International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A Novel Image Encryption Crypto-System Basedon Cascade Chaotic Systems and DNA Encoding

Mohammed Arif [1,3], Djamel Abed[2] , Abdelkader Medjouri[3], Salah Tlili[4]

Submitted:10/03/2024 **Revised**: 25/04/2024 **Accepted**: 02/05/2024

Abstract: In this paper, we proposed an efficient image encryption algorithm based on cascade chaotic Maps theory and DNA encoding rules. The Tent Map and Logistic Map are employed to generate all the parameters required by the proposed algorithm, while DNA encoding technology functions as an auxiliary tool. The proposed algorithm consists of the following steps: First, the Tent-Logistic Map is used to generate a key image, where its pixels are generated by chaotic sequences; second, both the plain image and the key image are encoded using DNA rules row by row, with different rows encoded according to various rules determined by the Logistic Map. After that, the encoded key image is used to perform DNA operations on the encoded plain image, row by row, to obtain an intermediate image, and the specific operation for each row is selected by the Logistic Map. Then, the intermediate image is decoded as the plain image for the next step. Finally, the above steps are repeated column-wise to generate the final cipher image. The experimental results and analysis indicate that the proposed algorithm is robust against typical attacks and exhibits strong security characteristics.

Keywords: *Image encryption. Chaos. DNA encoding. Tent-Logistic map. Logistic map*

1- Introduction

Image encryption algorithms based on DNA encoding and chaos represent a growing but evolving field. [1, 2, 3, 4, 5, 6, 7, 8]. Typically, chaos-based image encryption algorithms consist of two stages: scrambling and diffusion, which are performed sequentially or simultaneously [9, 10, 11, 12, 13, 14, 15, 16, 17]. On the other hand, algorithms that employ DNA encoding technology also involve two phases: encoding and decoding. In the encoding phase, the plain images are encoded using DNA rules, while a key image is generated and encoded as well. Then, certain DNA operations are carried out on the encoded key image and the encoded plain image. Finally, by decoding the intermediate image, the encrypted image is obtained. Throughout the process, all randomness elements are achieved through chaos. Chaos-based encryption algorithms process images in a mathematical context, while DNA encoding-based algorithms operate in a biological context [18].

In the proposed algorithm, specific DNA rules or operations are randomly determined by chaos, ensuring sufficient randomness. First, the Tent-Logistic Map is used to generate the key image. Next, both the plain image and the key image are encoded row by row using DNA rules selected from eight different types. Afterward, the two encoded images are processed row by row to generate the intermediate image, with the specific operation for each row determined by the Logistic Map. The intermediate image is then decoded as the plain image for the next step. Finally, these steps are repeated for the columns to produce the final encrypted image.

The remainder of this paper is organized as follows: related works are presented in Section 2; encryption and decryption algorithms are described in Section 3; deeper analysis on security and statistical aspects is provided in Section 4. Lastly, conclusions are outlined in Section 5, followed by the references.

2- Related works

2-1- *Tent-Logistic Map*

In the proposed algorithm, we use the Tent-Logistic Map and the Logistic Map to generate all the parameters required by the algorithm. According to references [2, 5, 7], the Tent-Logistic Map has several useful properties that assist in constructing the key image, while the Logistic Map performs auxiliary tasks such as selecting specific types of operations or DNA rules. The Tent-Logistic Map can be described by Eq (1), and the Logistic Map can be represented by Eq (2).

$$x_{n+1} = \begin{cases} aux_n(1-ux_n) & for\ x_n < 0.5 \\ au(1-x_n)\big(1-u(1-x_n)\big) & for\ x_n \geq 0.5 \end{cases} \quad (1)$$

Where $x_n \in (0, 1)$ and parameters a and u come from its two maps, the Logistic and Tent maps. Therefore, the

[1]*MESTEL Laboratory, Faculty of sciences and technology, University of Ghardaia, Algeria*
arif@univ-ghardaia.dz
[2]*LABCAV advanced control laboratory, Faculty of Sciences and Technology, University of Guelma, Algeria.*
[3]*LEVRES laboratory, University of EL Oued, 39000, El Oued, Algeria.*
[4] *LARENZA Laboratory, Faculty of Mathematics and Material Sciences, Kasdi Merbah Ouargla University, Algeria.*
tlilisalah2007@gmail.com

range of two parameters are $a\epsilon[0, 4]$ and $u\epsilon[0, 2]$. [19]

The bifurcation diagrams of the Tent-Logistic map has chaotic behaviors when $a\epsilon$ [3.57, 4] and $u\epsilon[1, 2]$, respectively. The Tent-Logistic map has wider chaotic ranges along with parameters a and u.

$$x_{n+1} = \mu x_n(1 - x_n) \quad (2)$$

where $x_n \in (0, 1)$ and $\mu \in (0, 4]$. From bifurcation diagram of logistic map, we can figure that when $\mu \in (3.9, 4]$, the pseudorandom sequence is in 0 and 1 [20]. In the proposed algorithm, we assign $\mu = 3.99999999$ [20].

2-2- *Deoxyribonucleic acid sequence*

Deoxyribonucleic acid (DNA) is a type of molecule composed of four types of nucleotides: Adenine (A), Thymine (T), Cytosine (C), and Guanine (G), along with other essential elements that form the principal components of nucleotides. According to DNA pairing rules, A pairs with T, and C pairs with G [20]. There are 8 valid combinations that comply with DNA complementary rules. These complementary rules share similarities with the binary system. For instance, '0' and '1' are complementary, just like '00' and '11', and '01' and '10' in the binary system. Since each pixel in a grayscale image contains 8 bits, for simplicity, we apply '00', '01', '10', and '11' as metadata. We can use the four types of nucleobases under 8 rules to encode the plain image. For example, if a pixel value is 201 in decimal, its corresponding binary value is '11001001'. By applying DNA rules, we can generate 8 possible combinations: 'TACG', 'TAGC', 'ATCG', 'ATGC', 'CGTA', 'CGAT', 'GCTA', and 'GCAT'. Furthermore, different DNA sequence operations can be applied to encrypt the image. DNA matching rules and operations are listed in the following tables, Table 1 to Table 4 [2, 5, 7, 8].

Table 1 Encoding and decoding rules

RULE	Rule1	Rule2	Rule3	Rule4	Rule5	Rule6	Rule7	Rule8
00	A	A	T	T	C	C	G	G
01	C	G	C	G	A	T	A	T
10	G	C	G	C	T	A	T	A
11	T	T	A	A	G	G	C	C

Table 2 Exclusive OR (XOR) Operation

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

Table 3 Addition (+) operation

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

Table 4 Subtraction (−) operation

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

2.3 *MD5 hash*

The Message Digest Algorithm 5 (MD5) is one of the most commonly used cryptographic hash functions, generating a 128-bit hash value typically represented as a 32-character hexadecimal number [21]. According to references [5, 22], MD5 is highly secure, as even a minor modification, such as a single bit change, can result in a significant difference between two hash outputs for different images. The MD5 hash value is used to generate the initial values for chaos maps, as indicated in Eq (3).

$$x_0 = mod(d_1 \oplus d_2 \oplus d_3 \oplus d_4, 256)/255 \quad (3)$$

where x_0 is the initial value of chaos map, the value of x_0 can be 0 or 1 with certain possibility,if this happened, skip this value, use eq. (3) to get another one. d_1, d_2, d_3, d_4 are extracted fromthe MD5 hash value of the plain image. The MD5 hash value of the plain image consists of128 bits, we use the first 32 bits to generate d_1, d_2, d_3, d_4, each of which is a single byte. For example, suppose the MD5 value is $b_1b_2b_3 \ldots b_{127}b_{128}$, d_1 is $b_1b_2b_3 \ldots b_7b_8$, d_2 is $b_9b_{10}b_{11} \ldots b_{15}b_{16}$, d_3 is $b_{17}b_{18}b_{19} \ldots b_{23}b_{24}$ and d_4 is $b_{25}b_{26}b_{27} \ldots b_{31}b_{32}$. Only need to transform d_1, d_2, $d_3$and d_4 from binary to decimal, before using Eq. (3).

3- Encryption and decryption algorithms

3-1- *Encryption algorithm*

Flow of the encryption algorithm is presented in Fig. 1, details are as follows:

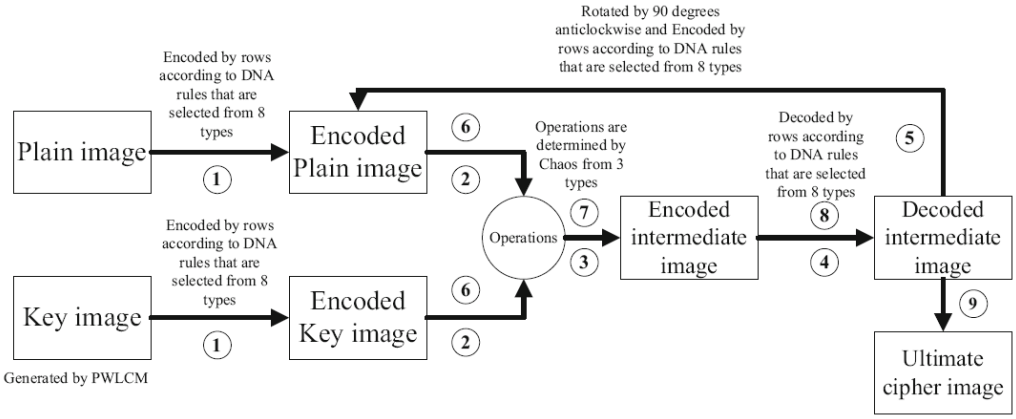

Fig. 1 Block diagram of encryption algorithm [18]

Step 1 Use Eq. (1) and Eq. (4) to generate the key image.

$$pixel = \lfloor x \times 256 \rfloor \quad (4)$$

Where *pixel* represents the pixel value of the key image, x∈(0,1) refers to the iteration value of the Tent-logistic map. Eq (1) and (4) are iterated to generate the key image. The initial value for Eq (1) is determined using Eq (3). The adjacent pixels in the key image are expected to exhibit weak correlation. To satisfy this condition, pixels derived from a chaos map are an ideal solution. The value of each successive pixel generated is independent of the preceding one.

Step 2 Encode plain image and key image respectively by rows with DNA rules that aredecided by Eq. (2) and Eq. (5).

$$Rule = \lfloor x \times 8 \rfloor + 1 \quad (5)$$

where "*Rule*" refers to the specific rule governing the encoding process, the initial value of Eq (1) is provided by Eq (3). The details of the DNA encoding rules are outlined in Table 1. Each pixel in a row is encoded using the specified DNA rule, with different rules applied to different rows, until all pixels in the image are encoded. Each pixel in a grayscale image consists of 8 bits, and based on the DNA encoding rules, these 8 bits are divided and encoded into four types of nucleobases. Assuming the original image has a size of M×N, where M is the width and N is the height, after encoding each row, the resulting image will have a size of 4×M×N.

Step 3 Perform DNA operations between the encoded plain image and the encoded key image on a row-by-row basis. The type of DNA operations to be applied is determined by Eq (2) and Eq (6). Details on DNA operations are presented in Table 2 to Table 4.

$$Operation = \lfloor x \times 3 \rfloor + 1 \quad (6)$$

where *operation* is the selected type of DNA operation. Perform the checked operation row by row until the encoded intermediate image is generated, during this time, three kinds of DNA operations (XOR, +, −) are alternatively performed. The size of encodedintermediate image is $4 \times M \times N$.

Step 4 Decode the encoded intermediate image to obtain a decoded intermediate image. The decoding rule is according to Eq. (5). Through this step we can get a primary cipher image. Randomly decoding and encoding enhance the performance of diffusion of the proposed algorithm.

The primary cipher image is with size $M \times N$.

3-2- *Decryption algorithm*

The decryption algorithm is the reverse process of the encryption algorithm. However, special attention must be given to certain details, particularly how to reverse the DNA-based subtraction operation. Before receivers can decode the cipher images, they must already possess the keys used to encrypt the plain images, which are transmitted prior to the cipher images. Once this is ensured, the decryption of the cipher images can be performed by following the steps below:

Step 1 Encode cipher image according to the exact inverse process which refers to rulesdescribed in Step 4 of Section "Encryption algorithm".

Step 2 Generate key image and Encode key image. The details refer to Step 1 to Step 2 in Encryption algorithm.

Step 3 Use encoded key image and encoded cipher image to generate the intermediate encoded image. The specific operation is described in Step 3 of Encryption algorithm.

Step 4 Decode the intermediate encoded image generated from Step 3 to get a decoded image.

Step 5 Rotate decoded image from Step 4 by 90° clockwise. So far, we have achieved thecipher image of plain image that is encrypted one round by rows.

Step 6 Do Step 1 to Step 4 again to obtain the plain image. Results of encryption and decryption are presented in Fig. 2.

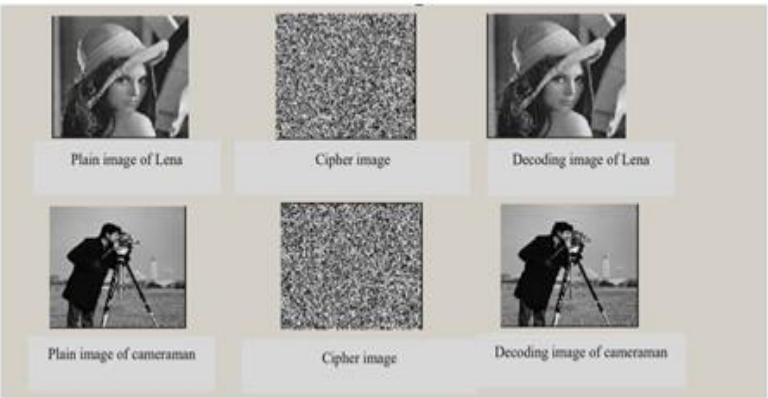

Fig. 2 Results of encryption and decryption

4- Security analysis and time complexity analysis

4-1- *Statistical analysis*

Histogram analysis and the correlation of adjacent pixels in cipher images are useful tools for evaluating the algorithm's robustness against statistical attacks. Ideally, an effective image encryption algorithm should demonstrate resistance to all forms of statistical attacks. The histogram represents the pixel intensity distribution across different gray levels. The more uniform the histogram of the cipher image, the stronger the performance of the proposed algorithm. Fig 3 presents the histograms of the Lena image and its corresponding cipher image.

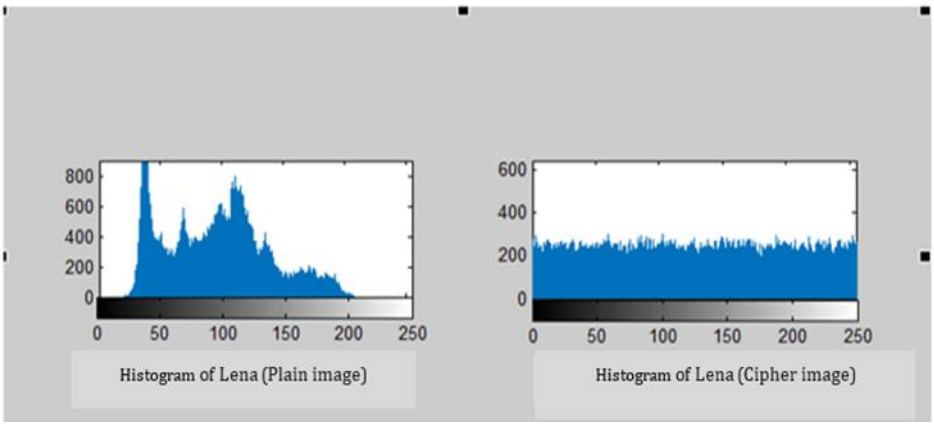

Fig. 3 Histograms of plain image and cipher image

In plain images, adjacent pixels exhibit a strong correlation. A robust image encryption algorithm should disrupt this correlation between pixels. To analyze the effectiveness of the algorithm, pairs of adjacent pixels are randomly selected from both the plain and cipher images in the horizontal, vertical, and diagonal directions. A detailed analysis of these correlations is shown in Fig. 4

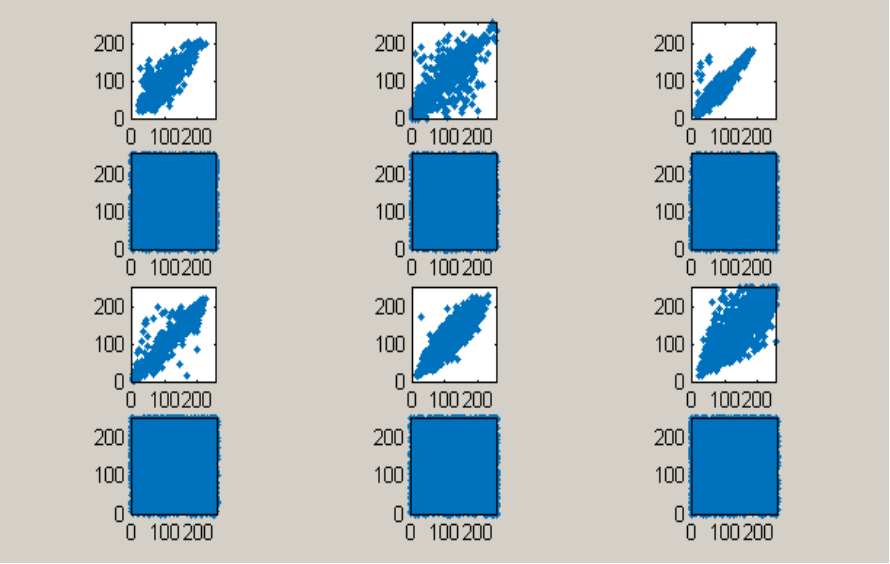

Fig. 4 Correlations of two adjacent pixels

The correlation coefficients $r_{x,y}$ are calculated to quantify the correlations among pixels byusing Eq. (7) and Eq. (8) [23], results are presented in Table 5.

$$\mathrm{cov}(\mathrm{x},\mathrm{y}) = \mathrm{E}\{(\mathrm{x}-\mathrm{E}(\mathrm{x}))(\mathrm{y}-\mathrm{E}(\mathrm{y}))\} \quad (7)$$

$$r_{x,y} = \frac{\mathrm{cov}(\mathrm{x},\mathrm{y})}{\sqrt{\mathrm{D}(\mathrm{x})}\sqrt{\mathrm{D}(\mathrm{y})}} \quad (8)$$

where x and y are values of two adjacent pixels in the plain image or cipher image,

$$\mathrm{E}(\mathrm{x}) = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}} \mathrm{x}_\mathrm{i}, \mathrm{D}(\mathrm{x}) = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}} (\mathrm{x}_\mathrm{i} - \mathrm{E}(\mathrm{x}))^2$$

We calculate the correlation coefficients of the input images and their corresponding cipher images, and the correlation coefficients are shown in Table 5. It can be seen from Table 5 that the correlation coefficients of the input images are close to 1, whereas the correlation coefficients of the cipher images are close to 0, which indicates that the adjacent pixels of cipher images are uncorrelated.

Table 5 Correlation coefficients of two adjacent pixels in the plain image and cipher image

Image	Plain image			Cipher image		
	Horizontal	Diagonal	Vertical	Horizontal	Diagonal	Vertical
Brain	0.9328	0.8934	0.9288	−0.0066	−0.0023	0.0061
Circuit	0.9484	0.9067	0.9620	−0.0014	-0.0032	0.0010
Lena	0.9294	0.9372	0.9668	- 0.0028	−0.0016	0.0011
Cameraman	0.9205	0.8995	0.9444	0.0054	- 0.0014	−0.0051
Peppers	0.9602	0.9452	0.9686	- 0.0016	0.0010	0.0016
Barbara	0.9053	0.8831	0.9612	0.0018	0.0027	0.0017

4-2- *Key space analysis*

4-2-1- *Key space*

An eligible image encryption algorithm should be sensitive to its keys and has a big key space [24]. From Fig.5, we can see that x_0, x_0', x_0'' u and a are secret keys, x_0 is the initial value of Eq. (2), u is the parameter of Eq. (2). x_0' is the initial value of Eq.(1), u and a are the parameters of Eq. (1). x_0'' is the initial value used to drive chaos system when encoding images. The key space of this proposed algorithm is $10^{16} \times 10^{16} \times 10^{16} \times 10^{15} = 10^{79} \approx 2^{263}$ [8].

4-2-2- *Sensitivity to secret key*

In this proposed algorithm, the initial value x_0' and parameter u of Eq. (2), the initial value x_0' and parameters u and a of Eq.(1), and the initialvalue x_0'' used to activate chaos

when encoding images are secret keys. In our experiment, $a = 3.96$ and $u = 1.74$ for Eq.(1). Results are presented in Fig.5.

Fig. 5 Sensibility of secret keys

4-3- *Information entropy*

Information entropy is used to measure the complexity of a system. This concept is employed by researchers to evaluate the effectiveness of encryption algorithms. It can be calculated using Eq. (9).

$$H(x) = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i) \qquad (9)$$

Where x represents a set of symbols, n is the total number of symbols, with each $x_i \in x$, and $p(x_i)$ is the probability of x_i in x. Theoretically, the higher the value of H(x), the better the encryption algorithm performance. For example, in experiments using 256 gray-level images, the theoretical information entropy of cipher images should be 8.[18] Table 6 presents the evaluation results of cipher images encrypted by the proposed algorithm. From Table 6, it can be observed that the actual information entropies of various cipher images are close to 8, indicating strong encryption performance. Additionally, the table provides a comparison between the cipher images and the corresponding plain images.

Table 6 Information entropies of plain images and cipher images

Name	Peppers	Circuit	Brain	Barbara	Cameraman	Lena
Cipher image	7.9992	7.9963	7.9985	7.9991	7.9974	7.9966
Plain image	6.7744	7.8234	6.1527	6.4963	6.8363	7.2789

4-4- *Analysis of resisting differential attack*

To withstand differential attacks, cipher images must be highly sensitive to their corresponding plain images, meaning that a small change in the plain image should cause a significant alteration in the cipher image. In the proposed algorithm, a checksum, derived from the MD5 hash of the plain image, is used as the initial value for the Tent-logistic map to generate all necessary parameters. This establishes a direct relationship between the plain and cipher images. Benchmark tests, such as *the Number of Pixels Change Rate* (NPCR) and *Unified Average Changing Intensity* (UACI), are employed to assess the impact of minor changes on the entire cipher image [12]. Let the plain image and its corresponding cipher image be denoted as P_1 and C_1. A single pixel in the plain image is altered by incrementing its value by 1 at a randomly selected position. The modified plain image and its cipher image are denoted as P_2 and C_2. An auxiliary matrix D is then constructed, where D(i, j) = 0 if $C_1(i, j) = C_2(i, j)$, and D(i, j) = 1 otherwise. NPCR and UACI are defined as:

$$NPCR = \frac{\sum_{(i,j)} D(i,j)}{M \times N} \times 100\% \qquad (13)$$

$$UACI = \frac{1}{M \times N}\left(\sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{255}\right) \times 100\% \qquad (14)$$

A pixel is randomly chosen from the plain image and changed its value by adding 1. NPCR refers the changed pixel numbers in cipher image that only one pixel is changed in the plain image. UACI indicates the average value of difference between two cipher images. The maximum theoretical values are 99.609375 % for NPCR and 33.463541 % for UACI [18]. Dueto various development of environment, the actual values can be floating near the standard values. Results are shown in Table7, which are based on the average value of the experiment.From Table 7 we can infer that the proposed algorithm has good character in resisting differential attacks.

Table 7 NPCRs and UACIs

Name	Peppers	Circuit	Brain	Barbara	Cameraman	Lena
NPCR	84.36	85.93	81.06	83.43	83.02	86.32
UACI	29.19	29.80	27.88	29.16	28.52	30.63

5- Conclusions

In this paper, we propose a novel image encryption algorithm based on cascade chaotic systems and DNA encoding rules. DNA encoding is an emerging but well-established technology, naturally suited for biological information storage. Due to its excellent information processing capabilities, it is widely applied in the field of image encryption.

In the proposed algorithm, the plain image is treated as a sequence of rows. The algorithm processes the image row by row, from top to bottom. For each row, one of eight possible DNA encoding rules is applied, with the specific rule determined randomly by a chaotic map. Simultaneously, an encoded key image is generated. After encoding the entire plain image, each row of the encoded plain image is combined with a corresponding row from the encoded key image to generate a row of encoded cipher data. The encoded cipher image is then decoded to obtain an intermediate cipher image, which is used as a new plain image for subsequent iterations until the final cipher image is produced. Importantly, the operations performed during this process are selected randomly. Experimental results demonstrate that the proposed algorithm is capable of securely encrypting images.

References

[1] Belazi A, Hermassi H, Rhouma R, Belghith S (2014) Algebraic analysis of a RGB image encryption algorithm based on DNA encoding and chaotic map. Nonlinear Dynam 76(4):1989–2004

[2] Enayatifar R, Abdullah AH, Isnin IF (2014) Chaos-based image encryption using a hybrid genetic algorithm and a DNA sequence. Opt Lasers Eng 56:83–93

[3] Huang XL, Ye GD (2014) An image encryption algorithm based on hyper-chaos and DNA sequence. Multimed Tools Applic 72(1):57–70

[4] Jain A, Rajpal N (2015) A robust image encryption algorithm resistant to attacks using DNA and chaotic logistic maps. Multimed Tools Applic 74:1–18

[5] Liu HJ, Wang XY (2012) Image encryption using DNA complementary rule and chaotic maps. Appl SoftComput 12(5):1457–1466

[6] Tang Z, Zhang X, Lan W (2015) Efficient image encryption with block shuffling and chaotic map. MultimedTools Applic 74:5429–5448

[7] Wang XY, Zhang YQ, Bao XM (2015) A novel chaotic image encryption scheme using DNA sequenceoperations. Opt Lasers Eng 73:53–61

[8] Zhang Y, Wen W, Su M, Li M (2014) Cryptanalyzing a novel image fusion encryption algorithm based on DNA sequence operation and hyper-chaotic system. Optik-Int J Light Electron Optics 125(4):1562–1564

[9] Akhavan A, Samsudin A, Akhshani A (2015) Cryptanalysis of an improvement over an image encryption method based on total shuffling. Opt Commun 350:77–82

[10] Behnia S, Akhshani A, Ahadpour S, Mahmodi H, Akhavan A (2007) A fast chaotic encryption scheme based on piecewise nonlinear chaotic maps. Phys Lett A 366(4):391–396

[11] Fouda JAE, Effa JY, Sabat SL, Ali M (2014) A fast chaotic block cipher for image encryption. Commun Nonlinear Sci Numer Simul 19(3):578–588

[12] Hussain I, Shah T, Gondal MA (2014) Image encryption algorithm based on total shuffling scheme and chaotic S-box transformation. J Vib Control 20(14):2133–2136

[13] Mannai O, Bechikh R, Hermassi H et al. (2015) A new image encryption scheme based on a simple first- order time-delay system with appropriate nonlinearity. Nonlinear Dynam 1–11

[14] Roohbakhsh D, Yaghoobi M (2015) Fast adaptive image encryption using chaos by dynamic state variablesselection. Int J Comput Applic 113(12)

[15] Vahidi J, Gorji M, Mazandaran I (2014) The confusion-diffusion image encryption algorithm with dynamical compound chaos. J Math Comput Sci (JMCS) 9(4):451–457

[16] Wang XL, Zhang HL (2015) A color image encryption with heterogeneous bit-permutation and correlatedchaos. Opt Commun 342:51–60

[17] Zhang Y, Xiao D, Wen W, Li M (2014) Breaking an image encryption algorithm based on hyper-chaotic system with only one round diffusion process. Nonlinear Dynam 76(3):1645–1650

[18] Xingyuan Wang, Chuanming Liu (2016) A novel and effective image encryption algorithm based on

chaos and DNA encoding, Multimed Tools Appl, Springer Science 10.1007/s11042-016-3311-8

[19] Yicong Zhou, Zhongyun Hua, Chi-Man Pun, C. L. Philip Chen, (2014) Cascade Chaotic System With Applications, IEEE TRANSACTIONS ON CYBERNETICS, 10.1109/TCYB.2014.2363168

[20] Wang XY, Wang Q, Zhang YQ (2014) A fast image algorithm based on rows and columns switch. Nonlinear Dynam 79(2):1141–1149

[21] Rivest R (1992) The MD5 message-digest algorithm

[22] Cheng H, Huang C, Ding Q et al. (2014) An efficient image encryption scheme based on ZUC stream cipher and chaotic logistic map. In Intelligent data analysis and its applications, volume II (pp. 301–310). SpringerInternational Publishing

[23] Hussain I, Shah T, Gondal MA, Mahmood H (2013) A novel image encryption algorithm based on chaotic maps and GF (28) exponent transformation. Nonlinear Dynam 72(1–2):399–406

[24] Monaghan DS, Gopinathan U, Naughton TJ, Sheridan JT (2007) Key-space analysis of double random phase encryption technique. Appl Opt 46(26):6641–6647