, the hash values obtained are radically different. The 256 bit key K is broken into different blocks each of 8 bit size expressed as follows:

$$K = k_1, k_2, \ldots, k_{32}. \tag{4}$$

subject to: $k_i = \{k_{i,0}, k_{i,1}, \ldots, k_{i,7}\}$, here i refers to the character number and j to the bit number in $k_{i,j}$. The following steps generate the chaotic data to be used in the proposed scheme.

Step 1: The initial values and the parameters of the chaotic system are being updated as follows:

$$p_0 = p'_0 + \frac{((k_1 \oplus k_2) + (k_3 \oplus k_4))}{4096} \tag{5}$$

$$q_0 = q'_0 + \frac{((k_5 \oplus k_6) + (k_7 \oplus k_8))}{4096} \tag{6}$$

$$r_0 = r'_0 + \frac{((k_9 \oplus k_{10}) + (k_{11} \oplus k_{12}))}{4096} \tag{7}$$

$$\mu = \mu'_0 + \frac{((k_{13} \oplus k_{14}) + (k_{15} \oplus k_{16}))}{4096} \tag{8}$$

$$k_1 = k'_1 + \frac{((k_{17} \oplus k_{18}) + (k_{19} \oplus k_{20}))}{4096} \tag{9}$$

$$k_2 = k'_2 + \frac{((k_{21} \oplus k_{22}) + (k_{23} \oplus k_{24}) + (k_{25} \oplus k_{26}))}{4096} \tag{10}$$

$$k_3 = k'_3 + \frac{((k_{27} \oplus k_{28}) + (k_{29} \oplus k_{30}) + (k_{31} \oplus k_{32}))}{4096} \tag{11}$$

where $p'_0, q'_0, r'_0, \mu'_0, k'_1, k'_2, x'_3$ are the initial values of the chaotic system before adding the plaintext sensitivity and $p_0, q_0, r_0, \mu_0, k_1, k_2, x_3$ are the initial values of the chaotic system after adding plaintext sensitivity. $\oplus$ denotes the XOR operation in the binary.

Step 2: By iterating the chaotic system (2) $(n_0 + 3MN)$ times, we generate the three chaotic sequences $u = [u_1, u_2, \ldots u_{3MN}]$, $v = [v_1, v_2, \ldots v_{3MN}]$ and $w = [w_1, w_2, \ldots w_{3MN}]$. Here $n_0 \geq 500$ and it is part of secret keys. For eliminating the transient effect of the chaotic system, we discard the first n_0 values of the chaotic sequence and then obtain the sequences.

Step 3: The chaotic sequences u, v and w have been again processed by the following Equations (12) to get the four sequences $Mask$, $Select$, $Rule1$ and $Rule2$.

$$\begin{cases} Select(i) = 25 - mod(floor(u(i) \times (10^{14}), 15), \\ Mask(j) = mod(floor(u(j) \times 10^{14}), 256), \\ Rule1(j) = mod(floor(v(j) \times (10^{14}), 8) + 1, \\ Rule2(j) = mod(floor(w(j) \times (10^{14}), 8) + 1, \end{cases} \tag{12}$$

where u_i, v_i and w_i are the corresponding elements of u, v and w respectively. $mod(s, t)$ gives the remainder when s is divided by t. Further, $Select(i)$ is the ith element of $Select$ and $Mask(j)$, $Rule1(j)$ and $Rule2(j)$ are the jth elements of the $Mask$, $Rule1$ and $Rule2$, $i = 1, 2, \ldots, \frac{3MN}{16}$ and $j = 1, 2, \ldots, 3MN$.

B. PROCEDURE FOR IMAGE CRYPTOSYSTEM

The following steps describe the proposed image cipher in detail.

Step 1: (Splitting and combining RGB components into single gray scale image)

Assume the size of the color plain image is $M \times N \times 3$, break it into its constituent components of red, green and blue colors. After concatenating them, we get the single gray scale image IMG of size $M \times 3N$.

Step 2: (Conversion of pixel data into an array)

Convert the single gray scale image IMG of size $M \times 3N$ into a 1D vector $IMG_1, IMG_2, \ldots,$ IMG_{3MN}.

Step 3: (Initialization)

Initialize $Initial\_State$ and the $Goal\_State$ as a 4×4 matrices as follows:

$$\begin{array}{cc} Initial\_State & Goal\_State \\ \begin{bmatrix} 0 & 1 & 2 & 3 \\ 4 & 5 & 6 & 7 \\ 8 & 9 & 10 & 11 \\ 12 & 13 & 14 & 15 \end{bmatrix} & \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 0 \end{bmatrix} \end{array}$$

Step 4: (Block level permutation (BLP))

A Block Level Permutation is proposed to scramble the 16 pixels at a time of the 4×4 pixel data by using the 15-puzzle game. Transform the IMG into the $\frac{3MN}{16}$ blocks of 16 pixels each, i.e., $B_1, B_2, \ldots, B_{\frac{3MN}{16}}$. Repeat the following steps (4.1 to 4.7) by applying the 15-puzzle on each of the $\frac{3MN}{16}$ blocks one by one.

Step 4.1:

Initialize the scrambling image D as $D(a,b) = -1$, $a = 1,2,3,4$ and $b = 1,2,3,4$ for each block B_k, k represents the block number and $k = 1,2,3,\ldots,\frac{3MN}{16}$.

Step 4.2: (Permutation)

Reshape the $Initial\_State(4 \times 4)$ matrix to $Initial\_State$ (1×16). To shuffle the position of each number in the $Initial\_State$, we have used $Select(B_k)$ times the chaotic tent map $f(w, 15, x)$ where $w = \sum_{i=0}^{15} Initial\_State(i) \bmod (15)$ and the input variable $x = \{0,1,2,\ldots,15\}$ represents the index number of each number of $Initial\_State$. Owing to the fact that the tent map is one-to-one, so it generates a permutation $\sigma(x)$. After it, this permutation has been applied to $Initial\_State$ and get $Initial\_State'$, in the way that each number will be sent to the place of its index value in $\sigma(x)$, i.e., $Initial\_State'(\sigma(x)) = Initial\_State(x)$. Reshape $Initial\_State$ to a 4×4 matrix.

Step 4.3: (Search of number Zero)

Zero refers to the empty tile. Search number zero in the 4×4 $Initial\_State$ matrix. Assume it is found at (i,j), i.e., $Initial\_State(i,j) = 0$.

Step 4.4: (Moving tiles)

As the tiles move, the values of i and j update. $Update\_Initial\_State$ is a function which takes six parameters $Initial\_State$, $Goal\_State$, m, n, i and j and returns the new address of the empty tile (ii, jj) and the new initial state of the puzzle $New\_Initial\_State$. This function has been illustrated in the Figure 4. This function further calls the function HE (Figure 5) for heuristic evaluation among all the new candidate states Up, $Left$, $Right$, and $Down$ as the empty tile moves in the up direction, left direction, right direction and the down direction respectively and the goal state $Goal\_State$. The variables hup, $hleft$, $hright$ and $hdown$ defined in the $Update\_Initial\_State$ function contain the heuristic values as the empty tile moves in the up, left, right and the down directions respectively. Depending upon the least heuristic value, the $New\_Initial\_State$ and the new address (ii, jj) are selected out of the four candidate states Up, $Left$, $Right$ and $Down$.

Initialize $o = 1$ for each block, where o is pixel number of the current block. As described earlier in this step, as the tiles move, the values of i and j change according to the following steps. The tiles will move for 16 times for each block. If there is no pixel at the address of the empty tile then the current pixel is placed at it. In the steps (c) to (e), the values of ii, jj and $New\_Initial\_State$ have been assigned to i, j and $Initial\_State$ respectively for the next iteration.

(a) $[ii, jj, New\_Initial\_State] = Update\_Initial\_State$ $(Initial\_State, Goal\_State, m, n, i, j)$

(b) if $D(ii,jj) = -1$, then $D(ii,jj) = B_{k,o}$, $o = o+1$. where k is the block number and o is the pixel number in the current block.

(c) $i = ii$

(d) $j = jj$

(e) $Initial\_State = New\_Initial\_State$

(f) Go to Step (a).

Step 4.5: Put the remaining pixels of the current block into the vacant positions of $D(4\times4)$. Reshape $D(4\times4)$ to $D(1\times16)$ and assign $D(1:16)$ to $IMG1_k(16(k-1)+1 : 16(k-1)+16)$ where k is the current block number.

$IMG1$ is the block level scrambled image of size $1 \times 3MN$.

Step 5: (Pixel level permutation)

Initialize a one dimensional array $Key = [1 : 3MN]$. Shuffle its elements t times as described in the Step 4.2 in the encryption procedure. After shuffling let we get the Key' array of size $1 \times 3MN$. Using the Key', let we get the shuffled image $IMG2$ as follows:

$$IMG2(a) = IMG1(Key'(a)) \tag{13}$$

for $a = 1,2,\ldots,3MN$.

Step 6: (DNA encoding)

Reshape matrices $Mask$, $Rule1$ and $Rule2$ into $M \times 3N$. The two 2-dimensional arrays $IMG2$, $MASK$ are converted into DNA strands to obtain arrays $IMG3$ and $MASK1$ according to the DNA conversion rule $Rule1$ as follows.

$$\begin{cases} IMG3(a,b) = DECIMAL\_TO\_DNA(IMG2(a,b), \\ \qquad\qquad Rule1(a,b)), \\ MASK1(a,b) = DECIMAL\_TO\_DNA(MASK(a,b), \\ \qquad\qquad Rule1(a,b)), \end{cases} \tag{14}$$

for $a = 1,2,\ldots,M$ and $b = 1,2,\ldots.,3N$. Each of the two 2-dimensional arrays $IMG3$ and $MASK1$ consists of $3MN$ DNA sequences of length 4.

Step 7: (DNA level diffusion) The DNA level diffusion of $IMG3$ is done with the DNA mask image $MASK1$ as follows.

$$IMG4(a,b) = IMG3(a,b) \oplus MASK1(a,b) \tag{15}$$

for $a = 1,2,\ldots,M$ and $b = 1,2,\ldots.,3N$. The 2-dimensional array $IMG4$ is the output image after diffusion and consists of DNA sequences of length 4. $\oplus$ represents the XOR operation.

Step 8: (Decimal conversion) Finally the 2-dimensional DNA array $IMG4$ is converted into the decimal array $IMG5$ by using the $Rule2$ as follows.

$$IMG5(a,b) = DNA\_TO\_DECIMAL(IMG4(a,b), \\ Rule2(a,b)) \tag{16}$$

for $a = 1,2,\ldots,M$ and $b = 1,2,\ldots.,3N$. $IMG5$ is the $M \times 3N$ output gray scale image consisting of the red, green and blue components placed after one another.

Step 9: (Splitting and combining gray scale image into the RGB image)

Lastly split the $IMG5$ image into its constituent components and combine them to make the final cipher output color image $IMG6$ of size $M \times N \times 3$.

Since the proposed image cipher is symmetric in character so the decryption algorithm has been done in the reverse order of the encryption one.

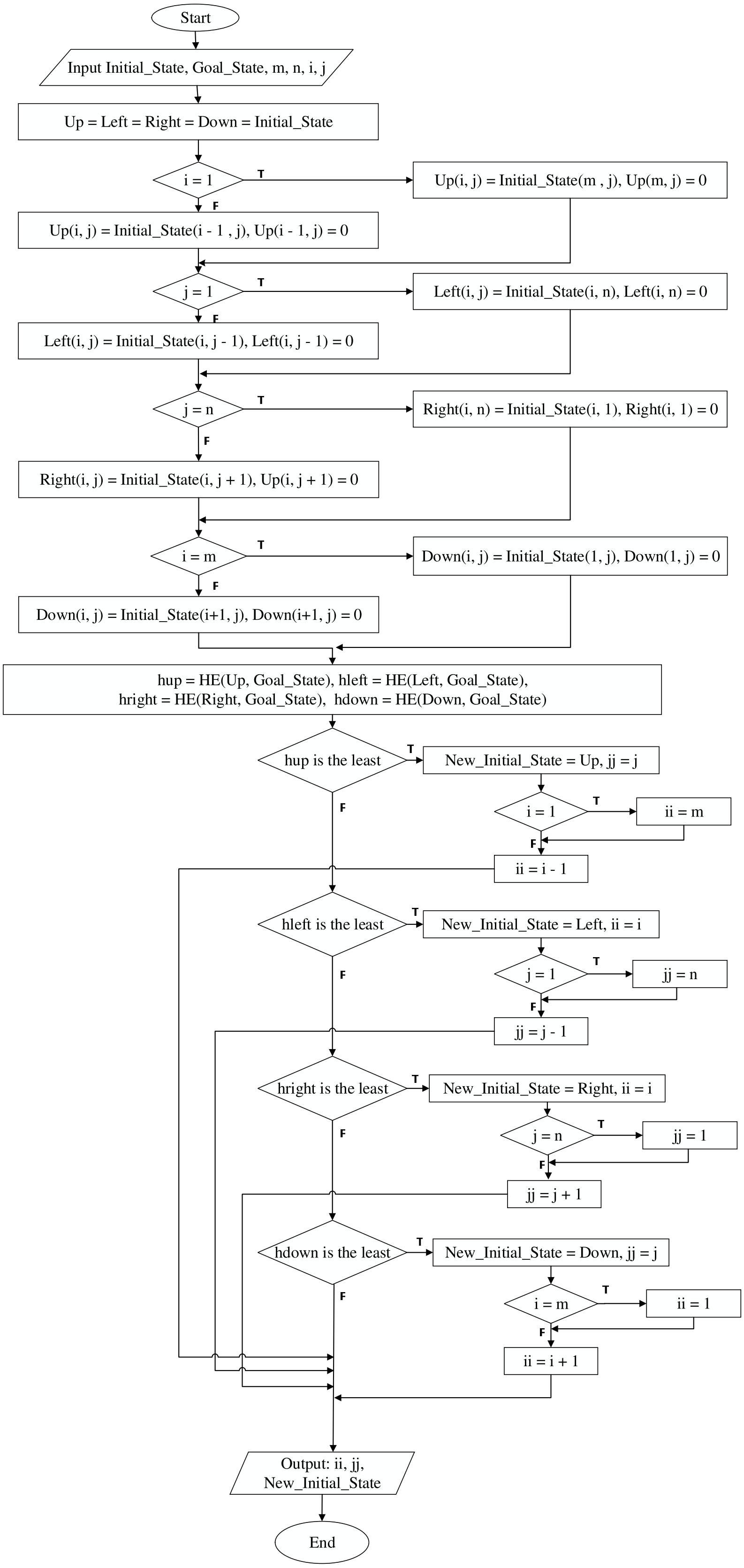

Start
Input Initial_State, Goal_State, m, n, i, j
Up = Left = Right = Down = Initial_State
i = 1
T
F
Up(i, j) = Initial_State(m , j), Up(m, j) = 0
Up(i, j) = Initial_State(i - 1 , j), Up(i - 1, j) = 0
j = 1
Left(i, j) = Initial_State(i, n), Left(i, n) = 0
Left(i, j) = Initial_State(i, j - 1), Left(i, j - 1) = 0
j = n
Right(i, n) = Initial_State(i, 1), Right(i, 1) = 0
Right(i, j) = Initial_State(i, j + 1), Up(i, j + 1) = 0
i = m
Down(i, j) = Initial_State(1, j), Down(1, j) = 0
Down(i, j) = Initial_State(i+1, j), Down(i+1, j) = 0
hup = HE(Up, Goal_State), hleft = HE(Left, Goal_State),
hright = HE(Right, Goal_State), hdown = HE(Down, Goal_State)
hup is the least
New_Initial_State = Up, jj = j
i = 1
ii = m
ii = i - 1
hleft is the least
New_Initial_State = Left, ii = i
j = 1
jj = n
jj = j - 1
hright is the least
New_Initial_State = Right, ii = i
j = n
jj = 1
jj = j + 1
hdown is the least
New_Initial_State = Down, jj = j
i = m
ii = 1
ii = i + 1
Output: ii, jj,
New_Initial_State
End

FIGURE 4. Update_Initial_State Function.

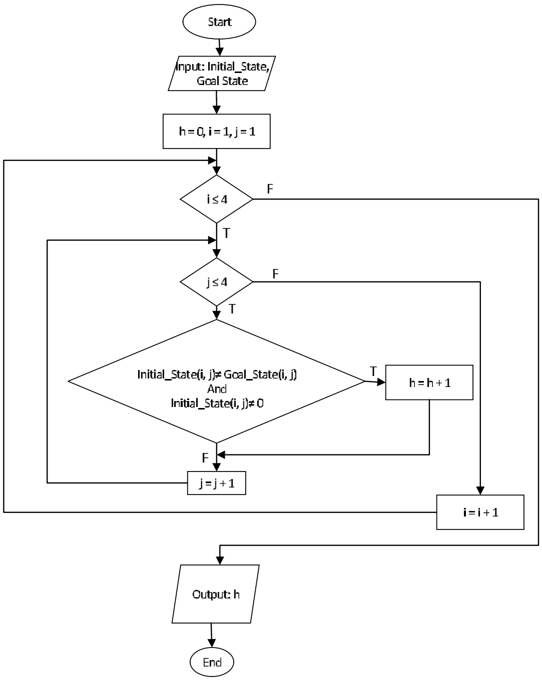


FIGURE 5. Heuristic Evaluation (HE) Function.

IV. SIMULATION RESULTS

It's an implicit assumption behind every images encryption scheme that it should have the sufficient capability to thwart attacks aimed at it like brute force attack, differential attack, statistical attack, entropy attack and chosen plaintext/ciphertext attack etc.

For demonstrating the do-ability of our scheme, we have taken eight color images of Lena, Baboon, Girl, Tree, House, Beans, F16, Couple with the sizes of 256 × 256. They have been taken from the USC-SIPI Image Database which can be reached at http://sipi.usc.edu/database/. MATLAB 2016 version with 64-bit double-precision according to the IEEE [57] stnadard 754 has been used for simulation.

For simulating the proposed scheme, the initial values and the systems parameters taken for the ILM are: $p_0 = 0.36$, $q_0 = 0.25$, $r_0 = 0.78$, $\mu = 1.5$, $k_1 = 35.5$, $k_2 = 38.2$, $k_3 = 36.0$. Further, the value of t taken for chaotic tent map is 15. Figures 6a to 6g show the original Lena color image, the combined gray scale image of Lena consisting of the red, green and blue channels put one after the other, the combined gray scale image after block level scrambling, the combined gray scale image after pixel level permutation, the combined gray scale image after DNA level diffusion followed by conversion into decimal, the single encrypted color image and the decrypted Lena color image respectively.

Further Figure 7 depicts the input images and their ciphered and decrypted images have been shown in the Figures 8 and 9 respectively. One can see that after the encryption process, the plain images have been completely morphed into noisy ones leaving no clue to some potential person with malicious intention. This fact clearly indicates the good encryption effect of the proposed cipher.

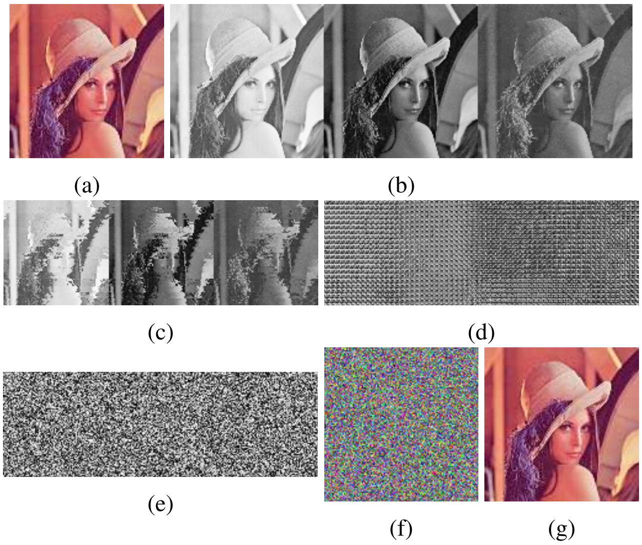


FIGURE 6. (a) Lena plain image; (b) The combined gray scale image of Lena consisting of red, green and blue channels; (c) after block level scrambling in (b); (d) after pixel level permutation in (c); (e) after DNA level diffusion followed by convertion it decimal in (d); (f) after combining into a single color cipher image in (e); (g) after decrypting into original Lena image in (f).

TABLE 3. A comparison of key space with some other schemes.

Algorithm	Key space
Ours	10^{108}
Ref. [50]	10^{88}
Ref. [59]	5.46×10^{80}
Ref. [60]	10^{60}
Ref. [61]	10^{48}

V. SECURITY ANALYSES

In this section, different security analyses will be presented.

A. KEY SPACE

Maintaining a reasonable key space is one of the cardinal foci while developing any images cryptosystem. A reasonably large key space has the capability to foil any brute-force attack potentially aimed at it. The key in the proposed cipher comprises of the values p_0, q_0, r_0 and system parameters μ, k_1, k_2 and k_3 of ILM. Further, the key t for the chaotic tent map is 15. The computer precision for the ILM has been taken to be 10^{-15}. Further the key space for the chaotic tent map is 10^3. So the total key space comes out to be $10^{108} \approx 2^{365}$. So it has the sufficient capability to defy any brute-force attack. Apart from that, a comparison of key space has been made in the Table 3 with some other existing schemes.

B. KEY SENSITIVITY

Eexterme sensitivity to a key is one of the necessary salient features of any image cipher. In other words, if a tiny change is made in one of the keys, there should be a phenomenal change in the output. How much sensitive a key is in a cipher image, is checked through two types. In the first case, a tiny change is made in one of the keys while encrypting

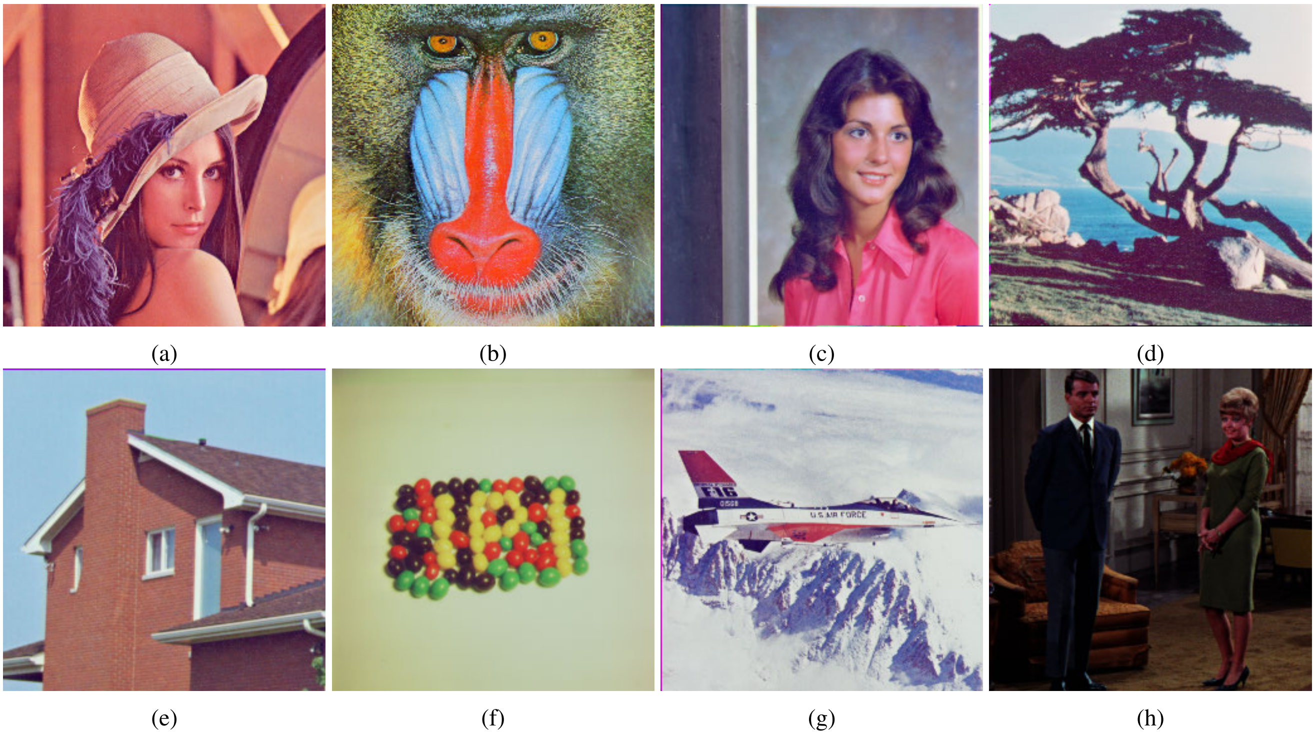


FIGURE 7. The original plain images: (a) Lena; (b); Baboon; (c) Girl; (d) Tree; (e) House; (f) Beans; (g) F16; (h) Couple.

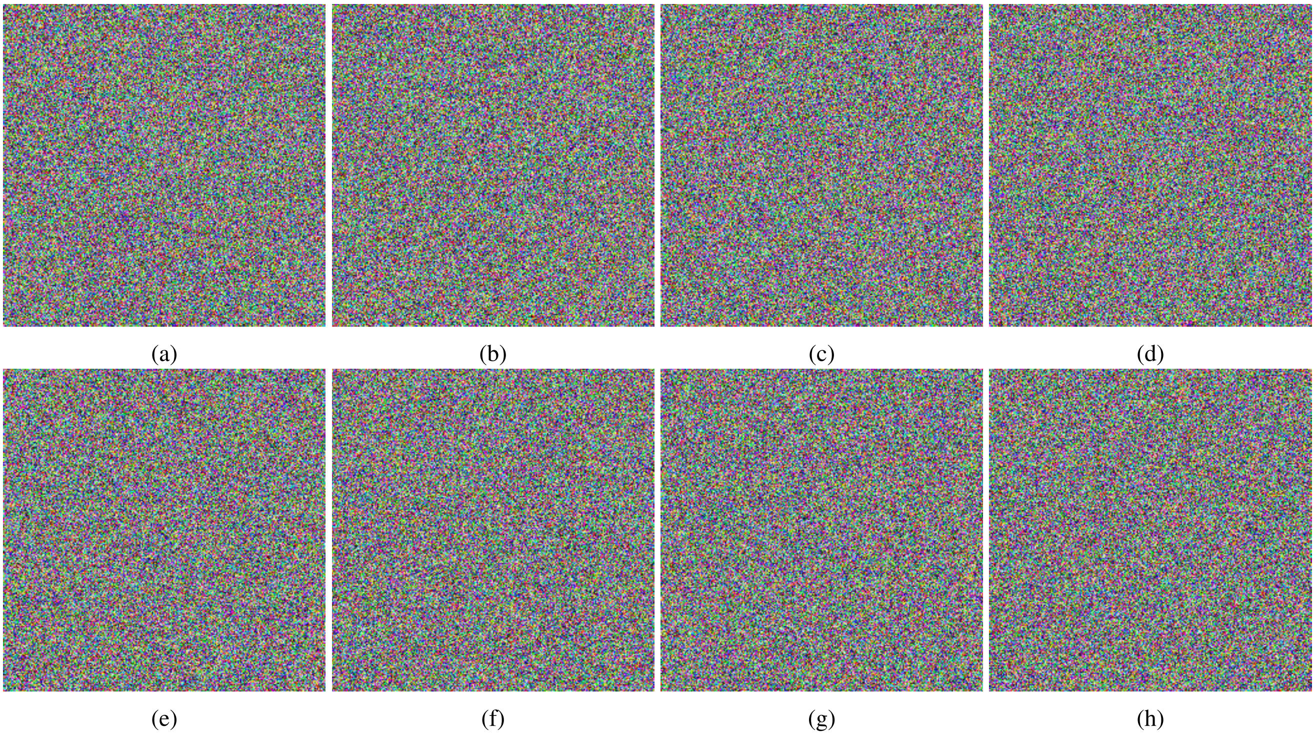


FIGURE 8. The cipher images: (a) Lena; (b) Baboon; (c) Girl; (d) Tree; (e) House; (f) Beans; (g) F16; (h) Couple.

some image. It should produce a radically different cipher image. In the second case, a tiny change is made in one of the keys while decryption. The cipher image should not be decrypted unless the correct set of keys is employed.

In the first key sensitivity tests, same plain image is encrypted through the two slightly different keys.

Let the initial key set be $S_0 = \{p_0, q_0, r_0, \mu, k_1, k_2, k_3\}$. The key S_0 was used for encrypting the Lena image in

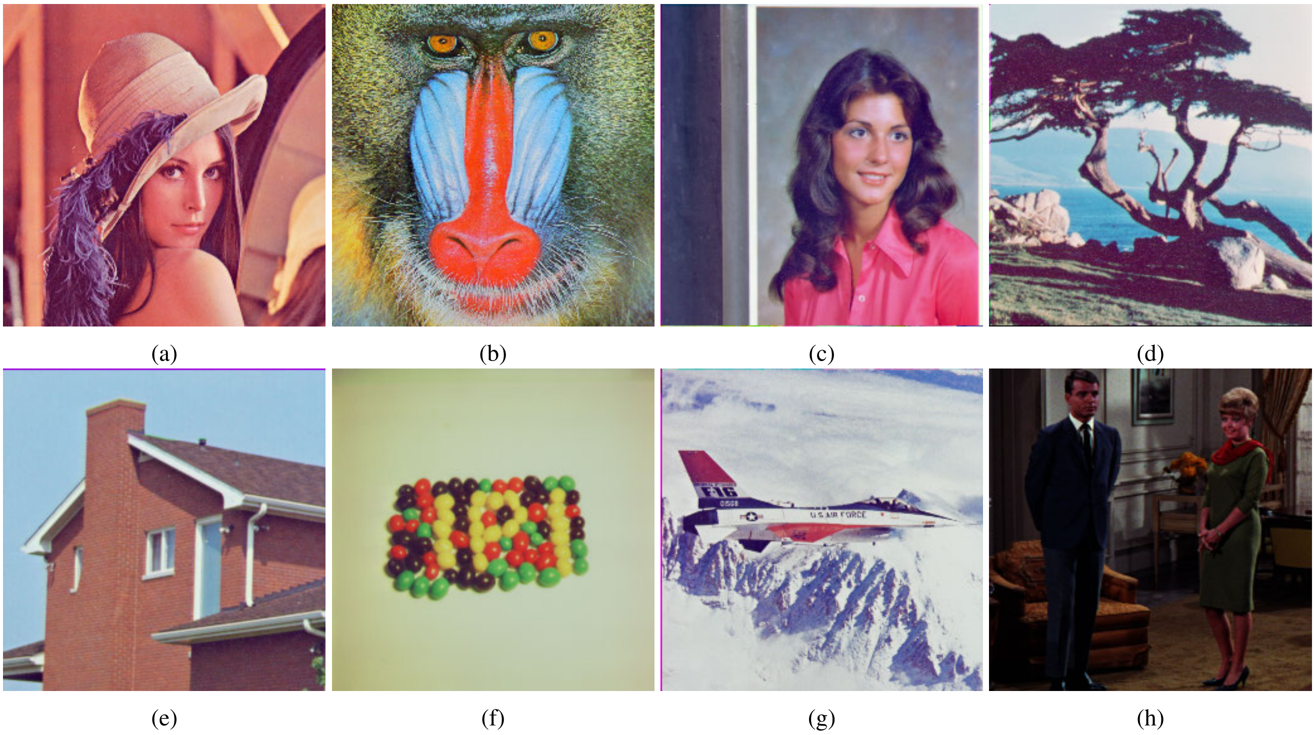


FIGURE 9. The decrypted images:(a) Lena; (b) Baboon; (c) Girl; (d) Tree; (e) House; (f) Beans; (g) F16; (h) Couple.

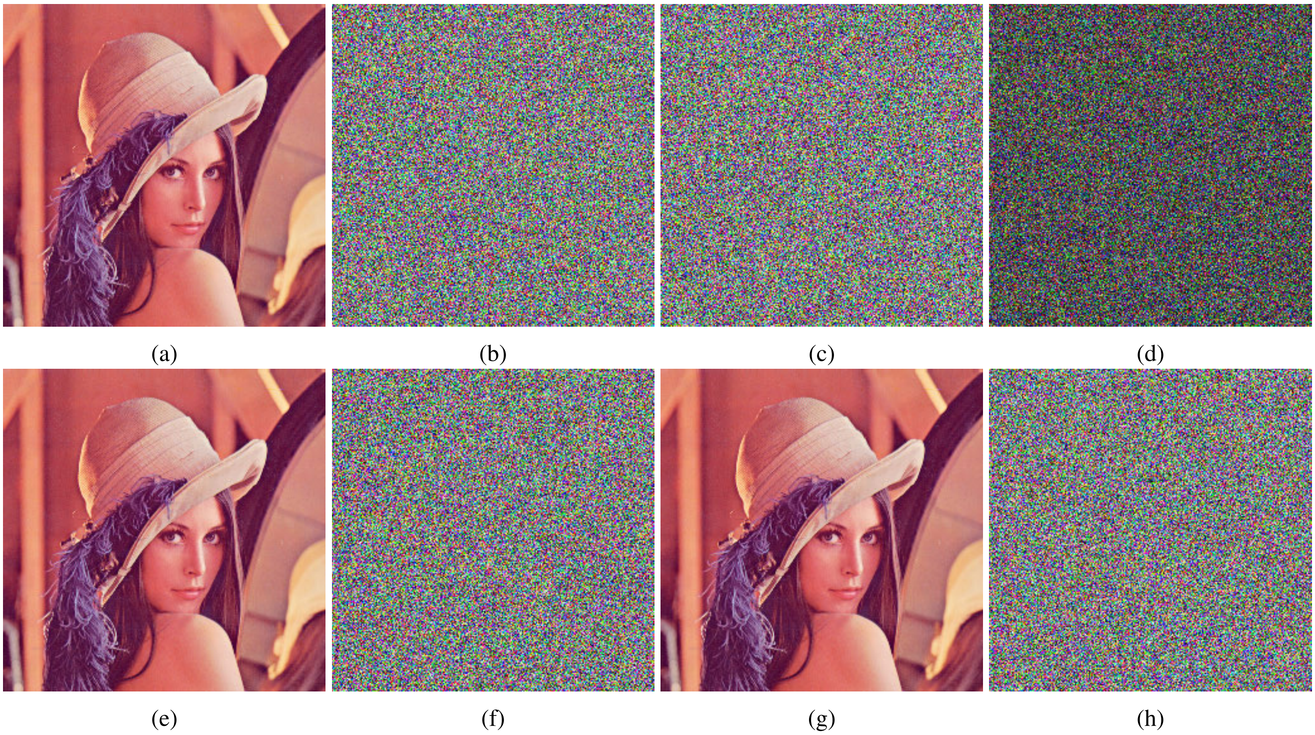


FIGURE 10. Key sensitivity test on Lena image: (a) The plain image; (b) The encrypted image with S_0; (c) The encrypted image with S_1; (d) The differential image between (b) and (c); (e) The decrypted image from (b) with the correct key set S_0; (f) The decrypted image from (b) with the wrong key set S_1; (g) The decrypted image from (c) with the correct key set S_1; (h) The decrypted image from (c) with the wrong key set S_0.

Figure 10a and the cipher image in Figure 10b has been obtained. After it, a slight change say $\Delta = 10^{-14}$ has been made to p_0, i.e., $p_0{}' = p_0 + \Delta$, and other keys have been kept unchanged. This act produced an other set of keys say S_1. Now S_1 has been used for encrypting the same Lena image in Figure 10a. The corresponding cipher image has been

TABLE 4. Rates of difference between two ciphered images obtained by slightly different keys.

Keys	Difference rates (%)							
	Lena	Baboon	Girl	Tree	House	Beans	F16	Couple
$S_1(p_0{}' = p_0 + \Delta)$	99.6312	99.5860	99.6053	99.5900	99.6267	99.6053	99.6119	99.6277
$S_2(q_0{}' = q_0 + \Delta)$	99.6251	99.6236	99.6043	99.5672	99.5890	99.6165	99.6134	99.5977
$S_3(r_0{}' = r_0 + \Delta)$	99.6165	99.5900	99.6023	99.6282	99.6017	99.6048	99.6226	99.6099
$S_4(\mu' = \mu + \Delta)$	99.6145	99.6089	99.6048	99.5967	99.6012	99.6333	99.6236	99.6160
$S_5(k_1{}' = k_1 + \Delta)$	99.6028	99.6124	99.6267	99.6104	99.6002	99.5992	99.6068	99.6068
$S_6(k_2{}' = k_2 + \Delta)$	99.6017	99.6084	99.6017	99.6506	99.6226	99.6287	99.6384	99.6007
$S_7(k_3{}' = k_3 + \Delta)$	99.6002	99.6175	99.6068	99.6257	99.6211	99.6211	99.5834	99.6063
$S_8(p_0{}' = p_0 - \Delta)$	99.6170	99.5860	99.5962	99.6348	99.6175	99.6109	99.6104	99.6211
$S_9(q_0{}' = q_0 - \Delta)$	99.6104	99.6002	99.6073	99.6053	99.5916	99.6206	99.6048	99.5972
$S_{10}(r_0{}' = r_0 - \Delta)$	99.6058	99.6257	99.6226	99.6129	99.6124	99.6262	99.6084	99.6119
$S_{11}(\mu' = \mu - \Delta)$	99.6033	99.6155	99.5926	99.5926	99.6150	99.6134	99.6007	99.5972
$S_{12}(k_1{}' = k_1 - \Delta)$	99.6140	99.6048	99.6307	99.6089	99.6063	99.5956	99.6023	99.6028
$S_{13}(k_2{}' = k_2 - \Delta)$	99.6094	99.6109	99.5972	99.6150	99.6129	99.6109	99.6201	99.6099
$S_{14}(k_3{}' = k_3 - \Delta)$	99.6277	99.5850	99.5946	99.6129	99.6028	99.6063	99.5997	99.6012
Average	**99.61**	**99.61**	**99.61**	**99.61**	**99.61**	**99.61**	**99.61**	**99.61**
Average of all	**99.61**	-	-	-	-	-	-	-

shown in Figure 10c. Figure 10d depicts the differential image between these two cipher images which has been obtained by the pixel-to-pixel difference. Besides, even a very minute change of (10^{-14}) is present in the secret keys, the pixels of the encrypted image in Figure 10b has 99.6312% differences from the one in Figure 10c. To observe the key sensitivity in a still greater extent, we have computed the rates of difference between the two cipher-images generated by S_0 and $S_a(a = 1, 2, \ldots 14)$.

Through the two slightly different key sets S_0 and S_a, the corresponding results are listed in Table 4. The Table 4 shows that the minimum difference rate between any two cipher images is 99.5672% which is better than [2], [60], [61]. The average key sensitivity has come out to be 99.61 which is equal to the [62] and better than [60], [61]. Therefore, it can be said that the proposed cipher has clear is relatively better than the others.

To gauge how much sensitive a key is as far as the second case is concerned, S_0 and S_1 have been used to retrieve the cipher images in Figures 10b and 10c respectively. The resultant retrieved images have been shown in the Figures 10e-10h. It is clear from the figures that only the correct key has the potential to recover the cipher images. Even a very minute change of the keys will render a radically distinct result and cannot give the valid plain image. So we are justified in saying that the proposed image cipher enjoys a high key sensitivity.

C. STATISTICAL ANALYSIS

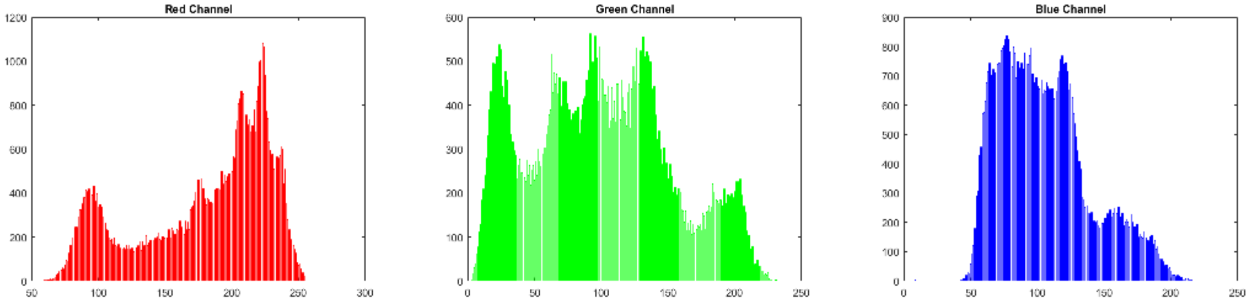
Defying statistical attacks on the image ciphers is one of the fundamental requirements while designing any image cryptosystem. Normally two kinds of statistical attacks are analyzed, i.e., histogram attack and the correlation attack.

1) HISTOGRAM

An image's histogram is a pictorial representation of the different pixel intensity values. Since each image is inherently a reasonable combination of pixels, so naturally the histogram made through this image has a characteristic curved bar. This curved bar is impregnated with much information of the image which can be tapped by the antagonist. So the histogram made through the usage of encrypted image should be uniform enough so that it may not leak any meaningful information. Figures 11 and 12 show the histograms of the plain as well as cipher images of Lena respectively. One can easily witness the fluctuating and uniform bars of the plain image and cipher images respectively. The uniform bars of the encrypted images make the statistical analysis attack very difficult.

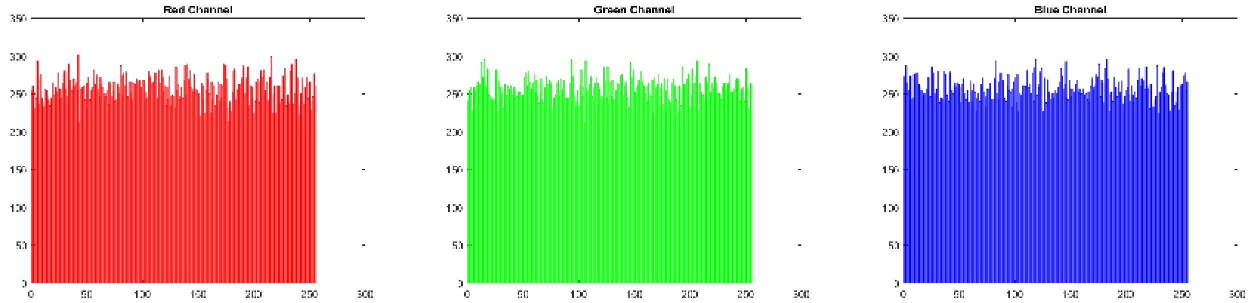
Besides, to quantify the uniformity of a histogram, a statistical metric called variance is normally calculated. The relatively smaller values of variance give the higher uniformity of the histograms of the cipher images [8], [63]. Table 5 shows this very important metric of the histograms of the encrypted Lena, Baboon, Girl, Tree, House, Beans, F16 and Couple images. The results shown in the first row have been calculated by the S_0, whereas the others in the remaining rows have been obtained by modifying the secret key which is S_a $(a = 1, 2, \ldots 14)$ defined in the Section V-B.

The average of the different results of the variance of the cipher-images is 255.9408, whereas for plain images, it is about 48,000. So the proposed image cipher is efficient. Besides, the impact of modifying the secret keys on the uniformity of the histograms of the cipher images have been computed. For this purpose, the percentages of the variance differences between the two cipher images have been calculated. The first cipher image was obtained using



(a) (b) (c)

FIGURE 11. Lena's plain image histogram in the (a) red, (b) green, (c) blue components.



(a) (b) (c)

FIGURE 12. Lena's cipher image histogram in the (a) red, (b) green, (c) blue components.

TABLE 5. The variance results using different keys.

Images	Lena	Baboon	Girl	Tree	House	Beans	F16	Couple	**Average**	**Average of all**
S_0	256.2083	263.0677	260.1771	251.0990	248.8333	254.3021	257.1016	260.5781	**256.4209**	**255.9408**
S_1	261.7005	236.1198	261.7292	250.7031	266.6432	234.3802	232.7917	253.9401	**249.7510**	-
S_2	239.2526	260.5677	254.5547	250.4740	260.5938	285.2786	264.7344	238.5677	**256.7529**	-
S_3	281.4219	248.4844	240.3203	250.2865	246.8281	239.7578	243.2396	242.6953	**249.1292**	-
S_4	246.4115	278.8073	269.8568	277.2630	246.3490	255.9141	277.2083	243.5208	**261.9164**	-
S_5	260.8698	251.8542	249.5156	253.2865	260.3151	237.6901	242.2708	262.7083	**252.3138**	-
S_6	232.2813	249.3281	270.3906	262.3620	243.2266	240.9740	264.7526	243.2292	**250.8181**	-
S_7	270.7656	248.6146	243.2188	280.3411	253.7526	255.3229	254.7839	285.5130	**261.5391**	-
S_8	248.7031	274.2734	257.5807	257.3516	252.2813	256.3516	251.2240	250.3958	**256.0202**	-
S_9	231.6797	278.2057	236.0625	261.1745	258.3906	265.3464	272.4583	254.5234	**257.2301**	-
S_{10}	260.1901	262.1953	261.0911	228.3125	257.9688	251.9974	263.8802	247.4063	**254.1302**	-
S_{11}	258.8307	254.2656	263.1823	256.0104	262.2500	249.5911	268.9766	277.7682	**261.3594**	-
S_{12}	252.0156	257.8516	272.9141	249.3333	255.1953	275.9766	259.1510	258.7682	**260.1507**	-
S_{13}	253.3359	266.0078	251.2057	257.7396	246.4505	252.5104	250.7188	259.8281	**254.7246**	-
S_{14}	259.8359	249.3906	236.1458	244.7604	271.2318	283.3698	274.1250	235.9844	**256.8555**	-

the key S_0 and the second one was obtained through the key set $S_a(a = 1, 2, ...14)$. The results have been listed in the Table 6. It can be seen that the average is 4.24% which is better than [60]–[62]. Further, the maximum of the averages of the percentages for all the fourteen keys of our scheme is 6.86% which is less than 10.07% [62], 11.23% [60] and 11.10% [61]. Moreover, the maximal variance fluctuation amplitude of the proposed scheme is 12.18% which is smaller than [62]. Hence, we can say that the proposed image cipher bears better security features.

2) CORRELATION ANALYSIS

Images are normally visually meaningful. So a high correlation exists among the adjacent/consecutive pixels of these images. Two consecutive horizontal, two consecutive vertical and two consecutive diagonal pixles are termed as adjacent pixels. One of the main jobs of any images cryptosystem is to smash this correlation among these adjacent pixels and to convert them into a noise-like image with almost nil correlation. There is no correlation among the consecutive pixels in case the given image is encrypted ideally.

TABLE 6. Variance difference percentages.

Test Images	Lena	Baboon	Peppers	Tree	House	Beans	F16	Couple	**Average**	**Average for all**
$S_1(\%)$	2.14	10.24	0.60	0.16	7.16	7.83	9.46	2.55	**5.02**	**4.24**
$S_2(\%)$	6.62	0.95	2.16	0.25	4.73	12.18	2.97	8.45	**4.79**	-
$S_3(\%)$	9.84	5.54	7.63	0.32	0.81	5.72	5.39	6.86	**5.26**	-
$S_4(\%)$	3.82	5.98	3.72	10.42	1.00	0.63	7.82	6.55	**4.99**	-
$S_5(\%)$	1.82	4.26	4.10	0.87	4.61	6.53	5.77	0.82	**3.60**	-
$S_6(\%)$	9.34	5.22	3.93	4.49	2.25	5.24	2.98	6.66	**5.01**	-
$S_7(\%)$	5.68	5.49	6.52	11.65	1.98	0.40	0.90	9.57	**5.27**	-
$S_8(\%)$	2.93	4.26	1.00	2.49	1.39	0.81	2.29	3.91	**2.39**	-
$S_9(\%)$	9.57	5.75	9.27	4.01	3.84	4.34	5.97	2.32	**5.63**	-
$S_{10}(\%)$	1.55	0.33	0.35	9.07	3.67	0.91	2.64	5.05	**2.95**	-
$S_{11}(\%)$	1.02	3.35	1.16	1.96	5.39	1.85	4.62	6.60	**3.24**	-
$S_{12}(\%)$	1.64	1.98	4.90	0.70	2.56	8.52	0.80	0.69	**2.72**	-
$S_{13}(\%)$	1.12	1.12	3.45	2.64	0.96	0.70	2.48	0.29	**1.60**	-
$S_{14}(\%)$	1.42	5.20	9.24	2.52	9.00	11.43	6.62	9.44	**6.86**	-

TABLE 7. Correlation coefficient for Lena plain image and its encrypted version.

Image	Channel	Correlation direction		
		Horizontal	Vertical	Diagonal
Original Lena image	Red	0.9451	0.9693	0.9271
	Green	0.9303	0.9590	0.9091
	Blue	0.8893	0.9123	0.8647
Encrypted Lena image	Red	0.0091	0.0124	0.0010
	Green	-0.0015	0.0023	0.0241
	Blue	-0.0068	0.0006	-0.0116

TABLE 8. A comparison of CC between the proposed cipher and the others.

Image	Encryption algorithm	Correlation direction		
		Horizontal	Vertical	Diagonal
Original Lena image		0.9216	0.9469	0.9003
Encrypted Lena image	our algorithm	0.0003	0.0051	0.0028
	Ref. [45]	0.0042	0.0033	0.0024
	Ref. [47]	0.0072	0.0058	0.0031
	Ref. [48]	0.0214	0.0465	-0.0090
CT_Abdomen	Ref. [60]	0.0032	-0.0344	-0.0123
MR_Cervical_vertebra	Ref. [60]	-0.0087	0.0078	0.0045
X_Lungs	Ref. [60]	-0.0140	-0.0125	-0.0168
Brain	Ref. [1]	0.0019	0.0263	0.0196

For evaluating the correlation between the two consecutive pixels, 3,000 pairs of consecutive pixels were randomly chosen in the three directions from the plain image and that of cipher one. The mathematical formula given in equation 17, as shown at the bottom of this page, is used to calculate the correlation coefficient (CC) [5]: where p and q are the pixel intensity values of two consecutive pixels and N is for the total number of pixels. Figure 13 depicts the correlation distribution of consecutive pixels in the horizontal, vertical and diagonal directions of the original and encrypted Lena image.

The correlation coefficients between two consecutive pixels for original and encrypted image of Lena have been given in Table 7. Table 7 shows that the correlation coefficients between the plain image is very close to 1 while they are close to 0 in the case of cipher image. Both the Table 7 and Figure 13 show that after the encryption process the relation between the input and output images has been reduced dramatically. It other words, this indicates an across the board smashing of the correlation of the consecutive pixels in the plain image. Table 8 has compared the correlation with some of the algorithms given by using different encryption schemes [1], [44], [46], [47], [59].

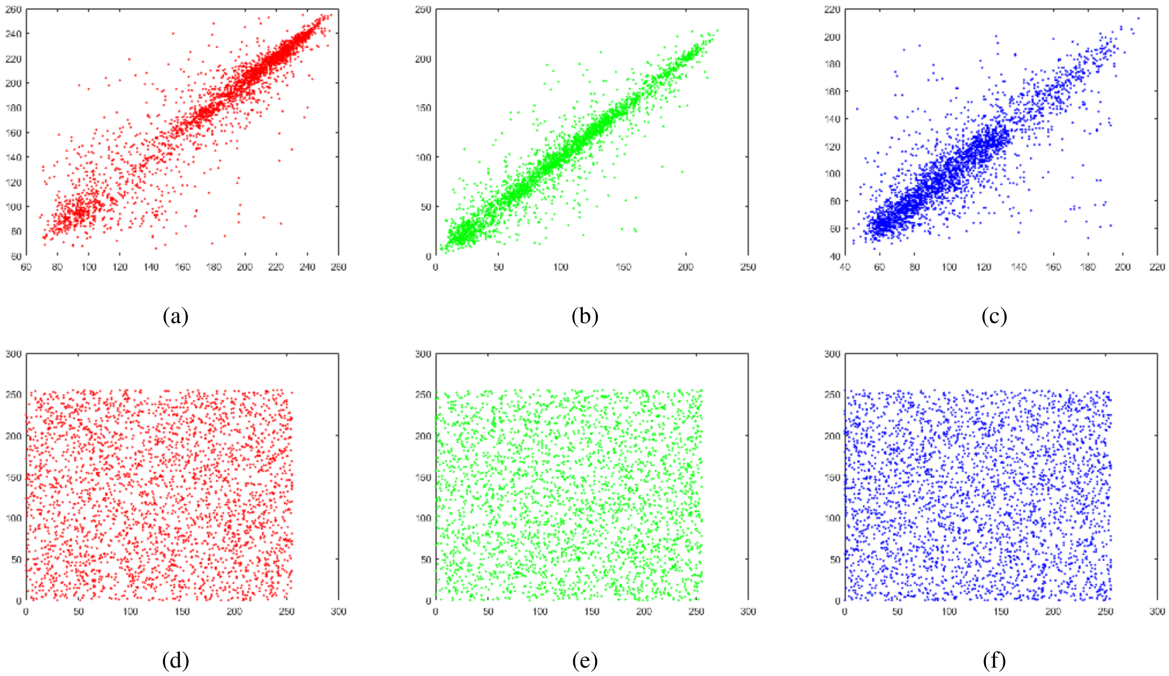
D. INFORMATION ENTROPY ANALYSIS

Randomness and unpredictability of some signal is frequently encountered in research. To quantify this concept, it was Shannon [64] who gave a mathematical formula in 1949:

$$R(m) = \sum_{a=0}^{2^n-1} q(m_a) log \frac{1}{q(m_a)} \tag{18}$$

where $R(m)$ is the entropy of some random source m. $q(m_a)$ is the probability of the symbol m_a. For an ideal random image with 256 gray values, the number 8 comes out for its

$$CC = \frac{N \sum_{l=1}^{N} (p_l \times q_l) - \sum_{l=1}^{N} p_l \times \sum_{l=1}^{N} q_l}{\sqrt{\left(N \sum_{l=1}^{N} p_l^2 - \left(\sum_{l=1}^{N} p_l\right)^2\right)\left(N \sum_{l=1}^{N} q_l^2 - \left(\sum_{l=1}^{N} q_l\right)^2\right)}} \tag{17}$$



(a) (b) (c)

(d) (e) (f)

FIGURE 13. Correlation distribution for Lena image: (a) Horizontally consecutive pixels in the red component; (b) Vertically consecutive pixels in the green component; (c) Diagonally consecutive pixels in the blue component; (d) Horizontally consecutive pixels in the red component; (e) Vertically consecutive pixels in the green component; (f) Diagonally consecutive pixels in the blue component.

TABLE 9. Information entropy results and comparison.

Image ciphers	Images	Original			Encrypted		
		Red	Green	Blue	Red	Green	Blue
Our algorihtm	Lena	7.2507	7.5931	6.9659	7.9967	7.9974	7.9974
	Baboon	7.6942	7.4637	7.7443	7.9971	7.9969	7.9973
	Girl	7.2549	7.2704	6.7825	7.9972	7.9972	7.9970
	Tree	7.2104	7.4136	6.9207	7.9972	7.9974	7.9970
	House	6.4311	6.5389	6.2320	7.9973	7.9972	7.9973
	Beans	5.2626	5.6947	6.5464	7.9975	7.9969	7.9972
	F16	6.7106	6.7962	6.2001	7.9971	7.9970	7.9974
	Couple	6.2499	5.9642	5.9309	7.9971	7.9970	7.9972
	Average	**6.7581**	**6.8419**	**6.6654**	**7.9972**	**7.9971**	**7.9972**
Ref. [17]	Lena				7.9973	7.9969	7.9971
Ref. [50]	Lena				7.9892	7.9896	7.9896
Ref. [66]	Lena				7.9895	7.9894	7.9894
Ref. [67]	Lena				7.9943	7.9943	7.9942
Ref. [60]	CT_Abdomen					7.9993	
Ref. [60]	MR_Cervical_vertebra					7.9993	
Ref. [60]	X_Lungs					7.9994	
Ref. [50]	Brain				7.9901	7.9902	7.9899
Ref. [1]	Brain					7.9971	

optimum value. So, it is always desirable to come up with such a value of an entropy which is very close to 8. Table 9 depicts the entropies of different images. The average values of the entropies of the eight chosen images is very nearly equal to the maximum value of 8. Hence our proposed cipher is very much impervious to the entropy attack. Table 9 further compares the average measure of entropy with some of the existing algorithms.

E. PLAINTEXT SENSITIVITY (DIFFERENTIAL ATTACK)

One of the ways to crack an encryption algorithm is a differential attack. This name stems from the fact that a very minute change is made in the plain image. After this, both the plain images before the minor change and after the minor change are encrypted. In this way, some potential relationing can be spotted between these two encrypted images which can lead to discover the secret key. Two measures have been

TABLE 10. Average values of differential attack metrics(NPCR, UACI).

Images	NPCR(%)			UACI(%)		
	Red	Green	Blue	Red	Green	Blue
Lena	99.6155	99.6155	99.5926	33.5034	33.4627	33.4094
Baboon	99.5712	99.5895	99.5941	33.5335	33.5389	33.3340
Girl	99.5834	99.6063	99.5895	33.5355	33.3697	33.3466
Tree	99.6048	99.6078	99.6155	33.3627	33.4240	33.4453
House	99.6185	99.6521	99.6216	33.5685	33.6147	33.4273
Beans	99.6124	99.5911	99.6094	33.4211	33.4706	33.4630
F16	99.5911	99.6155	99.6078	33.3735	33.4650	33.4423
Couple	99.5682	99.5834	99.6307	33.5159	33.5996	33.5232
Average	**99.5956**	**99.6077**	**99.6077**	**33.4768**	**33.4932**	**33.4239**
Average for all images		**99.6037**			**33.4646**	

put forward by the researchers to deal this situation. One is the ($NPCR$) and the other is ($UACI$). The first one stands for number of pixels change rate, whereas the second one refers to the unified average changing intensity. These measures test the aftermath repercussions on the ensuing cipher image the moment one changes just a single intensity value in a pixel of the input image. Their mathematical formulae are:

$$NPCR = \frac{\sum_{a,b} D(a,b)}{P \times Q} \times 100\% \qquad (19)$$

where P and Q represent the dimensions of the image. $D(a, b)$ can be defined by:

$$D(a,b) = \begin{cases} 1, & \text{if } \; C(a,b) \neq C'(a,b); \\ 0, & \text{if } \; C(a,b) = C'(a,b). \end{cases} \qquad (20)$$

$$UACI = \frac{1}{P \times Q}\left[\sum_{a,b} \frac{|C(a,b) - C'(a,b)|}{255}\right] \times 100\% \qquad (21)$$

C and C' are respectively the ciphered images. These images have been obtained as described earlier with no change in the pixel of the given image and with a one pixel change in the given image.

Table 10 shows the $NPCR$ and $UACI$ values of the chosen eight images. The averages of $NPCR$ and $UACI$ of red, green and blue components of the eight images are 99.6037% and 33.4646% respectively which clearly prove that the proposed image cipher is strong enough to defeat the differential attacks of $NPCR$ and $UACI$. Further Table 11 compares our values of $NPCR$ and $UACI$ with some other algorithms. One can see the superiority of our algorithm over the some of the recent algorithms.

F. PEAK SIGNAL-TO-NOISE RATIO ANALYSIS

The basic theme of the images encryption technology is to maximize the discrepancy between the input image and output image. Here we use a similarity measure called Peak-Signal-to-Noise Ratio (PSNR) to measure the difference between the input image and the output image. Mathematically it is defined as

TABLE 11. Comparison of the average differential attack metrics(NPCR, UACI) by different encryption algorithms.

Algorithm	Image	Average NPCR(%)	Average UACI(%)
Our Algorithm	Lena	99.6037	33.4646
Ref. [59]	Lena	99.59	33.41
Ref. [60]	Lena	99.6037	33.4463
Ref. [68]	Lena	99.5991	33.4650
Ref. [60]	CT_Abdomen	99.6109	33.4311
Ref. [60]	MR_Cervical_vertebra	99.6056	33.4897
Ref. [60]	X_Lunags	99.6151	33.4188
Ref. [50]	Brain	99.6090	33.4727
Ref. [1]	Brain	99.62	33.46

$$\begin{cases} PSNR = 20log_{10}(255/\sqrt{MSE})dB \\ MSE = \dfrac{1}{P \times Q}\displaystyle\sum_{a=1}^{P}\sum_{j=b}^{Q}(P_0(a,b) - P_1(a,b))^2 \end{cases} \qquad (22)$$

where P and Q are the dimensions of the test image. $P_0(a, b)$ and $P_1(a, b)$ are the pixel values of the original and encrypted images respectively. Besides, MSE stands for the mean squared error and it is the error or departure between the original image and its encrypted version. The MSE and $PSNR$ are reciprocally interrelated. In other words, for the larger values of the MSE, we will get relatively smaller values of the $PSNR$. Further, the larger values of $PSNR$ are better for the encryption security.

The $PSNR$ values by different techniques have been given in Table 12. As can be seen from the table, the $PSNR$ values are always ∞. This phenomenon refers to the reality that the output image is exactly identical to its input counterpart. And this phenomenon happens due to $MSE = 0$. It signals to the fact that the proposed cipher does not let lose any information of the given input image. Further, the $PSNR$ value of Lena image obtained by our algorithm is the smallest in comparison of other algorithms [68]–[70]. So our algorithm has a better encryption effect.

TABLE 12. The PSNR results: 'O-C' is original and ciphered images; 'O-D' is the original and decrypted images.

		Lena	Baboon	Girl	Tree	House	Beans	F16	Couple
Ours	PSNR (O-D)	∞	∞	∞	∞	∞	∞	∞	∞
	PSNR (O-C)	7.7840	8.7616	8.4196	8.7127	9.7593	8.4511	8.1576	6.6861
Ref. [69]	PSNR (O-D)	96.2956							
	PSNR (O-C)	9.2322							
Ref. [70]	PSNR (O-C)	8.1717							
Ref. [71]	PSNR (O-C)	9.0486							

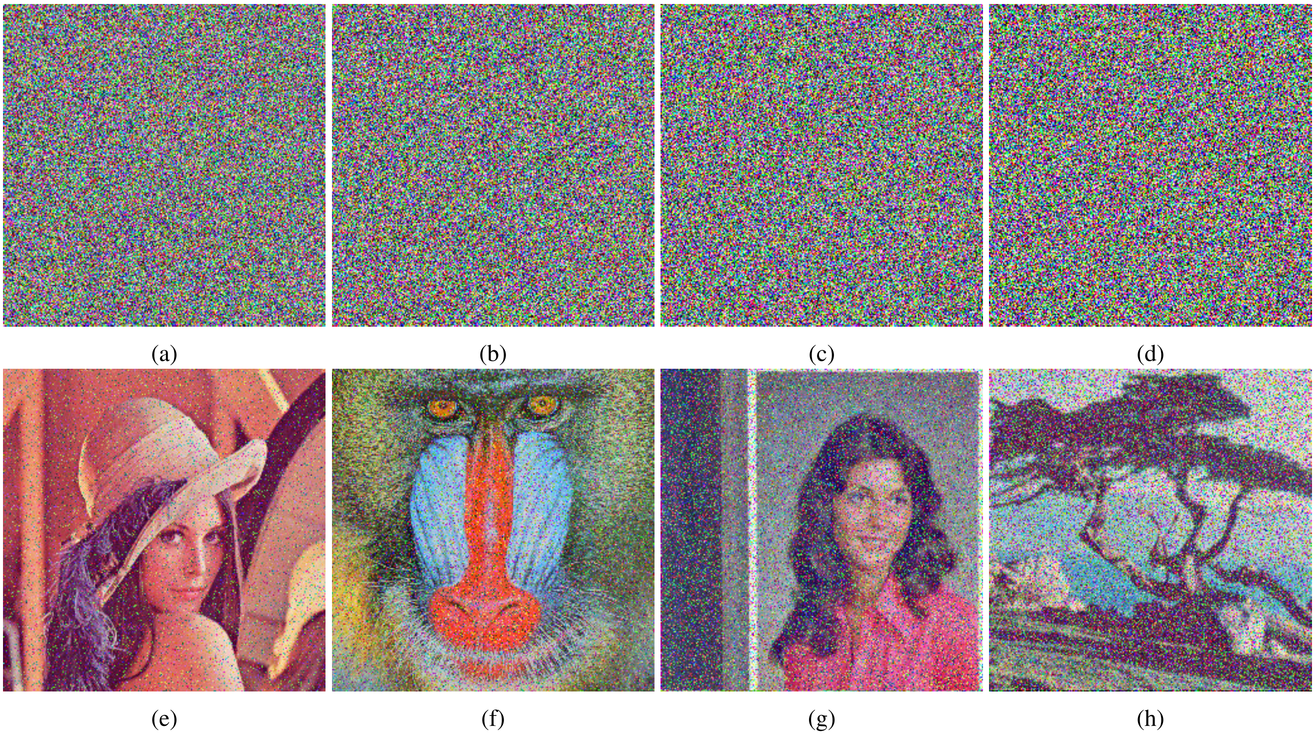


FIGURE 14. Pepper & Salt noise attack:(a) Ciphered Lena image with noise density 0.1; (b) Ciphered Baboon image with noise density 0.2; (c) Ciphered Girl image with noise density 0.3; (d) Ciphered Tree image with noise density 0.4; (e) Decrypted image from (a); (f) Decrypted image from (b); (g) Decrypted image from (c); (h) Decrypted image from (d).

G. MEAN ABSOLUTE ERROR (MAE)

One of the objectives of any image cipher is to maximize the difference between the input and output images. Mean absolute error (MAE) is used for this purpose. Mathematically, this can be written as:

$$MAE_{R,G,B} = \frac{1}{R \times S} \sum_{a=1}^{R} \sum_{b=1}^{S} |C_{R,G,B}(a,b) - P_{R,G,B}(a,b)| \tag{23}$$

where P and C are the plain image and cipher image respectively, R and S being the width and height of the image. The larger value of MAE is better. Table 13 gives the results of MAE produced by our algorithm and compares with the result of an other scheme [71].

TABLE 13. The *MAE* results.

Image	MAE		
	Red	Green	Blue
Lena	84.2148	78.0518	70.4546
Baboon	76.4598	72.4911	79.4111
Girl	79.2246	78.1964	69.9497
Tree	76.8297	87.1676	80.5626
House	69.5224	76.6176	80.0007
Beans	79.0065	82.7562	71.8427
F16	81.4952	84.6688	83.3977
Couple	97.0392	104.6060	106.2350
Average for each component	**80.4740**	**83.0694**	**80.2318**
Average for all images		**81.2584**	
Ref. [71]		79.354	

H. NOISE AND DATA CROP ATTACKS

In real life situation, the transmission of images may suffer from contamination due to some kind of noise. Occasionally, some part of the image during transmission is also lost. A good encryption scheme is assumed to defeat both the noise and crop attacks. Figures 14a to 14d show the encrypted images contaminated by Pepper & Salt noise with different noise densities, i.e., 0.1, 0.2, 0.3 and 0.4 and

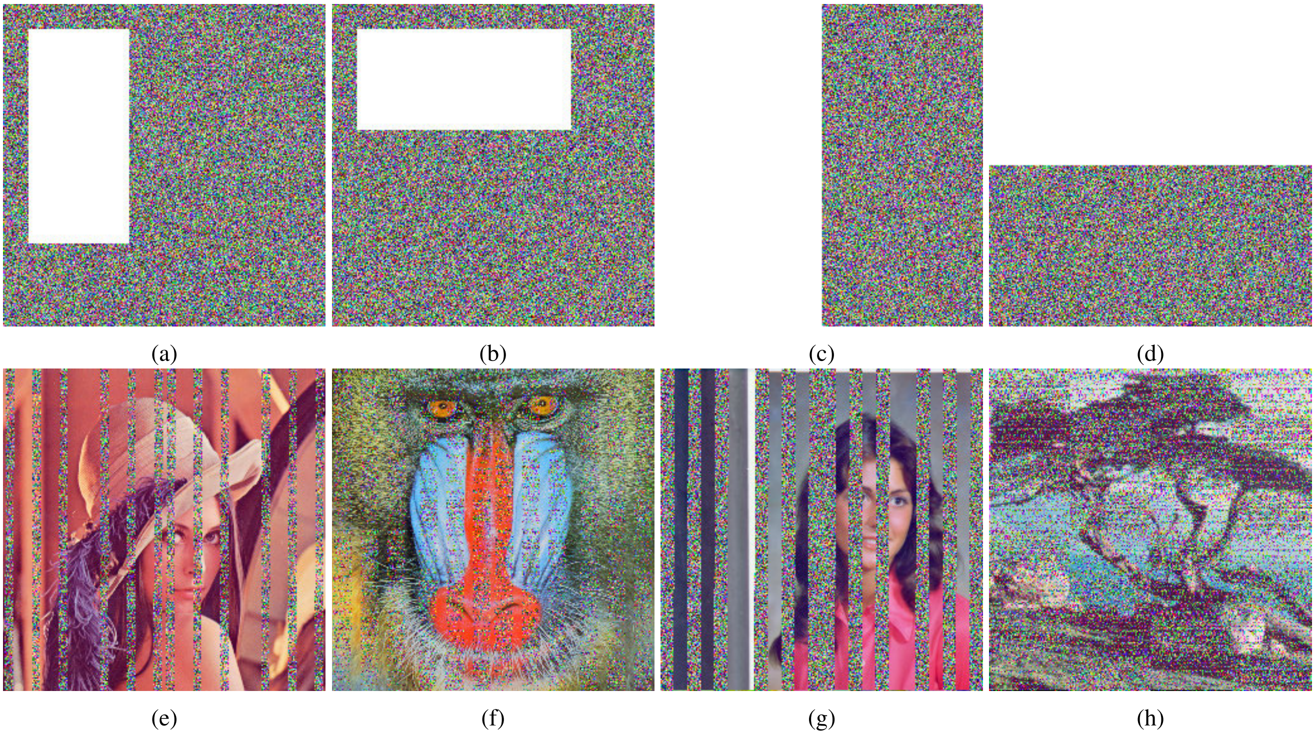


FIGURE 15. Data loss attack: (a)Encrypted Lena image with 170 × 80 data loss; (b) Encrypted Baboon image with 80 × 170 data loss; (c) Encrypted Girl image with 128 × 256 data loss; (d)Encrypted Tree image with 256 × 128 data loss; (e) Decrypted Lena image from (a); (f) Decrypted Baboon image from (b); (g) Decrypted Girl image from (c); (h) Decrypted Tree image from (d).

TABLE 14. Speed performance of the proposed algorithm and some other ciphers.

Algorithm	Speed (MBps)
Proposed	0.053
Ref. [1]	0.020
Ref. [73]	0.051
Ref. [74]	0.051

Figures 14e to 14h depict the corresponding decrypted images using our proposed scheme. One can easily recognize the original visual information.

Further Figures 15a to15d plot the encrypted Lena image, encrypted Baboon image, encrypted Girl image and encrypted Tree image with data loss attacks of 170 × 80, 80 × 170, 128 × 256 and 256 × 128 respectively. After it, the decryption algorithm has been applied to these cropped cipher images. Figures 15e to 15h show the corresponding decrypted images. Clearly, the decrypted images from our scheme still has most of the visual information. So, we are justified in saying that the proposed scheme has an excellent capability to avert any data loss threat during the transmission of images.

I. SPEED PERFORMANCE

Apart from the security considerations, an encryption algorithm should be efficient as well vis-à-vis its running time. Such an algorithm has more prospects in the real world situation for its application. The proposed algorithm has been coded and compiled under Intel(R) Core(TM) i5-4210U CPU @ 1.70 GHz 2.40 GHz, 8 GB memory, Windows 10, MATLAB R2016a. The Table 14 compares the time taken between our algorithm and the other schemes.

VI. CONCLUSION

By using the chaotic systems, DNA technology and 15-Puzzle artificial intelligence problem, a new and a secured image encryption technique has been proposed in this study. This technique falls in the domain of classical permutation-diffusion architecture. It differs from it in the sense that before starting the processes of confusion and diffusion upon the three gray scale images individually, they are concatenated into a single gray scale image. It gives three benefits; Firstly, the pixels do not remain restricted within the confines of a specific color component while they are scrambled, rather they move freely in all the three components of the color image thus creating the scrambling effect in a greater degree. Secondly, the merging of three color components into a single entity creates more difficulty for the potential cryptanalyst to find the secret key. Thirdly, it improves the time efficiency because confusion and diffusion will be done on a single image. Further, in DNA diffusion process, only single round of DNA XOR operation provided very good results. Apart from that, the SHA 256 hash function of input RGB image has been used to temper the initial values of the chaotic system being used. All the four streams of random numbers generated through the chaotic system depend heavily upon the input color image being used. This promises high plaintext sensitivity. Lastly the comprehensive security analyses further strengthen our thesis regarding the superiority of validation metrics over those of many of the current image ciphers. So we highly recommend our image cipher for the real world application.

REFERENCES

[1] L. Xu, Z. Li, J. Li, and W. Hua, "A novel bit-level image encryption algorithm based on chaotic maps," *Opt. Lasers Eng.*, vol. 78, pp. 17–25, Mar. 2016.

[2] Z. Parvin, H. Seyedarabi, and M. Shamsi, "A new secure and sensitive image encryption scheme based on new substitution with chaotic function," *Multimedia Tools Appl.*, vol. 75, no. 17, pp. 10631–10648, 2016.

[3] J.-W. Han, C.-S. Park, D.-H. Ryu, and E.-S. Kim, "Optical image encryption based on XOR operations," *Opt. Eng.*, vol. 38, pp. 47–54, Jan. 1999.

[4] S. Liu, Q. Mi, and B. Zhu, "Optical image encryption with multistage and multichannel fractional Fourier-domain filtering," *Opt. Lett.*, vol. 26, no. 16, pp. 1242–1244, 2001.

[5] G. Chen, Y. Mao, and C. K. Chui, "A symmetric image encryption scheme based on 3D chaotic cat maps," *Chaos, Solitons Fractals*, vol. 21, no. 3, pp. 749–761, Jul. 2004.

[6] N. K. Pareek, V. Patidar, and K. K. Sud, "Image encryption using chaotic logistic map," *Image Vis. Comput.*, vol. 24, no. 9, pp. 926–934, 2006.

[7] H. Liu and X. Wang, "Color image encryption using spatial bit-level permutation and high-dimension chaotic system," *Opt. Commun.*, vol. 284, nos. 16–17, pp. 3895–3903, 2011.

[8] Z.-L. Zhu, W. Zhang, K.-W. Wong, and H. Yu, "A chaos-based symmetric image encryption scheme using a bit-level permutation," *Inf. Sci.*, vol. 181, no. 6, pp. 1171–1186, 2011.

[9] X. Huang, "Image encryption algorithm using chaotic Chebyshev generator," *Nonlinear Dyn.*, vol. 67, no. 4, pp. 2411–2417, 2012.

[10] O. Mirzaei, M. Yaghoobi, and H. Irani, "A new image encryption method: Parallel sub-image encryption with hyper chaos," *Nonlinear Dyn.*, vol. 67, no. 1, pp. 557–566, 2012.

[11] M. Babaei, "A novel text and image encryption method based on chaos theory and DNA computing," *Natural Comput.*, vol. 12, no. 1, pp. 101–107, 2013.

[12] Q. Zhang, L. Guo, and X. Wei, "A novel image fusion encryption algorithm based on DNA sequence operation and hyper-chaotic system," *Optik*, vol. 124, no. 18, pp. 3596–3600, Sep. 2013.

[13] Y. Zhou, L. Bao, and C. P. Chen, "A new 1D chaotic system for image encryption," *Signal Process.*, vol. 97, no. 11, pp. 172–182, 2014.

[14] Y.-Q. Zhang and X.-Y. Wang, "A new image encryption algorithm based on non-adjacent coupled map lattices," *Appl. Soft. Comput.*, vol. 26, pp. 10–20, Jan. 2015.

[15] W. Zhang, H. Yu, Y.-I. Zhao, and Z.-L. Zhu, "Image encryption based on three-dimensional bit matrix permutation," *Signal Process.*, vol. 118, pp. 36–50, Jan. 2016.

[16] X. Chai, "An image encryption algorithm based on bit level Brownian motion and new chaotic systems," *Multimedia Tools Appl.*, vol. 76, no. 1, pp. 1159–1175, 2017.

[17] X. Chai, X. Fu, Z. Gan, Y. Lu, and Y. Chen, "A color image cryptosystem based on dynamic DNA encryption and chaos," *Signal Process.*, vol. 155, pp. 44–62, Feb. 2019.

[18] C. Li, "Cracking a hierarchical chaotic image encryption algorithm based on permutation," *Signal Process.*, vol. 118, pp. 203–210, Jan. 2016.

[19] C. Li, D. Lin, and J. Lü, "Cryptanalyzing an image-scrambling encryption algorithm of pixel bits," *IEEE Multimedia*, vol. 24, no. 3, pp. 64–71, Mar. 2017.

[20] Y. Liu, H. Fan, E. Y. Xie, G. Cheng, and C. Li, "Deciphering an image cipher based on mixed transformed logistic maps," *Int. J. Bifurcation Chaos*, vol. 25, no. 13, 2015, Art. no. 1550188.

[21] A. Jain and N. Rajpal, "A robust image encryption algorithm resistant to attacks using DNA and chaotic logistic maps," *Multimedia Tools Appl.*, vol. 75, no. 10, pp. 5455–5472, May 2016.

[22] Q. Zhang, L. Liu, and X. Wei, "Improved algorithm for image encryption based on DNA encoding and multi-chaotic maps," *Int. J. Electron. Commun.*, vol. 68, no. 3, pp. 186–192, Mar. 2014.

[23] R. Boriga, A. C. Dăscălescu, and I. Priescu, "A new hyperchaotic map and its application in an image encryption scheme," *Signal Process., Image Commun.*, vol. 29, no. 8, pp. 887–901, Sep. 2014.

[24] J. Fridrich, "Symmetric ciphers based on two-dimensional chaotic maps," *Int. J. Bifurcation Chaos*, vol. 8, no. 6, pp. 1259–1284, 1998.

[25] Z. Hua, Y. Zhou, C.-M. Pun, and C. L. P. Chen, "2D Sine Logistic modulation map for image encryption," *Inf. Sci.*, vol. 297, pp. 80–94, Mar. 2015.

[26] W. Zhang, H. Yu, and Z.-L. Zhu, "Color image encryption based on paired interpermuting planes," *Opt. Commun.*, vol. 338, pp. 199–208, Mar. 2015.

[27] J.-X. Chen, Z.-L. Zhu, C. Fu, L.-B. Zhang, and Y. Zhang, "An efficient image encryption scheme using lookup table-based confusion and diffusion," *Nonlinear Dyn.*, vol. 81, no. 3, pp. 1151–1166, 2015.

[28] G. Zhou, D. Zhang, Y. Liu, Y. Yuan, and Q. Liu, "A novel image encryption algorithm based on chaos and Line map," *Neurocomputing*, vol. 169, pp. 150–157, Dec. 2015.

[29] R. Guesmi, M. A. Farah, A. Kachouri, and M. Samet, "Hash key-based image encryption using crossover operator and chaos," *Multimed Tools Appl.*, vol. 75, no. 8, pp. 4753–4769, 2016.

[30] E. Yavuz, R. Yazici, M. C. Kasapba i, and E. Yamaç, "A chaos-based image encryption algorithm with simple logical functions," *Comput. Electr. Eng.*, vol. 54, pp. 471–483, Aug. 2016.

[31] Z. Hua and Y. Zhou, "Image encryption using 2D logistic-adjusted-sine map," *Inf. Sci.*, vol. 339, pp. 237–253, Apr. 2016.

[32] J. Choi, S. Seok, H. Seo, and H. Kim, "A fast ARX model-based image encryption scheme," *Multimedia Tools Appl.*, vol. 75, no. 22, pp. 14685–14706, 2016.

[33] C. Pak and L. L. Huang, "A new color image encryption using combination of the 1D chaotic map," *Signal Process.*, vol. 138, pp. 129–137, Sep. 2017.

[34] K. Wang, L. Zou, A. Song, and Z. He, "On the security of 3D Cat map based symmetric image encryption scheme," *Phys. Lett. A*, vol. 343, no. 6, pp. 432–439, 2005.

[35] X. Wang, D. Luan, and X. Bao, "Cryptanalysis of an image encryption algorithm using Chebyshev generator," *Digit. Signal Process.*, vol. 25, pp. 244–247, Feb. 2014.

[36] W. Wen, Y. Zhang, M. Su, R. Zhang, J.-X. Chen, and M. Li, "Differential attack on a hyper-chaos-based image cryptosystem with a classic bi-modular architecture," *Nonlinear Dyn.*, vol. 87, no. 1, pp. 383–390, Jan. 2017.

[37] D. Ponnain and K. Chandranbabu, "Security analysis of an image encryption algorithm based on paired interpermuting planes and a modified scheme," *Optik*, vol. 127, no. 19, pp. 8111–8123, 2016.

[38] G. Hu, D. Xiao, Y. Wang, and X. Li, "Cryptanalysis of a chaotic image cipher using Latin square-based confusion and diffusion," *Nonlinear Dyn.*, vol. 88, no. 2, pp. 1305–1316, 2017.

[39] L. Chen, B. Ma, X. Zhao, and S. Wang, "Differential cryptanalysis of a novel image encryption algorithm based on chaos and line map," *Nonlinear Dyn.*, vol. 87, no. 3, pp. 1797–1807, Feb. 2017.

[40] H. Wang, D. Xiao, X. Chen, and H. Huang, "Cryptanalysis and enhancements of image encryption using combination of the 1D chaotic map," *Signal Process.*, vol. 144, pp. 444–452, Mar. 2018.

[41] Y.-Q. Zhang and X.-Y. Wang, "Analysis and improvement of a chaos-based symmetric image encryption scheme using a bit-level permutation," *Nonlinear Dyn.*, vol. 77, no. 3, pp. 687–698, Mar. 2014.

[42] J. Wu, X. Liao, and B. Yang, "Cryptanalysis and enhancements of image encryption based on three-dimensional bit matrix permutation," *Signal Process.*, vol. 142, pp. 292–300, Jan. 2018.

[43] L. M. Adleman, "Molecular computation of solutions to combinatorial problems," *Science*, vol. 266, no. 5187, pp. 1021–1024, 1994.

[44] X. Wei, L. Guo, Q. Zhang, J. Zhang, and S. Lian, "A novel color image encryption algorithm based on DNA sequence operation and hyper-chaotic system," *J. Syst. Softw.*, vol. 85, no. 2, pp. 290–299, 2012.

[45] L. Liu, Q. Zhang, and X. Wei, "A RGB image encryption algorithm based on DNA encoding and chaos map," *Comput. Electr. Eng.*, vol. 38, no. 5, pp. 1240–1248, Sep. 2012.

[46] R. Enayatifar, A. H. Abdullah, and I. Isnin, "Chaos-based image encryption using a hybrid genetic algorithm and a DNA sequence," *Opt. Lasers Eng.*, vol. 56, pp. 83–93, May 2014.

[47] P. Zhen, G. Zhao, L. Min, and X. Jin, "Chaos-based image encryption scheme combining DNA coding and entropy," *Multimedia Tools Appl.*, vol. 75, no. 11, pp. 6303–6319, 2016.

[48] R. Guesmi, M. A. B. Farah, A. Kachouri, and M. Samet, "A novel chaos-based image encryption using DNA sequence operation and secure hash algorithm SHA-2," *Nonlinear Dyn.*, vol. 83, no. 3, pp. 1123–1136, Feb. 2016.

[49] X. Wu, K. Wang, X. Wang, H. Kan, and J. Kurths, "Color image DNA encryption using NCA map-based CML and one-time keys," *Signal Process.*, vol. 148, pp. 272–287, Jul. 2018.

[50] F. Özkaynak, A. B. Özer, and S. Yavuz, "Security analysis of an image encryption algorithm based on chaos and DNA encoding," in *Proc. 21st Signal Process. Commun. Appl. Conf. (SIU)*, Apr. 2013, pp. 1–4.

[51] Q. Zhang, L. Guo, and X. P. Wei, "Image encryption using DNA addition combining with chaotic maps," *Math. Comput. Model.*, vol. 52, nos. 11–12, pp. 2028–2035, Dec. 2010.

[52] O. D. King and P. Gaborit, "Binary templates for comma-free DNA codes," *Discrete Appl. Math.*, vol. 155, nos. 6–7, pp. 831–839, 2007.

[53] I. S. Sam, P. Devaraj, and R. Bhuvaneswaran, "An intertwining chaotic maps based image encryption scheme," *Nonlinear Dyn.*, vol. 69, no. 4, pp. 1995–2007, 2012.

[54] N. Masuda and K. Aihara, "Cryptosystems with discretized chaotic maps," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 49, no. 1, pp. 28–40, Jan. 2002.

[55] *15 Puzzle*. Accessed: Aug. 3, 2019. [Online]. Available: https://en.wikipedia.org/wiki/15_puzzle

[56] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: Harmony search," *J. Simul.*, vol. 76, no. 2, pp. 60–68, Feb. 2001.

[57] W. Kahan, "IEEE standard 754 for binary floating-point arithmetic," Dept. Elect. Eng. Comput. Sci., Univ. California, Oakland, CA, USA, Lect. Note 754, 1985.

[58] X. Chai, Y. Chen, and L. Broyde, "A novel chaos-based image encryption algorithm using DNA sequence operations," *Opt. Lasers Eng.*, vol. 88, pp. 197–213, Jan. 2017.

[59] J. Chen, Z.-L. Zhu, L.-B. Zhang, Y. Zhang, and B.-Q. Yang, "Exploiting self-adaptive permutation-diffusion and DNA random encoding for secure and efficient image encryption," *Signal Process.*, vol. 142, pp. 340–353, Jan. 2018.

[60] A. Kulsoom, D. Xiao, 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," *Multimedia Tools Appl.*, vol. 75, no. 1, pp. 1–23, Jan. 2016.

[61] Aqeel-ur-Rehman, X. Liao, A. Kulsoom, and S. Ullah, "A modified (dual) fusion technique for image encryption using SHA-256 hash and multiple chaotic maps," *Multimedia Tools Appl.*, vol. 75, no. 18, pp. 11241–11266, 2016.

[62] Aqeel-ur-Rehman, X. Liao, M. A. Hahsmi, and R. Haider, "An efficient mixed inter-intra pixels substitution at 2bits-level for image encryption technique using DNA and chaos," *Optik*, vol. 153, pp. 117–134, Jan. 2018.

[63] Y.-Q. Zhang and X.-Y. Wang, "A symmetric image encryption algorithm based on mixed linear–nonlinear coupled map lattice," *Inf. Sci.*, vol. 273, pp. 329–351, Jul. 2014.

[64] C. E. Shannon, "Communication theory of secrecy systems," *Bell Labs Tech. J.*, vol. 28, no. 4, pp. 656–715, Oct. 1949.

[65] X. Wu, J. Kurths, and H. Kan, "A robust and lossless DNA encryption scheme for color images," *Multimed Tools Appl.*, vol. 77, no. 10, pp. 12349–12376, 2018.

[66] L. Teng, X. Wang, and J. Meng, "A chaotic color image encryption using integrated bit-level permutation," *Multimedia Tools Appl.*, vol. 77, no. 3, pp. 6883–6896, 2018.

[67] T. Hu, Y. Liu, L.-H. Gong, S.-F. Guo, and H.-M. Yuan, "Chaotic image cryptosystem using DNA deletion and DNA insertion," *Signal Process.*, vol. 134, pp. 234–243, May 2017.

[68] C. Zhu, "A novel image encryption scheme based on improved hyperchaotic sequences," *Opt. Commun.*, vol. 285, no. 1, pp. 29–37, 2012.

[69] B. Norouzi and S. Mirzakuchaki, "A fast color image encryption algorithm based on hyper-chaotic systems," *Nonlinear Dyn.*, vol. 78, no. 2, pp. 995–1015, Oct. 2014.

[70] N. Taneja, B. Raman, and I. Gupta, "Combinational domain encryption for still visual data," *Multimedia Tools Appl.*, vol. 59, no. 3, pp. 775–793, 2012.

[71] B. Norouzi, S. M. Seyedzadeh, S. Mirzakuchaki, and M. R. Mosavi, "A novel image encryption based on hash function with only two-round diffusion process," *Multimedia Syst.*, vol. 20, no. 1, pp. 45–64, 2014.

[72] X. Wu, B. Zhu, Y. Hu, and Y. Ran, "A novel color image encryption scheme using rectangular transform-enhanced chaotic tent maps," *IEEE Access*, vol. 5, pp. 6429–6436, 2017.

[73] H. Diab, "An efficient chaotic image cryptosystem based on simultaneous permutation and diffusion operations," *IEEE Access*, vol. 6, pp. 42227–42244, 2018.

NADEEM IQBAL received the M.Phil. degree in computational science and engineering from NUST, Islamabad, Pakistan. He was with various academic institutions, and has supervised numerous bachelor's and master's students. He is currently an Assistant Professor with the School of Computing and Information Sciences, Imperial College of Business Studies (ICBS), Lahore, Pakistan. His current research interests include images cryptography, computer graphics, and philosophy of mathematics.

SAGHEER ABBAS received the M.Phil. degree in computer science and the Ph.D. degree from the School of Computer Science, NCBA&E, Lahore, Pakistan.

He has been teaching graduate and undergraduate students in computer science and engineering for the past eight years. He is currently an Assistant Professor with the School of Computer Science, NCBA&E. He has published about 60 research articles in international journals and reputed international conferences. His current research interests include cloud computing, the IoT, intelligent agents, image processing, and cognitive machines with various publications in international journals and conferences.

MUHAMMAD ADNAN KHAN received the Ph.D. degree from ISRA University, Pakistan. He held various academic and industrial roles in Pakistan. He has been teaching graduate and undergraduate students in computer science and engineering for the past ten years. He is currently an Assistant Professor with the School of Computer Science, NCBA&E, Lahore, Pakistan. He is currently supervising four Ph.D. and three M.Phil. Scholars. He has published about 120 research articles in international journals and reputed international conferences. His current research interests include MUD, channel estimation in multicarrier communication systems, and image processing and medical diagnosis using soft computing with various publications in journals and conferences of international repute.

TAHIR ALYAS received the M.Phil. degree in computer sciences from the Department of Computer Science, NCBA&E, Lahore, Pakistan, and the Ph.D. degree from the School of Computer Science, NCBA&E. He is currently an Assistant Professor with the Department of Computer Science, Lahore Garrison University, Lahore. His current research interests include cloud computing, the IoT, and Intelligence age.

AREEJ FATIMA received the M.Phil. degree in computer science from the Department of Computer Science, NCBA&E, Lahore, Pakistan, where she is currently pursuing the Ph.D. degree with the School of Computer Science. She is currently a Lecturer with the Department of Computer Science, Lahore Garrison University, Lahore. Her current research interests include cloud computing, the IoT, intelligent agents, image processing, and cognitive machines with various publications in international journals and conferences.

AIESHA AHMAD is currently an Assistant Professor with the Department of Computer Science, NCBA&E, Multan, Pakistan. Her current research interests include artificial intelligence, machine consciousness, deliberative and non-deliberative rationality evaluation, fuzzy modeling, and knowledge base systems.