Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 18(4): 3887–3906.
DOI: 10.3934/mbe.2021194
Received: 13 March 2021
Accepted: 28 April 2021
Published: 06 May 2021

Research article

A new DNA coding and hyperchaotic system based asymmetric image encryption algorithm

Min Liu, and Guodong Ye*

Faculty of Mathematics and Computer Science, Guangdong Ocean University, Zhanjiang 524088, China

* **Correspondence:** Email: guodongye@hotmail.com.

Abstract: In this paper, an asymmetric image encryption algorithm based on DNA coding and hyperchaotic system is designed. Unlike other image encryption schemes, for example, sharing of same private keys between sender and receiver, and fixed rules with simple operation, three stages are studied as follows to deal with these problems. Firstly, to eliminate the possible risk of key transmission and management, the initial values of the hyperchaotic system are generated for ahead by the RSA (Rivest-Shamir-Adleman) algorithm and the plain image, in which the sum of odd rows, even rows, odd columns, and even columns are computed respectively to extra the plain message from the plain image as input of RSA algorithm. Then, a mathematical map is established to transform all of them into initial values of the hyperchaotic system. Secondly, the pixel level permutation is performed to confuse the image according to the chaotic sequences generated. Finally, to solve the problem of fixed rules with simple operations in current DNA based image encryption algorithms, dynamical DNA encryption is designed to diffuse the permuted image. The process of DNA encryption includes DNA coding, DNA operation and DNA decoding. In particular, DNA rules are selected according to chaotic sequences dynamically, rather than fixed rules with simple operation. Theoretical analysis and numerical simulations show that the proposed algorithm is secure and reliable for image encryption.

Keywords: DNA; hyperchaotic system; image encryption; RSA; security

1. Introduction

With the rapid development of smart phones and other intelligent terminals, multimedia communication is becoming more and more important and indispensable in today's society, for example, image, video, and audio. However, the openness and sharing of networks bring many security risks to digital communication. As an important medium of multimedia communication, digital image plays an important role in military, medical, biological and social life. Digital image carries a lot of useful information, which is widely spread communicated over the open Internet. However, some of them are private or contain sensitive information. So, it is an urgent task to protect the security of image content. Compared with traditional texts or words, images have some special characteristics such as large amount of data, high redundancy, strong correlation between adjacent pixels, and low information entropy. Therefore, persons need a long keystream to encrypt images [1] for security requirement.

Recently, chaos as a new interdisciplinarity has shown its inherent advantages such as pseudo-random and ergodic. It is a deterministic but unpredictable nonlinear system, which is very sensitive to each initial condition and parameter. Therefore, many image encryption algorithms, techniques and methods based on chaotic systems have been proposed. In [2], Ye *et al.* proposed a new meaningful image encryption algorithm based on compressive sensing [3] and random numbers insertion using information hiding technology, in which the measurement matrix for compressive sensing is generated by a low dimensional complex chaotic tent-sine system. In [4], a novel multi-image visually encryption algorithm based on compressive sensing and Schur decomposition was proposed. Chaotic logistic sine coupling map was employed to confuse the part Hadamard matrix. By regarding two secret images as the real part and imaginary part respectively, a double image encryption [5] by using random binary encoding and gyrator transform was proposed. Liu and Wang [6] proposed an image encryption algorithm based on one-time keys. The piecewise linear chaotic map was used to produce pseudo-random keystream sequences. By designing a dynamic random growth technique [7], a chaotic block image encryption algorithm was proposed with a diffusion process depending on both the key and the plain image. Although those schemes take some measurements to improve the security during the encryption process, they cannot completely resist the chosen plaintext and known plaintext attacks. For example, Ma *et al.* [8] gave a thorough security analysis on the algorithm [9] and made successful break on it. Moreover, many pixel-level based encryption algorithms or improved schemes have been proposed such as [10].

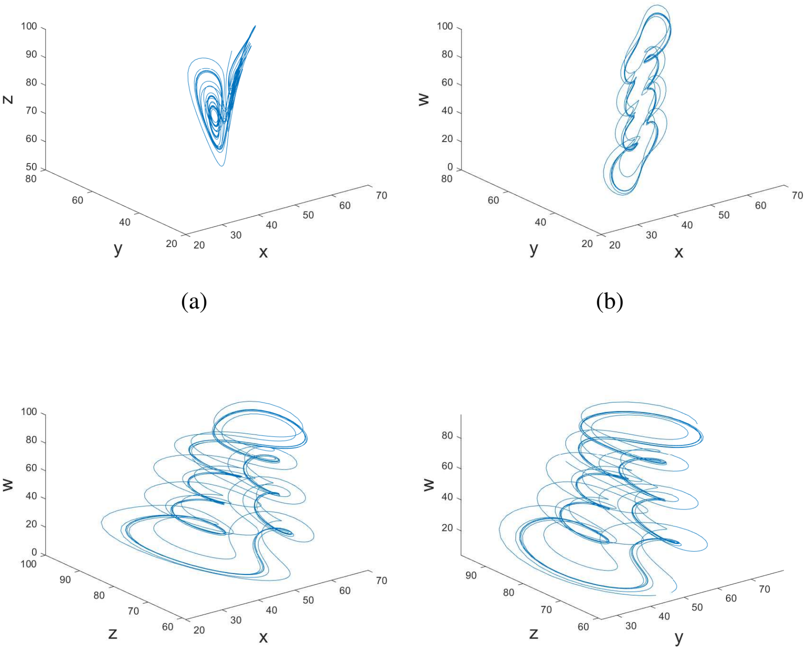
To solve the above problem, a new local image encryption scheme [14] was proposed, in which the scheme encrypted only the necessary part of the sensitive information in the frequency domain of lifting wavelet transform (LWT). Due to the advantages of bit-level operation, some image encryption algorithms using bit-level permutation have been proposed [15–17]. The reason is that it can change pixel position and alter their values at the same time. For example, in [18], heterogeneous bit-permutation was employed to reduce computation cost and change pixel values at the permutation stage. Li *et al.* [19] presented a new bit-level based image encryption algorithm. They diffused two binary sequences mutually in the diffusion stage. Moreover, An new image cryptosystem adopting bit-level permutation by Arnold cat map with a diffusion was designed in [20].

Considering the complex biological structure, some cryptographic algorithms based on DNA coding have also been proposed [21–23] for image encryption. DNA molecule has the advantages of high information density, good parallelism and ultra-low energy consumption, which is suitable in the field of cryptography [24, 25]. The core of these algorithms is DNA coding and DNA operating, including DNA complementarity, DNA addition, DNA subtraction and DNA XOR. Yildirim [26] studied a DNA encoding based image encryption algorithm together with neuron model, circuit design with memristor, and chaos theory. In [27], an image encryption algorithm based on DNA coding and two Logistic maps was proposed. However, Hermassi *et al.* [28] pointed out that the encryption method [27] has serious defects, that is, it is non-invertible even having correct keys, and cannot resist the known plaintext attack. Furthermore, the key can be recovered by a pair of plain images and corresponding cipher images. The encryption process is also not sensitive to the change of plain image or key. In addition, the rules of DNA coding and DNA decoding are fixed. To improve the sensitivity, hash values of the plain image were also seen as a part of the key [29]. However, one needs to transfer the corresponding image hash values when decrypting the cipher images.

For symmetric ciphers such as [30–33], the sender and the receiver should share the same keys for communication. Therefore, it is hard to ensure the secure exchange of keys in a special channel. On the contrary, asymmetric ciphers can solve the problems of key transmission and management. So, the public cryptosystem based encryption algorithms have been introduced in the protection of image content [34,35]. For example, the El-Gamal encryption algorithm [34] was employed to encrypt the permuted image followed by scrambling to pixel locations in the confusion and diffusion stages. However, the keystream is only with key dependence in these algorithms. So, it cannot frustrate the known plaintext and chosen plaintext attacks. Furthermore, traditional modular multiply operation suffers much time cost. The new methods to improve the operation speed have also been studied more and more in recent years [36–39]. For example, Huang and Wang [36] proposed a novel and efficient design for an RSA cryptosystem with a large key space, in which a new modular multiplier architecture was proposed different from traditional method. In [37], a high frequency as well as low latency RSA cryptosystem was presented to perform efficient hardware implementation. By an improved processor architecture and software implementation, Reference [38] designed new hardware stages at the same time uses an open-source big number library to execute the operation. Adiono *et al* [39] proposed a Montgomery multiplier hardware design with primitive gates, adders, shifters, multiplexers, and registers to compute RSA. All these are aiming to speed up the computation.

In this paper, to avoid the problems seen above, an asymmetric image encryption scheme based on RSA and a four-dimensional hyperchaotic system is proposed. Firstly, using RSA to generate initial values for hyperchaotic system and produce the random number sequences needed for image encryption. Especially, the plaintext related messages are extracted from odd rows, even rows, odd columns, and even columns respectively as the input of RSA. Then, pixel-level permutation is employed to permute the plain image. Secondly, using DNA coding and DNA operation to diffuse the permuted image. The combination of pixel-level permutation and DNA coding diffusion can improve the security of our cryptosystem. Moreover, different DNA decoding rules are used to fetch the cipher image. Compared with the existing image encryption algorithms based on hyperchaotic systems, the proposed algorithm has three advantages: (1) Different DNA coding and DNA decoding rules are designed. (2) The rules of DNA coding, DNA decoding and DNA operating are all related to chaotic sequences dynamically. (3) The initial values of the hyperchaotic system are related to the plain image by using the RSA algorithm and a mathematical map. As a result, the sender in our algorithm does not need to transfer the extra transmission to the receiver. Both sender and receiver do not need to manage the secret keys.

This paper is organized as follows. In Section 2, some preliminary knowledge are introduced, which are used in the proposed algorithm including RSA, DNA, and hyperchaotic system. The proposed image encryption scheme is described in detail in Section 3. In Section 4, some simulations are carried out with test results displayed. In Section 5, many security analyses are evaluated. Section 6 gives some conclusions for this paper.



(a) (b)

(c) (d)

Figure 1. Chaotic attractors in 3D: (a) $z - y - x$ plane, (b) $w - y - x$ plane, (c) $w - z - x$ plane, (d) $w - z - y$ plane.

2. Preliminary knowledge

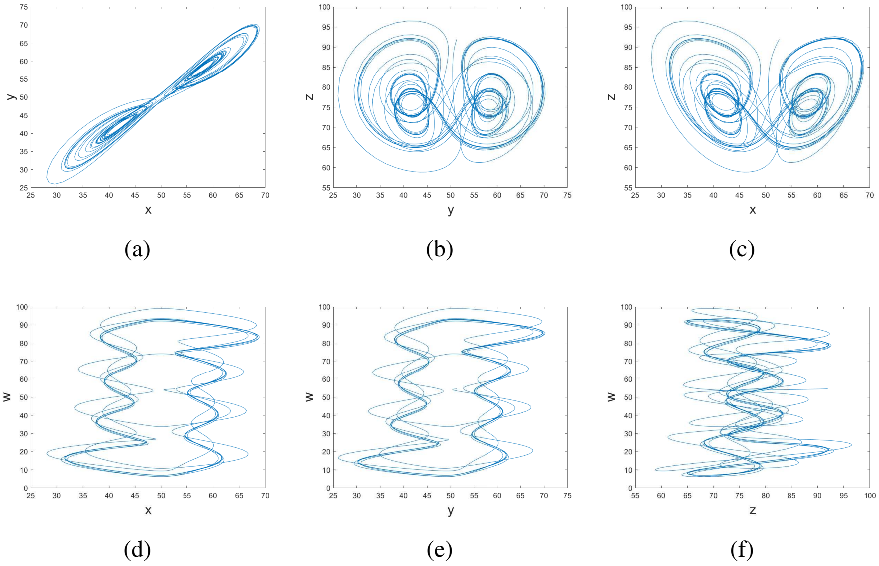
2.1. Hyperhaotic system

Hyperchaotic can show good sensitivity to initial conditions [40] and has the characteristics of random like behavior, which provides a new way to improve the security of encryption system. A new four-dimensional hyperchaotic system was studied in [41] which can be defined by the following equation.

$$\begin{cases} \dot{x} = a(y - x), \\ \dot{y} = cx - y - xz + u, \\ \dot{z} = xy - bz, \\ \dot{w} = -kx. \end{cases} \tag{2.1}$$

where a, b, c and k are the system parameters. If one set a, b, c, and k be 35, 83, 28, and 5, respectively, the Lyapunov exponents of above hyperchaotic system are [41]: $\lambda_1 = 0.3997$, $\lambda_2 = 0.3113$, $\lambda_3 = 0$ and $\lambda_4 = -14.3776$. Apparently, the above system is in a hyperchaotic state due to double positive Lyapunov exponents. With parameters $a = 35$, $b = 83$, $c = 28$, $k = 5$ and all initial conditions (x_0, y_0, z_0, w_0) = $(-10, -10, -10, -10)$, Figure 1 and Figure 2 depict chaotic attractors by different real axis, while Figure 3 shows the time series for differen planes.

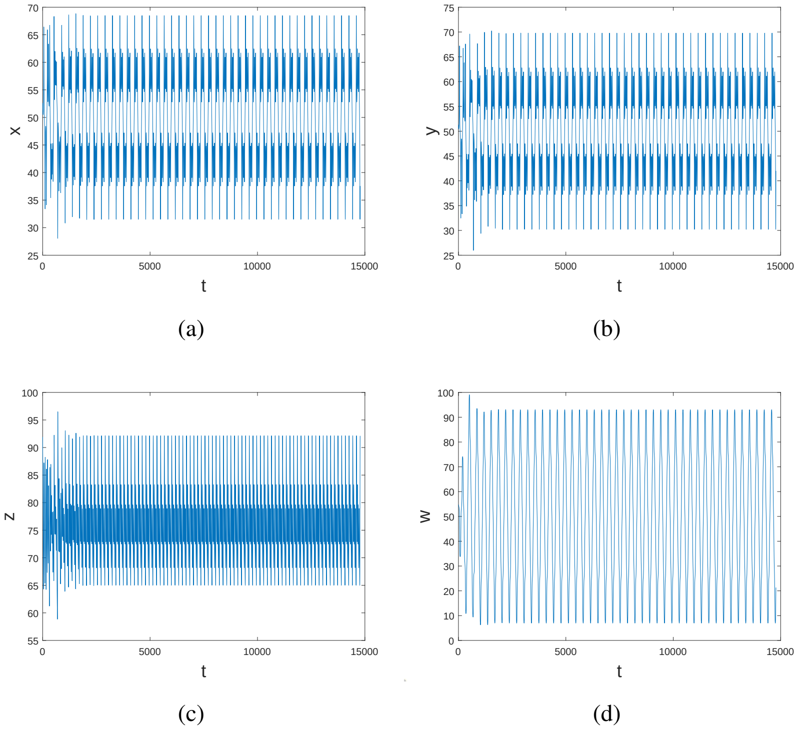
To further show the good statistical properties: unpredictability, randomness, independence, and uniform distribution, the NIST SP800-22 is tested for the above hyperchaotic system with results listed



(a) (b) (c)

(d) (e) (f)

Figure 2. Chaotic attractors in 2D: (a) $x - y$ plane, (b) $y - z$ plane, (c) $x - z$ plane, (d) $x - w$ plane, (e) $y - w$ plane, (f) $z - w$ plane.



(a) (b)

(c) (d)

Figure 3. Time series in 1D: (a) x plane, (b) y plane, (c) z plane, (d) w plane.

Table 1. NIST SP800-22 test for outputs of hyperchaotic system.

Subtest	P value	Status
The Frequency (Monobit) Test	0.7144	Pass
Frequency Test within a Block	0.3933	Pass
The Runs Test	0.7794	Pass
Tests for the Longest-Run-of-Ones in a Block	0.3528	Pass
The Binary Matrix Rank Test	0.1143	Pass
The Discrete Fourier Transform (Spectral) Test	0.2457	Pass
The Non-overlapping Template Matching Test	0.6781	Pass
The Overlapping Template Matching Test	0.7659	Pass
Maurers Universal Statistical Test	0.5071	Pass
The Linear Complexity Test	0.8287	Pass
The Serial Test	0.5233	Pass
The Approximate Entropy Test	0.5485	Pass
The Cumulative Sums (Cusums) Test	0.9963	Pass
The Random Excursions Test	0.3731	Pass
The Random Excursions Variant Test	0.6189	Pass

Algorithm 1 RSA cryptography

```
Input: Big prime numbers $p$ and $q$, plain message $m$.
Output: cipher message $c$.
(1) The receiver: Compute $n = p \times q$, $\varphi(n) = (p-1)(q-1)$. Select a public key $e$ satisfying $gcd(e, \varphi(n)) = 1$, $1 < e < \varphi(n)$. Compute private key $d$, that is, $d \times e \equiv 1 \ mod \ \varphi(n)$
(2) The sender: Encrypt $c = m^e \ mod \ n$.
(3) The receiver: Decrypt $m = c^d \ mod \ n$.
```

in Table 1. So, we can see that all items can pass the NIST SP800-22 test and it is suitable to be the randomness of random number generator (TRNG) and pseudorandom number generator (PRNG).

2.2. RSA cryptography

RSA algorithm is a public key cryptosystem widely used in data security transmission, in which the public key is open for doing encryption which is different to the decryption key seen as private key. Anyone can encrypt the message with the public key, but it can only be decrypted correctly by the person who knows the private key. The security of RSA cryptography is dependent on the difficulty of factoring the product of large primes. Algorithm 1 gives the process of RSA as follows.

Table 2. The xnor operation.

Input 1	Input 2	Output
1	0	0
1	1	1
0	1	0
0	0	1

2.3. Operation of xnor

The xnor operation is defined binary numbers 0 and 1 as: if the input variables are the same, the output is 1, while the input variables are different, the output is 0. The calculation results are presented in Table 2.

2.4. DNA coding

A DNA sequence contains four bases, i.e., A (adenine), C (cytosine), G (guanine) and T (thymine), in which A and T are one pair, and C and G are another pair. In order to comply with the complement rules, eight coding schemes are commonly considered as listed in Table 3.

In the process of image encryption, the gray pixel value can be represented by a 8-bit binary number and with a 4-bit DNA sequence. For example, a pixel value is 210, then its binary number is expressed as 11010010. Using DNA rules, we can get eight combinations: TCAG, TGAC, ACTG, AGTC, GACT, GTCA, CAGT and CTGA. In addition, operations for DNA addition, subtraction, xor and xnor are listed in Tables 4, 5, 6, and 7, respectively.

Table 3. DNA pair rule.

Rule	1	2	3	4	5	6	7	8
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 4. DNA addition operation.

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

Table 5. DNA subtraction operation.

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

Table 6. DNA xor operation.

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

Table 7. DNA xnor operation.

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

3. Algorithm description

3.1. Encryption algorithm

Step 1: For an plain image A with size $M \times N$, suppose the operation of RSA is $Rfunction$, the initial values of the hyperchaotic system are computed with a mathematical model as seen in Algorithm 2.

Step 2: Generate four chaotic sequences by iterating the initial values x_0, y_0, z_0, and w_0 into hyperchaotic system to get $\{x, y, z, w\}$.

Step 3: Use the following equation for x, y, z and w to obtain new sequences x', y', z', and w' as,

$$\begin{cases} x' = mod(x \times 10^{14}, M) + 1 \\ y' = mod(y \times 10^{14}, N) + 1 \\ z' = mod(z \times 10^{14}, M) + 1 \\ w' = mod(w \times 10^{14}, N) + 1 \end{cases} \tag{3.1}$$

Step 4: Confuse the plain image using Algorithm 3 to scramble all the pixel positions.

Although the four sequences are not the same, the repeated permutation of pixels may occur, resulting in the degradation of encryption effect, so it is necessary to remove the repeated permutation.

Algorithm 2 The generation of initial values

```
Input: plain image $A$.
Output: initial values $x_0$, $y_0$, $z_0$, and $w_0$.
(1) Calculate plain messages $m_1$, $m_2$, $m_3$, and $m_4$, where, $m_1$ is the sum of the pixel values for all odd rows, $m_2$ is the sum of the pixel values for all even rows, $m_3$ is the sum of the pixel values for all odd columns, $m_4$ is the sum of the pixel values for all even columns.
(2) Encrypt $m_i$ by $Rfunction$, i.e. $c_i = Rfunction(m_i)$, $i = 1, 2, 3, 4$.
(3) Establish a new mathematical map $X_i = \frac{\sqrt{\ln(c_i+1)}}{\sqrt{\ln(c_i+1)}+1} \times 10 + \frac{1}{\sqrt{m_i}} + 3$, $i = 1, 2, 3, 4$.
(4) Let $x_0 = -X_1$, $y_0 = -X_2$, $z_0 = -X_3$, and $w_0 = -X_4$.
```

Algorithm 3 Permutation operation

```
Input: Plain image $A$, and $x'$, $y'$, $z'$, $w'$.
Output: Permuted image $A'$.
(1) For $i \leftarrow 1$ do
(2) Delete repeat numbers in $x'$, $y'$, $z'$, and $w'$.
(3) Swap the values of $A(x'_i, y'_i)$ and $A(z'_i, w'_i)$.
(4) Get new image $A'$.
```

Step 5: Construct a sequence $x'' = mod(floor(x \times 10^{14}), 256)$, and do the xor operations for each row and column in A' as Algorithm 4,

Step 6: Convert A'' to a binary number matrix to obtain matrix B.

Step 7: Construct four sequences as,

$$\begin{cases} xx = mod(x \times 10^{14}, 8) + 1 \\ yy = mod(y \times 10^{14}, 256) + 1 \\ zz = mod(z \times 10^{14}, 4) + 1 \\ ww = mod(w \times 10^{14}, 8) + 1 \end{cases} \tag{3.2}$$

Step 8: By using xx as index to choose DNA encoding rule, perform it on B to obtain matrix B'

Algorithm 4 XOR operation

```
Input: $A'$, $x''$.
Output: $A''$.
(1) for $i \leftarrow 1$ to $M$ do
(2) $AA'(i, :) = xor(A'(i, :), x''_i)$.
(3) for $j \leftarrow 1$ to $N$ do
(4) $A''(:, j) = xor(AA'(:, j), x''_{M+j})$
(5) Get $A''$
```

Algorithm 5 DNA encoding

```
Input: $B$, $xx$.
Output: $B'$.
(1) for $i \leftarrow 1$ to $M$ do
(2) $B'(i,:) = D\_encode(B(i,:), xx_i)$.
```

Algorithm 6 DNA encoding

```
Input: $H$, $xx$.
Output: $H'$.
(1) for $i \leftarrow 1$ to $M$ do
(2) $H'(i,:) = D\_encode(H(i,:), xx_i)$.
```

with M rows and $4 \times N$ columns. Algorithm 5 shows the process, where $D\_encode(R, s)$ is a function that implements DNA encoding on R using xx by eight combinations in Table 3.

Step 9: Convert sequence yy to a matrix H of M rows and N columns, and perform similarly on H using DNA encoding according to coding rule xx to obtain matrix H' with M rows and $4 \times N$ columns, seeing following algorithm 6.

Step 10: Perform a DNA operation on the each row of matrix B' and matrix H' according to the corresponding operation rule in zz to obtain matrix C. Algorithm 7 shows the process with function $D\_opera(P, Q, k)$ for performing DNA operations, where P and Q represent DNA coding sequences with the same dimensions. Set $k = 1, 2, 3, 4$, in which the number 1 represents the addition of P and Q, the number 2 represents the subtraction of P and Q, the number 3 represents the xor operation of P and Q, and the number 4 represents the xnor operation of P and Q.

Step 10: Decode matrix C into an image E with decimal numbers according to decoding rule ww, and get the cipher image. Algorithm 8 shows the process with function $D\_decode(R, s)$ to implement DNA sequence decoding, where R is a DNA sequence to be decoded, and $s = 1, 2, 3, 4, 5, 6, 7, 8$ respectively represent the eight combinations in Table 3.

3.2. Decryption algorithm

Decryption is the inverse of encryption. First, we need to compute the chaotic sequences generated by the hyperchaotic system using private key and cipher message $c_i, i = 1, 2, 3, 4$. Then, the inverse op-

Algorithm 7 DNA operation

```
Input: $H'$, $B'$, $zz$.
Output: $C$.
(1) for $i \leftarrow 1$ to $M$ do
(2) $C(i,:) = D\_opera(H'(i,:), B'(i,:), zz_i)$
```

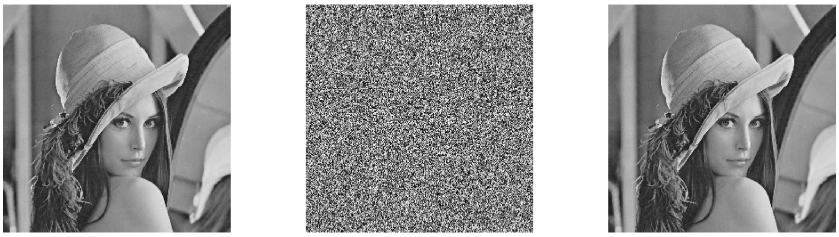
Algorithm 8 DNA decoding

```
Input: $C$, $ww$.
Output: $E$.
(1) for $i \leftarrow 1$ to $M$ do
(2) $E(i, :) = D\_decode(C(i, :), ww_i)$
```

eration of decoding is performed on the cipher image E. Next, the inverse operation of DNA operations is performed. Finally, we get the plain image A through the inverse operation of pixel permutation.

4. Experimental results

MATLAB of 2019b version including Main Toolbox, Control System Toolbox, Image Processing Toolbox, and Signal Processing Toolbox is used to perform encryption and decryption to test images on a computer Windows 10 with AMD FX-8300 3.3 GHz CPU, and 8GB RAM. Lena image with size 256 × 256 is randomly selected. Figure 4 shows the results of encryption and decryption.



(a) (b) (c)

Figure 4. Test for Lena: (a) plain image, (b) cipher image, (c) correct decrypted image.

5. Performance analysis

5.1. Information entropy analysis

Information entropy [42] can display a quantitative reflection of the uncertainty for an image. The greater the information entropy, the greater the uncertainty, that is, the stronger randomness of the image. The calculation formula of image information entropy is following.

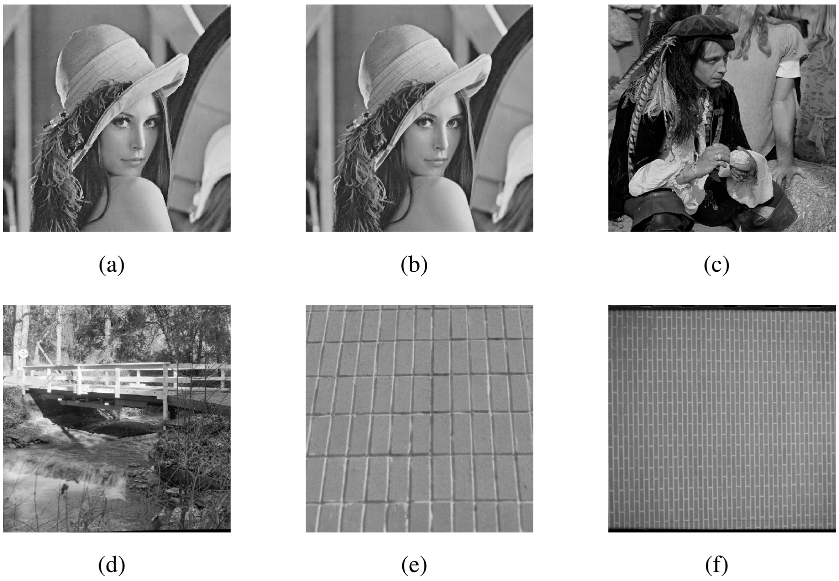
$$H(x) = -\sum_{i=1}^{L} p(x_i) \log_2 p(x_i) \tag{5.1}$$

where $p(x_i)$ is the probability of symbol x_i and L is the total number of x_i. The ideal value of entropy for an 8-bit gray scale image is 8. If the value is closer to 8, the more uncertain the image is.

The information entropy tests for different images, Lena$_{(256 \times 256)}$, Lena$_{(512 \times 512)}$, male$_{(1024 \times 1024)}$, Bridge$_{(512 \times 5126)}$, brick wall$_{(512 \times 512)}$, and brick wall$_{(1024 \times 1024)}$, are listed in Table 8 with test images shown in Figure 5. Therefore, the simulation results tell us that the pixel distribution in the cipher image is very random by using the proposed algorithm.

Table 8. Information entropy tests.

	Lena	Lena	Male	Bridge	Brick wall	Brick wall
Size	256×256	512×512	1024×1024	512×512	512×512	1024×1024
Results	7.9976	7.9994	7.9998	7.9993	7.9993	7.9998



(a) (b) (c)

(d) (e) (f)

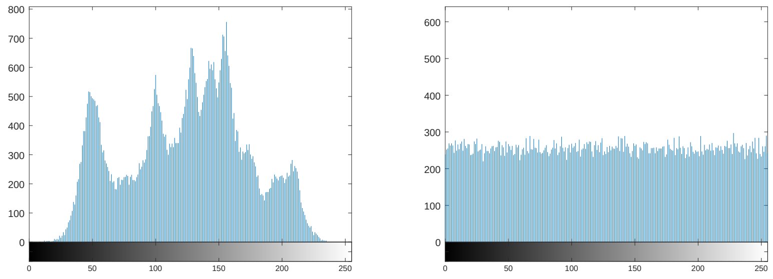
Figure 5. Test images: (a) Lena with size 256×256, (b) Lena with size 512×512, (c) Male with size 1024×1024, (d) Bridge with size 512×512, (e) Brick wall with size 512×512, (f) Brick wall with size 1024×1024.

5.2. Key space analysis

For a secure algorithm, the key space should keep large enough to resist brute force attacks. Moreover, the keys should be easy to establish and exchange [43] for practical communication. In our algorithm, the keys consist of the four plain messages extracted from the plain image. Then, they are converted into the initial conditions of the hyperchaotic system. To simply compute, the size of the key space is determined by the initial value of the hyperchaotic system indirectly. The hyperchaotic system has four initial values x_0, y_0, z_0, w_0. When the precision is set to be 10^{-14}, the key space can reach 10^{56}, which is about 2^{186} and bigger than 2^{100}. Therefore, the key space of our algorithm is large, and it can resist the brute force attack.

5.3. Histogram analysis

Histogram describes the distribution of pixel values for an image visually, which is an important feature to reflect whether the designed algorithm can resist the attack of statistical analysis. If the distribution is not uniform, attackers can obtain a certain amount of plain information through statistical analysis. So, as a good encryption system, the histogram of cipher image should be evenly distributed. Figure 6 shows the histograms before and after encryption for the image Lena. Obviously, one can find that the histogram of plain image before encryption is not uniform, while the histogram of cipher image becomes uniform. Therefore, the proposed algorithm shows good performance.



(a) (b)

Figure 6. Histogram test for Lena: (a) plain image, (b) cipher image.

5.4. Correlation coefficients analysis

Generally, there is a strong correlation between adjacent pixels of a natural meaningful plain image. In order to analyze and compare the correlation between adjacent pixels in the plain image and its corresponding cipher image, 10000 pairs of adjacent pixels in each direction are randomly selected for test. Figure 7 shown the correlation distribution of two adjacent pixels in three directions. It can be seen that the adjacent pixels of the plain image have a strong linear relationship, while the adjacent pixels of the cipher image have a random relationship, which indicates that the redundancy and correlation of the cipher image are removed by using our method.

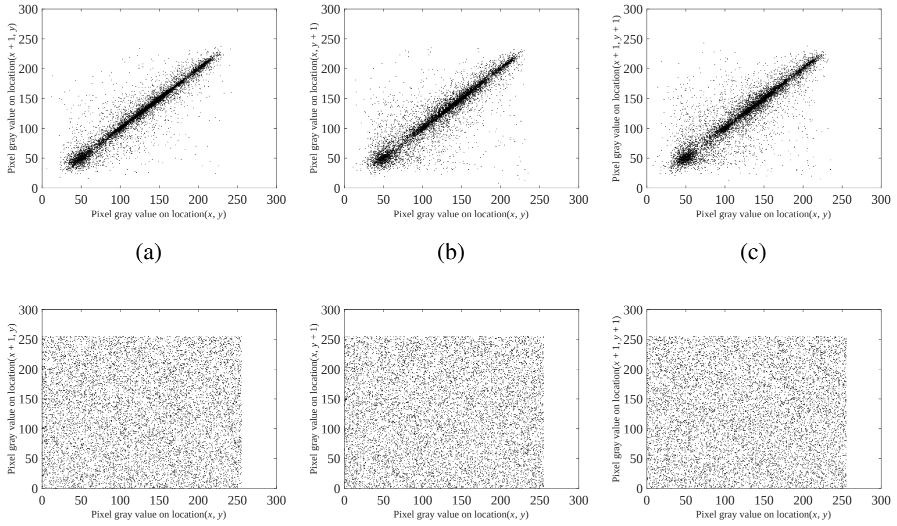
More specially, for calculating the correlation coefficient for a message, the equations can be seen as following.

$$E(x) = \frac{1}{K}\sum_{i=1}^{K} x_i \tag{5.2}$$

$$D(x) = \frac{1}{K}\sum_{i=1}^{K}(x_i - E(x))^2 \tag{5.3}$$

$$Cov(x, y) = \frac{1}{K}\sum_{i=1}^{K}(x_i - E(x))(y_i - E(y)) \tag{5.4}$$

$$r_{xy} = \frac{Cov(x, y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{5.5}$$



(a) (b) (c)

(d) (e) (f)

Figure 7. Correlation analysis for: plain image Lena: (a) horizontally, (b) vertically, (c) diagonally; cipher image Lena: (d) horizontally, (e) vertically, (f) diagonally.

where x and y are the gray values of two adjacent pixels in the image and k is the total number of pixels. Table 9 lists the test results. So, we can see that the correlation coefficient of the plain image is close to 1, while the correlation coefficient of the cipher image is close to 0. Therefore, the proposed algorithm can effectively resist statistical attacks.

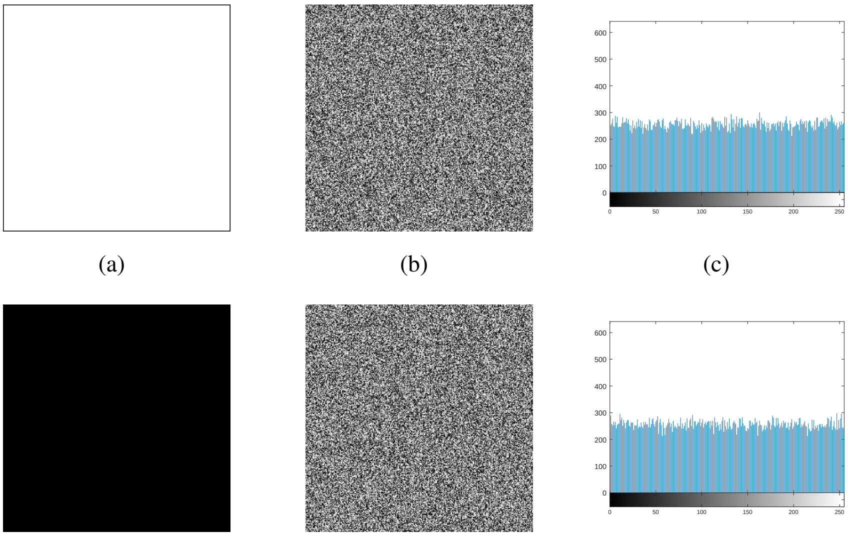
Table 9. Correlation between adjacent pixels.

Directions	Horizontally	Vertically	Diagonally
Plain image	0.9460	0.9017	0.8842
Cipher image	-0.0007	-0.0033	-0.0060

5.5. Chosen plaintext attack and known plaintext attack analysis

In this paper, the plain message is extracted from the plain image, then, the RSA algorithm is employed to compute the cipher message. Both of them are designed to generate the initial values for the hyperchaotic system. Consequently, the keystream used in our encryption algorithm has a high connection with the plain image. Fortunately, due to the public key cryptosystem, the sender and receiver do not need to save and manage the same secret key. Different images will produce different keystreams, that is to say, the chosen plaintext attack and known plaintext attack are difficult to break the proposed algorithm.

Besides, some attacks may use a black image or white image as a special pure image to attack encryption algorithms. Figure 8 shows the plain images with pure black and white images , cipher images with their histograms. So, the pixels in the cipher image are uniformly distributed with random noise, and it is impossible to obtain useful information from the image to crack the encryption algorithm. Therefore, our encryption algorithm can effectively resist the chosen plaintext attack and known plaintext attack.



(a) (b) (c)

(d) (e) (f)

Figure 8. Tests for special images: (a) white image, (b) cipher image of (a), (c) histogram for (b), (d) black image, (e) cipher image of (d), (f) histogram for (e).

5.6. UACI and NPCR analysis

To measure quantitatively the difference between two images, The number of pixels change rate (NPCR) and the unified average changing intensity (UACI) defined as the following equations are commonly used as test tools.

$$\text{NPCR} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} F(i, j) \times 100\% \tag{5.6}$$

$$\text{UACI} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{|P_1(i, j) - P_2(i, j)|}{255} \times 100\% \tag{5.7}$$

where P_1 is the cipher image, and P_2 is another cipher image just with a pixel changed in the same plain image. If $P_1(i, j)$ is equal to $P_2(i, j)$, then $F(i, j)$ is equal to 0; otherwise, $F(i, j)$ is equal to 1. Lena image is randomly chosen for test. The pixel value in the first row and the first column is changed from 165 to 166. The NPCR value is 99.6185%, and the UACI is 33.4058%. Table 10 also lists the results of some other images. Therefore, our algorithm has a strong ability to resist differential attacks.

5.7. Comparisons

As mentioned above, the proposed encryption algorithm based on the idea RSA, permutation, confusion, and DNA operation is presented. The beauty of our work is that we have achieved very compet-

Table 10. Results of NPCR and UACI.

image	image size	NPCR	UACI
Lena	256×256	99.6185%	33.4058%
Lena	512×512	99.6128%	33.4671%
Male	1024×1024	99.6059%	33.4266%
Brick wall	512×512	99.6136%	33.4665%

itive and promising results. To show better performance, some other similar studies [44–47] are also compared (by Lena image) with our work as listed in Table 11 and Table 12. So, one can see that the proposed algorithm has a good effect for image encryption.

Table 11. Comparisons of information entropy and correlation coefficient.

Methods	image size	entropy	horizontally	vertically	diagonally
Proposed	256×256	7.9976	-0.0007	-0.0033	-0.0060
Proposed	512×512	7.9994	-0.0011	0.0014	0.0043
Ref. [44]	256×256	7.9975	0.0085	0.0054	0.0049
Ref. [45]	256×256	7.9971	0.0038	0.0024	0.0051
Ref. [46]	512×512	7.9994	-0.0164	-0.0083	0.0080
Ref. [47]	256×256	7.9974	0.0010	-0.0001	0.0014

Table 12. Comparisons of NPCR and UACI.

Methods	image size	NPCR	UACI
Proposed	256×256	99.6185%	33.4058%
Proposed	512×512	99.6128%	33.4671%
Ref. [31]	256×256	99.6173%	33.5831%
Ref. [34]	512×512	99.61%	33.4902%
Ref. [44]	256×256	99.636%	33.465%
Ref. [45]	256×256	99.6%	33.4%
Ref. [46]	512×512	99.64%	33.42%
Ref. [47]	256×256	99.62%	33.44%

6. Conclusions

Combing with the RSA algorithm and DNA coding, a new asymmetric image encryption algorithm has been proposed in this paper. Compared with current encryption schemes, our contributions are: (1) The process of DNA coding, operation, and decoding are not fixed but dynamically. (2) Odd rows, even rows, odd columns, and even columns in the plain image are extracted as plain message and as the input of RSA. (3) A new mathematical model is established to map both plain message and cipher message got from RSA into initial values for the hyperchaotic system. The advantages are: (1) The proposed encryption algorithm can make the encryption process be associated with the plain image by RSA. (2) No extra transmission is needed because of the open of the cipher messages produced from plain messages by RSA. (3) Different plain images would generate different keystreams with respect to different plain messages. Due to the binary operations in DNA codes like most current DNA based image encryption algorithms, our algorithm still needs much time to encrypt larger images. So, in the future, we will focus on improving the operation time for the encryption process and hardware implement.

Acknowledgments

The authors would like to thank the reviewers for their valuable comments and the editor's helpful suggestions. This work was supported in part by the National Natural Science Foundation of China (No.61972103); the Natural Science Foundation of Guangdong Province of China (No.2019A1515011361); and the Key Scientific Research Project of Education Department of Guangdong Province of China (No.2020ZDZX3064).

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Y. Zhang, Test and verification of AES used for image encryption, *3D Res.*, **9** (2018), 3.
2. G. D. Ye, C. Pan, Y. X. Dong, Y. Shi, X. L. Huang, Image encryption and hiding algorithm based on compressive sensing and random numbers insertion, *Signal Process*, **172** (2020), 107563.
3. L. H. Gong, K. D. Qiu, C. Z. Deng, N. R. Zhou, An image compression and encryption algorithm based on chaotic system and compressive sensing, *Opt. Laser Technol.*, **115** (2019), 257–267.
4. G. D. Ye, C. Pan, Y. X. Dong, K. X. Jiao, X. L. Huang, A novel multi-image visually meaningful encryption algorithm based on compressive sensing and Schur decomposition, *Transact. Emerg. Telecommun. Technol.*, **32** (2021), e4071.
5. Z. J. Liu, Q. Guo, L. Xu, M. A. Ahmad, S. T. Liu, Double image encryption by using iterative random binary encoding in gyrator domains, *Opt. Express*, **18** (2010), 12033–12043.
6. H. J. Liu, X. Y. Wang, Color image encryption based on one-time keys and robust chaotic maps, *Comput. Math. Appl.*, **59** (2010), 3320–3327.
7. X. Y. Wang, L. T. Liu, Y. Q. Zhang, A novel chaotic block image encryption algorithm based on dynamic random growth technique, *Opt. Laser. Eng.*, **66** (2015), 10–18.
8. Y. L. Ma, C. Q. Li, B. Ou, Cryptanalysis of an image block encryption algorithm based on chaotic maps, *J. Inf. Secur. Appl.*, **54** (2020), 102566.
9. L. F. Liu, S. D. Hao, J. Lin, Z. Wang, X. Y. Hu, S. X. Miao, Image block encryption algorithm based on chaotic maps, *IET Signal Process*, **12** (2018), 22–30.

10. W. H. Liu, K. H. Sun, C. X. Zhu, A fast image encryption algorithm based on chaotic map, *Opt. Laser. Eng.*, **84** (2016), 26–36.

11. Z. Y. Hua, Y. C. Zhou, H. J. Huang, Cosine-transform-based chaotic system for image encryption, *Inform. Sci.*, **480** (2019), 403–419.

12. M. Alawida, J. S. Teh, A. Samsudin, W. H. Alshoura, An image encryption scheme based on hybridizing digital chaos and finite state machine, *Signal Process*, **164** (2019), 249–266.

13. H. J. Li, Y. R. Wang, Z. W. Zuo, Chaos-based image encryption algorithm with orbit perturbation and dynamic state variable selection mechanisms, *Opt. Laser. Eng.*, **115** (2019), 197–207.

14. A. Belazi, A. A. A. El-Latif, A. V. Diaconu, R. Rhouma, S. Belghith, Chaos-based partial image encryption scheme based on linear fractional and lifting wavelet transforms, *Opt. Laser. Eng.*, **88** (2017), 37–50.

15. M. Kumari, S. Gupta, A. Malik, A superlative image encryption technique based on bit plane using key-based electronic code book, *Multimed. Tools Appl.*, **79** (2020), 33161–33191.

16. K. Shahna, A. Mohamed, A novel image encryption scheme using both pixel level and bit level permutation with chaotic map, *Appl. Soft Comput.*, **90** (2020), 106162.

17. L. Teng, X. Y. Wang, A bit-level image encryption algorithm based on spatio temporal chaotic system and self-adaptive, *Opt. Commun.*, **285** (2012), 4048–4054.

18. X.Y. Wang, H. L. Zhang, A color image encryption with heterogeneous bit-permutation and correlated chaos, *Opt. Commun.*, **342** (2015), 51–60.

19. L. Xu, Z. Li, J. Li, W. Hua, A novel bit-level image encryption algorithm based on chaotic maps, *Opt. Laser. Eng.*, **78** (2016), 17–25.

20. Z. L. Zhu, W. Zhang, K. W. Wong, H. Yu, A chaos-based symmetric image encryption scheme using a bit-level permutation, *Inform. Sci.*, **181** (2011), 1171–1186.

21. X. L. Chai, Y. R. Chen, L. Broyde, A novel chaos-based image encryption algorithm using DNA sequence operations, *Opt. Laser. Eng.*, **88** (2017), 197–213.

22. H. J. Liu, X. Y. Wang, A. kadir, Image encryption using DNA complementary rule and chaotic maps, *Appl. Soft Comput.*, **12** (2012), 1457–1466.

23. D. Ravichandran, A. S. Banu, B. K. Murthy, V. Balasubramanian, S. Fathima, R. Amirtharajan, An efficient medical image encryption using hybrid DNA computing and chaos in transform domain, *Med. Biol. Eng. Comput.*, **59** (2021), 589–605.

24. X. P. Wei, L. Guo, Q. Zhang, J. X. Zhang, S. G. Lian, A novel color image encryption algorithm based on DNA sequence operation and hyper-chaotic system, *J. Syst. Software*, **85** (2012), 290–299.

25. R. Enayatifar, F. G. Guimaraes, P. Siarry, Index-based permutation-diffusion in multiple-image encryption using DNA sequence, *Opt. Laser. Eng.*, **115** (2019), 131–140.

26. M. Yildirim, DNA encoding for RGB image encryption with memristor based neuron model and chaos phenomenon, *Microelectron. J.*, **104** (2020), 104878.

27. Q. Zhang, L. Guo, X. P. Wei, Image encryption using DNA addition combining with chaotic maps, *Math. Comput. Model.*, **52** (2010), 2028–2035.

28. H. Hermassi, A. Belazi, R. Rhouma, S. M. Belghith, Security analysis of an image encryption algorithm based on a DNA addition combining with chaotic maps, *Multimed. Tools Appl.*, **72** (2014), 2211–2224.

29. S. Q. Zhu, C. X. Zhu, W. H. Wang, A new image encryption algorithm based on chaos and secure hash sha-256, *Entropy*, **20** (2018), 716.

30. E. E. García-Guerrero, E. Inzunza-González, O. R. López-Bonilla, J. R. Cárdenas-Valdez, E. Tlelo-Cuautle, Randomness improvement of chaotic maps for image encryption in a wireless communication scheme using PIC-microcontroller via Zigbee channels, *Chaos Soliton. Fract.*, **133** (2020), 109646.

31. E. Rodríguez-Orozco, E. E. García-Guerrero, E. Inzunza-Gonzalez, O. R. López-Bonilla, A. Flores-Vergara, J. R. Cárdenas-Valdez, et al., FPGA-based chaotic cryptosystem by using voice recognition as access key, *Electronics*, **7** (2018), 414.

32. C. Tanougast, Hardware implementation of chaos based cipher: Design of embedded systems for security applications, *Chaos-Based Cryptography*, **354** (2011), 297–330.

33. A. Flores-Vergara, E. E. García-Guerrero, E. Inzunza González, O. R. López-Bonilla, E. Rodrguez-Orozco, J. R. Cárdenas-Valdez, et al., Implementing a chaotic cryptosystem in a 64-bit embedded system by using multiple-precision arithmetic, *Nonlinear Dynam.*, **96** (2019), 497-516.

34. S. F. Yousif, A. J. Abboud, H. Y. Radhi, Robust image encryption with scanning technology, the El-Gamal algorithm and chaos theory, *IEEE Access*, **8** (2020), 155184–155209.

35. X. Z. Dong, L. Zhang, X. W. Gao, An efficient FPGA implementation of ECC modular inversion over F256, *Proceedings of the 2nd International Conference on Cryptography, Security and Privacy*, (2018), 29–33.

36. X. M. Huang, W. Wang, A novel and efficient design for an RSA cryptosystem with a very large key size, *IEEE T. Circuits-II*, **62** (2015), 972–976.

37. S. D. Thabah, M. Sonowal, R. U. Ahmed, P. Saha, Fast and area efficient implementation of RSA algorithm, *Proceed. Computer Sci.*, **165** (2019), 525–531.

38. G. C. Marchesan, N. R. Weirich, E. C. Culau, I. I. Weber, F. G. Moraes, E. Carara, et al., Exploring RSA performance up to 4096-bit for fast security processing on a flexible instruction set architecture processor, *IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, (2018), 18420517.

39. T. Adiono, H. Ega, H. Kasan, S. Fuada, S. Harimurti, Full custom design of adaptable montgomery modular multiplier for asymmetric RSA cryptosystem, *International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, (2017), 17524004.

40. H. Chen, C. Tanougast, Z. J. Liu, B. Y. Hao, Securing color image by using hyperchaotic system in gyrator transform domains, *Opt. Quant. Electron.*, **48**(2016), 396.

41. N. Zhu, Y. T. Wang, J. Liu, J. H. Xie, H. Zhang, Optical image encryption based on interference of polarized light, *Opt. Express*, **17** (2009), 13418–13424.

42. G. D. Ye, K. X. Jiao, H. S. Wu, C. Pan, X. L. Huang, An asymmetric image encryption algorithm based on a fractional-order chaotic system and the RSA public-key cryptosystem, *Int. J. Bifurcat. Chaos*, **30** (2020), 2050233.

43. J. S. Teh, M. Alawida, Y. C. Sii, Implementation and practical problems of chaos-based cryptography revisited, *J. Inf. Secur. Appl.*, **50** (2020), 102421.

44. T. Wang, M. H. Wang, Hyperchaotic image encryption algorithm based on bit-level permutation and DNA encoding, *Opt. Laser Technol.*, **132** (2020), 106355.

45. R. S. Ye, Y. T. Xi, Y. L. Ma, A chaotic image encryption scheme using swapping based confusion approach, *IEEE international conference on computer communication and the internet*, (2016), 16525393.

46. X. L. Chai, Z. H. Gan, M. H. Zhang, A fast chaos-based image encryption scheme with a novel plain image-related swapping block permutation and block diffusion, *Multimed. Tools Appl.*, **76** (2017), 15561–15585.

47. X. L. Xue, D. S. Zhou, C. J. Zhou, New insights into the existing image encryption algorithms based on DNA coding, *PLoS One*, **15** (2020), e0241184.

AIMS Press