Nonlinear Dyn (2016) 84:833–849
DOI 10.1007/s11071-015-2530-2

ORIGINAL PAPER

A coding and substitution frame based on hyper-chaotic systems for secure communication

Shun Zhang · Tiegang Gao

Received: 12 July 2015 / Accepted: 27 November 2015 / Published online: 21 December 2015

Abstract Secure communication has always been important ever since ancient times. Both encryption and steganography are effective ways for secure communication. Inspired by the central dogma that regulates the transfer of genetic information in molecular biology, this paper introduces a coding and substitution frame for both encryption and steganography. The frame consists of three parts: the construction of the pseudo-codon table, the encoding of the original media and additional data, and the substitution. After the original media is encoded, it is substituted codon by codon according to the encoded additional data and a constructed pseudo-codon table. Hyper-chaotic systems have been used in the construction of the pseudo-codon table, the encoding of original media and additional data, and the substitution, all of which have offered more randomness and larger key space to the frame. To demonstrate the validity and efficiency of the frame, this paper offers the implementation of the frame in image encryption and in image steganography. The frame offers more security to steganography algorithms and better efficiency to encryption algorithms. Experiments and analysis demonstrate the excellent performances of the steganography and encryption schemes under the proposed frame.

S. Zhang (✉) · T. Gao
College of Software, Nankai University, Wei Jin Road No. 94, Nankai District, Tianjin 300071, China
e-mail:shentengvip@gmail.com;
zhangshun@mail.nankai.edu.cn

Keywords Secure communication · DNA coding · Arbitrary N-nary system · Hyper-chaotic system

1 Introduction

In the modern times, more and more information are transmitting on the Internet. However, due to its original design in transfer protocols, the Internet is insecure for information transmission. Therefore, secure information transmission on the Internet is quite important. Transmitting secure information on insecure channels has always been the key task in information transmission since ancient times. Two ways can achieve such a goal—steganography and encryption.

Steganography is a process that links two sets—the cover media and the additional information. It merges these two sets and expresses features of the cover media but conceals features of the additional information at the same time. In that way, it conceals the existence of the secure additional information. Many image steganography schemes have been proposed since the end of last century. The Least Significant Bit (LSB) based [1] steganography schemes are simple and efficient. Additional information is hidden into images by substituting the least significant bits of pixels. Steganography schemes based on compression [2] make use of the redundancy of cover media. The character of cover media limits the capacity and quality of steganography. Difference expansion-based steganography schemes [3,4] hide additional information by extending the difference between neighbor pixels. The distortion to the image increases quickly along with the increase in additional information. Steganography schemes based on histogram modification [5] cause less distortion to the cover image. The peak points of the histogram limit the payload. Recently, chaotic maps have also been used in the steganography [6].

Encryption is a traditional way. With some complicated computing methods, it destroys the format of original media, relieves the relations between those basic units that compose the original media, and rearranges those basic units into another form, which is unrecognizable in some degree. Most encryption schemes in the fields of communication follow two basic steps—confusion and diffusion which is proposed by Shannon [7]. According to the basic unit involved, image encryption schemes can be classified into two categories. The traditional image encryption schemes are operated on the pixel level [8–13], while some newly proposed schemes are operated on the bit level [14–18]. With different permutation methods, such as Arnold cat map [19], constructed shuffling matrix [10,11,14], and chaotic permutations [9,11,12,15,16,20], encryption schemes based on permutation can achieve nice visual results. Permutation on the pixel level does not change the statistical features, such as histogram, of the original images. However, if conducted on the bit level, permutation will be more efficient. Experimental results have verified that even if only permutation is imposed on the bit level, encryption can achieve excellent results. The intrinsic features in the bit-level distribution of images were explored in Ref. [18]. It also proposed a bit-level permutation scheme to realize the encryption. Permutation schemes on bit level [14–17] are simple and efficient. However, some statistical features may reveal the permutation rule if the permutation is not complicated enough. For example, the histogram after encryption in Ref. [14] has obvious rhythm. Recently, many encryption schemes based on chaotic systems have been proposed. Encryption schemes based on chaotic systems can achieve better performances in the correlation coefficients, information entropy, and histogram, etc. Chaotic systems are utilized for the generation of permutation/shuffling matrix [14] in the confusion process, or for the generation of bit streams for the exclusive OR (XOR) operation [8,11,13,15,20,21] in the diffusion process. It can also be used in other ways. For example, in Ref. [13] the chaotic system is used for the generation of delay time. Recently, some novel encryption schemes based on fractional-order chaotic system have been proposed [22], which has achieved nice results.

Different from the above-mentioned steganography and encryption schemes, this paper brings out a novel frame to implement steganography and encryption for secure communication. It is a simple coding and substitution frame. However, it can also achieve excellent results in both encryption and steganography. Besides, encryption schemes based on the proposed frame are more efficient. Moreover, steganography schemes based on the proposed frame offer more security. There are three coding part for the final substitution, as described in the basic flow chart in Fig. 1. The first part is the construction of pseudo-genetic codon table. The pseudo-genetic codon table is a random coding-grouping table, which is similar to the genetic codon table (Table 2). The second part is the processing of the original media. According to a DNA coding method, original media is encoded into codon sequence. Then, with the randomly constructed pseudo-codon table and the encoded original media, we construct an arbitrary N-nary system for coding the additional information. The last part is the processing of additional information. The additional information is encoded into arbitrary N-nary form according to the constructed arbitrary N-nary system. Finally, the substitution part substitutes the encoded codons of the original media with codons in the same group according to the pseudo-codon table. A random number generator based on the hyper-chaotic system is designed to offer the hyper-chaotic randomness to the frame. It is used in the construction of pseudo-codon table, the encoding, and deciding of codons in the substitution.

The rest of the paper is organized as follows: Sect. 2 details the proposed frame and discusses the possibility of the implementation in encryption and steganography. The implementation of the proposed frame is offered in Sect. 3. Based on the implementation in Sect. 3, simulation and analysis are presented in Sect. 4. Section 5 draws the conclusion.

2 Details of the frame

The coding and substitution frame includes coding of the original media and additional information, pseudo-genetic codon table construction, arbitrary N-nary sys-

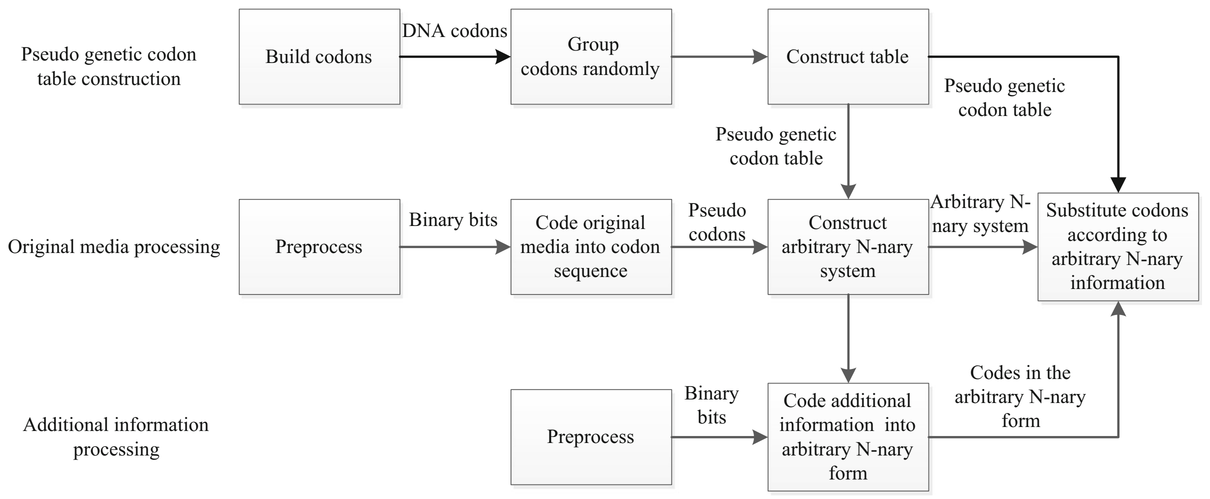

Pseudo genetic codon table construction
Build codons
DNA codons
Group codons randomly
Construct table
Pseudo genetic codon table
Pseudo genetic codon table
Original media processing
Preprocess
Binary bits
Code original media into codon sequence
Pseudo codons
Construct arbitrary N-nary system
Arbitrary N-nary system
Substitute codons according to arbitrary N-nary information
Additional information processing
Preprocess
Binary bits
Code additional information into arbitrary N-nary form
Codes in the arbitrary N-nary form

Fig. 1 Flow chart of the proposed frame

tem construction, and substitution. They are introduced in detail in the followings.

2.1 DNA coding

DNA computing was firstly proposed by Adleman [23] in 1994 for solving the Hamiltonian path problem. In the following years, some NP-complete problems were solved with DNA computing [24–26], which demonstrated its superiority. Due to the nice features in parallel computing, massive storage, and the low energy consumption, many scholars in molecular biology, information science, and mathematics have studied DNA computing extensively. DNA coding is the basis of DNA computing. There are four kinds of nucleosides in the DNA double helix (A, C, G, T) or the RNA double helix (A, C, G, U). Two binary bits can present such four states. Here are eight mapping schemes presented in Table 1. Every three nucleosides constitute one genetic codon. With this map, any digital media can be encoded into pseudo-codon sequence.

Table 1 Different mapping ways according to DNA

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

2.2 Construction of pseudo-genetic codon table

The generational transfer of the genetic information follows the central dogma [27] in biology. In the process of formulating the protein, DNA sequences are firstly translated into mRNA sequences. Then every three nucleosides make up one codon. Finally, different amino acids are selected according the codon sequences and the genetic codon table (Table 2). There are 64 codons, but only 20 amino acids in the living organism. Therefore, there must be several codons mapped to the same amino acids. In fact, there are one, two, four, or six codons mapped to one amino acid, as in Table 2. It means that substitution of these codons with other codons in the same group will not change the corresponding protein and tissue. Such substitutions will not affect the biological characters of living organism.

Inspired by such a table, we can construct other grouping table similar to this standard genetic codon table. For example, a grouping table which maps 64 'codons' to 20 'amino acids' is generated like Table 3, denoted as pseudo-genetic codon table. This pseudo-codon table can be of various styles using the randomizing technique. The hyper-chaotic systems described in the next subsection will be used to acquire the randomness. Suppose there is a string of Q binary bits. Then there are $S (S = 2^Q)$ kinds of codes (or codons)

Table 2 Standard genetic codon table

1st base	2nd base								3rd base
	U		C		A		G		
U	UUU	(Phe/F)	UCU	(Ser/S)	UAU	(Tyr/Y)	UGU	(Cys/C)	U
	UUC		UCC		UAC		UGC		C
	UUA	(Leu/L)	UCA		UAA	Stop	UGA	Stop	A
	UUG		UCG		UAG		UGG	(Trp/W)	G
C	CUU		CCU	(Pro/P)	CAU	(His/H)	CGU	(Arg/R)	U
	CUC		CCC		CAC		CGC		C
	CUA		CCA		CAA	(Gln/Q)	CGA		A
	CUG		CCG		CAG		CGG		G
A	AUU	(Ile/I)	ACU	(Thr/T)	AAU	(Asn/N)	AGU	(Ser/S)	U
	AUC		ACC		AAC		AGC		C
	AUA		ACA		AAA	(Lys/K)	AGA	(Arg/R)	A
	AUG	(Met/M)	ACG		AAG		AGG		G
G	GUU	(Val/V)	GCU	(Ala/A)	GAU	(Asp/D)	GGU	(Gly/G)	U
	GUC		GCC		GAC		GGC		C
	GUA		GCA		GAA	(Glu/E)	GGA		A
	GUG		GCG		GAG		GGG		G

Table 3 Constructed pseudo-codon table

1st base	2nd base								3rd base
	0		1		2		3		
0	000	A	010	B	020	C	030	D	0
	001		011		021		031		1
	002		012		022		032		2
	003		013		023		033		3
1	100	H	110	G	120	F	130		0
	101		111		121		131	E	1
	102		112		122		132		2
	103		113		123		133		3
2	200		210		220		230	L	0
	201		211	I	221		231		1
	202	J	212		222	K	232	M	2
	203		213		223		233		3
3	300	N	310		320	O	330	P	0
	301		311		321		331		1
	302	Q	312	R	322	S	332	T	2
	303		313		323		333		3

in the codon set C. They are cataloged into N groups, where $N \leq S$. We can construct the pseudo-codon table with the following steps. Generate a random string Y containing N integer numbers in a range of $[t_1, t_2]$, where $0 \leq t_1 < t_2 \leq S$. Obviously, the sum of all the numbers in the string Y is S, that is $S = \sum_{i=1}^{N} Y(i)$.

Then, select $Y(i)$, $1 \leq i \leq N$ codes from set C as the ith group successively. Finally, rearrange these groups in the genetic codon table form.

We construct Table 3, when $Q = 6$, $S = 2^6 = 64$, $t_1 = 2$, $t_2 = 6$, and $N = 20$, every 2 binary bits are mapped to the four nucleosides—A, C, G, T. In Table 3, the different quanternary elements in set C are:

{000, 001, 002, 003, 100, 101,102,103,200, 201,
202, 203, 300, 301, 302, 303,010, 011,012,
013, 110, 111, 112,113, 210, 211, 212, 213,
310, 311, 312, 313,020, 021,022,023, 120,
121, 122, 123, 220, 221, 222, 223, 320, 321,
322, 323, 030, 031,032,033, 130, 131, 132, 133,
230, 231, 232, 233, 330, 331, 332, 333};

They are cataloged into $N = 20$ groups of {A,B,C,D,E, F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T}. The random string is $Y = [4,4,4,5,3,6,5,6,5,2,2,2,2,2,2,2,2,2,2,2]$.

Table 3 is quite similar to Table 2. They both map a set of 64 codons into 20 groups. With randomizing technique, we can construct various tables different from these two tables. When Q and N change, $Q = 5$, $N = 18$, for example, the constructed pseudo-codon table varies greatly.

2.3 Construction of arbitrary N-nary system

When the digital media is encoded into the pseudo-codon sequence, an arbitrary N-nary system can be generated according to the pseudo-genetic codon table, as described in Fig. 2. Suppose the arbitrary N-nary system has the bases $P = \{P_i, i = 1, 2, ..m\}$. Then, the numerical codes of different bases are integer numbers in a range $[0 \sim P_i - 1]$. In Fig. 2, $P = \{4, 3, 3, 5, 3, 6\}$ and their numerical codes are $0 \sim 3$, $0 \sim 2$, $0 \sim 2$, $0 \sim 4$, $0 \sim 2$,$0 \sim 5$, respectively. The weight of every base are $1, 6 \times 1, 2 \times 6 \times 1, 5 \times 2 \times 6 \times 1, 2 \times 5 \times 2 \times 6 \times 1, 2 \times 2 \times 5 \times 2 \times 6 \times 1$ respectively if the rightmost bits is selected as the lowest bit. The base P is arbitrary, it is determined by the pseudo-codon table and current encoded codon. And it is the number of codons that are in the same group with current encoded codon in the pseudo-codon table. Different selected groups are shown as A, J, J, D, R, H in Fig. 2, and the current encoded codons are 000,203,202,130,312,103 respectively.

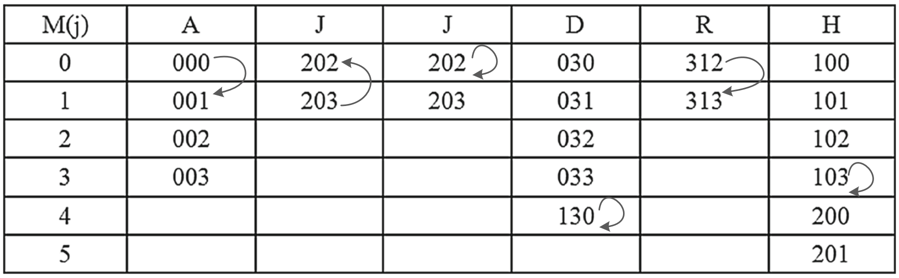
All information can be presented in binary sequences through sampling and coding from the perspective of computer science. Thus, suppose the additional information is a binary stream, it can be presented in decimal, octal, or hexadecimal form. Of course, it can also be presented into "N-nary" form, where N is arbitrary. A binary stream B can be presented in the arbitrary N-nary form by the following two steps. Transform B into decimal form D and then calculate the "N-nary" form M with equation $M(i) = \bmod\ (B/\Pi_{k=1}^{i-1} P_k, P_i), i \geq 1$, where P_i is the arbitrary base. As mentioned above, the base is the number of codons that are in the same group with current encoded codon in the pseudo-codon table.

2.4 Substitution process

After the original media and additional data are encoded, we can conduct the substitution process. With the encoded pseudo-codon sequence C, we can construct an arbitrary N-nary system, like in Fig. 2, according to the pseudo-codon table (Table 3). With the additional information M in arbitrary N-nary form, substitute the corresponding pseudo-codon by the $M(j)$th codon in the same group. Finally, rearrange the substituted pseudo-codon sequence into the new codon sequence and decode them into original form.

Here is an example for better understanding the substitution process. Assume the encoded codon sequence $C = \{000, 203, 202, 130, 312, 103\}$. In Fig. 2, the codons pointed by the arrowhead are the substituted codons. The arbitrary N-nary system, as in Fig. 2, can present a maximum integer decimal number $T = 4 \times 2 \times 2 \times 5 \times 2 \times 6 - 1 = 959$. The codon sequence after substitution is $C' = \{001, 202, 202, 130, 313, 103\}$. When in the same column, the encoded additional information in the arbitrary N-nary form regulates which codon is chosen to substitute the current codon. Suppose the additional information in binary form is $B = \{100101001\}$. Its decimal form is $D = 297$. Its arbitrary N-nary form is $M = \{1, 0, 0, 4, 1, 3\}$. The detailed process by module N is: $M(H) = \mathrm{mod}(297, 6)$, $M(R) = \mathrm{mod}(297/6, 2)$,$M(D) = \mathrm{mod}(297/6/2, 5)$, $M(J) = \mathrm{mod}(297/6/2/5, 2)$,$M(J) = \mathrm{mod}(297/6/2/5/2, 2)$, $M(A) = \mathrm{mod}(297/6/2/5/2/2, 4)$.

The basic process of the frame has been described in the above. The core of the coding and substitution frame is the substitution based on the constructed arbitrary



M(j)	A	J	J	D	R	H
0	000	202	202	030	312	100
1	001	203	203	031	313	101
2	002			032		102
3	003			033		103
4				130		200
5						201

Fig. 2 Substitution process

N-nary system. It is determined by the original media and the constructed pseudo-codon table. The pseudo-codon table is constructed with the hyper-chaotic systems according to different need, which offers large key space. The original media will be encoded with different strategies in different applications. For example, when in the steganography, the least significant bits of the pixels in the image will be selected to decrease the distortion. However, when in the encryption, usually the most significant bits of the pixels in the image will be selected to improve the efficiency. Different substitution strategies will be adopted in encryption and steganography too. When in the encryption algorithm, the controlling bits in the arbitrary N-nary form are generated randomly by the hyper-chaotic system. However, when in the steganography algorithm, the controlling bits are the hidden information in the arbitrary N-nary form.

Just as described in Fig. 2, additional information is presented by the substitution. At the same time, the substitution causes few changes. It is pleasing for steganography. In the receiving end, the receiver can acquire the hidden information just by reading the arbitrary N-nary numbers and translating them back into original form. Of course, they must firstly get the pseudo-codon table that is constructed randomly and secretly. However, if codons in the same group in the pseudo-codon table vary greatly, substitution of codons may cause great changes in the original media, which will encrypt the original media. At that time, random number generators, such as the chaotic systems described in the followings Sect. 2.5, can generate the additional information to enhance the security.

2.5 Randomness offered by the hyper-chaotic system

As mentioned above, the randomness in the frame is offered by the hyper-chaotic system. Randomness will be used in constructing the pseudo-codon table, deciding the arbitrary bases in the arbitrary N-nary system, and deciding the codons in the substitution. The hyper-chaotic system [28] achieves better performance because it increased the dimensions of the nonlinear system compared with traditional chaotic systems:

$$\begin{cases} \dot{y}_1 = a(-y_1 + y_2), \\ \dot{y}_2 = dy_1 + cy_2 - y_1 y_3 - y_4, \\ \dot{y}_3 = y_1 y_2 - by_3, \\ \dot{y}_4 = y_1 + k \end{cases} \tag{1}$$

in which a, b, c, d and k are constant parameters of the nonlinear system. The system is hyper-chaotic when $a = 36, b = 3, c = 28, d = -16$ and $-0.7 \le k \le 0.7$. The additional information E can be generated through steps as follows. Iterate the hyper-chaotic system for N_0 times by Runge-Kutta algorithm to get four discrete states sequences y_k, $(k = 1, 2, 3, 4)$. Then construct the random integer sequences x_k, $(k = 1, 2, 3, 4)$ with y_k:

$$x_k(i) = \bmod\ ((\text{abs}(y_k(i)) - \text{floor}(\text{abs}(y_k(i)))) \times 10^t, D(i)) \tag{2}$$

where $x_k(i)$ is the ith value of sequence x_k, abs(x) represents the absolute value of x, and floor(x) returns the nearest integer less than or equal to x. Finally, construct the random sequence $E(j)$ with x_k, $(k = 1, 2, 3, 4)$: $E(j) = x_k(i)$, where $k = j - (i - 1) \times 4$. $D(i)$ is the intensity that regulates every random integer number in the range of $[0, D(i) - 1]$. Random sequence in arbitrary N-nary form can be generated with different bases $D(i)$, $i \ge 1$.

3 Implementation

The proposed frame changes some features while keeping other features of the original media, such as length, unchanged. It links the original media and the additional information. It is also a kind of map in some degree. Therefore, it can be used in the design of steganography and the encryption schemes of digital media in various formats. However, the purposes of steganography and encryption are quite different and even opposite sometimes. The main purpose of steganography is concealing the existence of additional data and keeping features of original media unchanged as much as possible. To the contrary, in the case of encryption, the main purpose is to make the encrypted media different from original media as much as possible. Considering such differences, there are many detailed differences in the encoding and substituting process between steganography and encryption. The implementation in steganography and encryption are detailed in the following subsections respectively.

3.1 Schemes for steganography

Through the substitution, we can hide some additional information into the original media. As long as the original media can be encoded, there is the possibility in steganography. The implementation of steganography in the image is presented in the followings.

3.1.1 Steganography in the image

In steganography schemes based on images, images act as the cover media. They are the corresponding original media in the coding and substitution frame. All things needed are to preprocess the cover image to accommodate the proposed frame. Suppose the two-dimensional image matrix I with the size $M \times N$ act as the cover media, and the additional information in the binary form is B.

- Steganography

  a. Scan the cover image I to get an one-dimensional sequence S with size $(1, M \times N)$, the scanning strategy can be the simple 'from left to right and from top to bottom' method, or other ways such as the zigzag way or inverse S way;
  b. Transform every pixel $S(i)$, $(i = 1, 2, \cdots, M \times N)$ of decimal integers into binary sequence $S(i, 1:8)$;
  c. Select the last k, $(1 \leq k \leq 8)$ bits $S(i, (8-k):8)$ in every $S(i, 1:8)$, and encode them into the pseudo-codon sequence C according to coding method proposed in Sect. 2.1;
  d. Generate the random string Y with the hyper-chaotic system, and construct the pseudo-codon table with the pseudo-codon sequence according to the hyper-chaotic random string Y;
  e. Transform the additional information B into arbitrary N-nary form B' with method proposed in Sect. 2.3;
  f. Substitute pseudo-codon sequence of the cover image with codons in the same group according to the arbitrary N-nary additional information B', just like in Sect. 2.4.
  g. Translate the substituted pseudo-codon sequence back into image form.

Parameters for coding the cover media and the parameters of the hyper-chaotic systems for randomly generating the pseudo-codon table are encoded as the keys for data extraction.

- Data extraction

  a. Transform the received image into codon sequence with the coding key just as in the steganography process;
  b. Generate the pseudo-codon table as Table 3 with the hyper-chaotic random pseudo-codon table key;
  c. Construct the arbitrary N-nary system table like Fig. 2 with the pseudo-codon sequence according to the pseudo-codon table;
  d. Read the N-nary information M by finding the corresponding position of the current pseudo-codon;
  e. Transform M into a big decimal integer D and then transform D into binary form B.

The example in Fig. 2 in Sect. 2.4 can just act as the demo for the steganography scheme. The encoded codon sequence is $C = \{000, 203, 202, 130, 312, 103\}$. The substituted pseudo-codon sequence is $C' = \{001, 202, 202, 130, 313, 103\}$. There are very few changes in the codon sequence after steganography.

The constructed pseudo-codon table which is chaotic random offers better security. However, if the pseudo-codon table for grouping the encoded codons is completely randomized, the distortion introduced after steganography will be awful for a high payload. It is annoying in steganography, but it is pleasing in the design of encryption schemes. The implementation of encryption with a more randomized pseudo-genetic codon table is described in the next subsection.

3.2 Schemes for image encryption

As mentioned above, if the pseudo-codon table is completely randomized, the codon sequence after substitution will vary greatly. Another pseudo-codon table can be generated through permutation of the codons, as depicted in Table 4.

The encoding method for original media is the same as encoding method for cover media in the steganography scheme. However, the substitution strategy must be different. It is inconvenient to encode so much additional information as the key for encryption and decryption. Therefore, we generate the additional information by the chaotic system. The initial values and the parameters of the chaotic system can be encoded as the keys.

With method proposed in Sect. 2.5, the additional information can be determined by several chaotic parameters, the encryption and decryption can be explained as follows.

- Encryption

  a. Encode original media into pseudo-codon sequence, denoted by C, with the method proposed in Sect. 2.1;
  b. Permute codons and construct the permuted pseudo-codon table like Table 4;
  c. Generate the arbitrary N-nary additional information A using the hyper-chaotic system (1) with method proposed above;
  d. Substitute the pseudo-genetic codon with $C'(i) = M(i, j')$, where $j' = \text{mod}\ (j + A(i), D(i))$ and $D(i)$ is the base of the current bit in the arbitrary N-nary system;
  e. Decode the substituted pseudo-codon sequence C' into its original form.

Both the encoding of the original media and the construction of permuted pseudo-codon table contribute to the encryption keys. There are eight mapping ways in the encoding of original media as presented in Table 1. There binary bits, denoted by $[opq], o, p, q \in \{0, 1\}$ can be utilized as the encoding key, denoted by key1. The initial values $[y_1', y_2', y_3', y_4']$ and parameters $a_3 = 36, b_3 = 3, c_3 = 28, d_3 = -16, k_3 = z'$, where $-0.7 \leq z' \leq 0.7$ of the hyper-chaotic system for the generating additional information are encoded as another key, denoted by key2. The seed for generating the random string Y and permutation strategy in the construction of permuted pseudo-genetic codon table are encoded as the third key, denoted by key3. In fact, the hyper-chaotic system can also be utilized in the construction of the random string Y, which will enlarge the key space greatly. Notice that, permutation in the b) step is quite necessary for the whole encryption scheme.

- Decryption

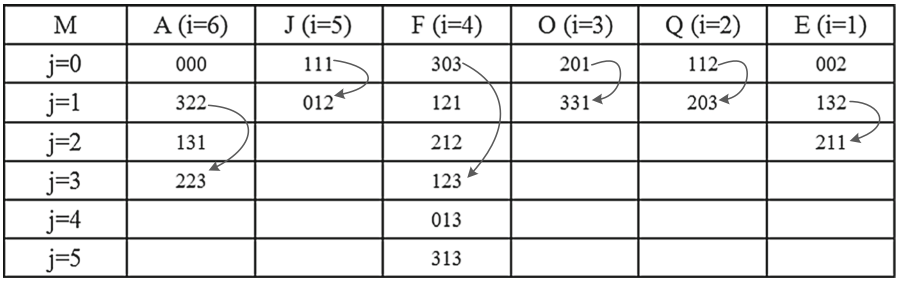
  a. Encode the encrypted media into pseudo-codon sequence C' with key1;
  b. Permute codons and generate the random string Y with key3, then construct the permuted pseudo-codon table;
  c. Generate the arbitrary N-nary system with base D according to C' and the permuted pseudo-codon table;
  d. Generate additional information A in the arbitrary N-nary form with key2 and the hyper-chaotic system (1), the bases $D(i)$ of the arbitrary N-nary system is incorporated;
  e. Substitute the pseudo-codon $C'(i)$ to decrypt the pseudo-codons with $C(i) = M(i, j)$, where $j = \text{mod}\ (j' - A(i), D(i))$;
  f. Decode the substituted pseudo-codon sequence C into its original form.

In the image encryption, pixels of the image are decoded into binary sequence. Then they are encoded into pseudo-codon sequences. More effectively, some MSBs (Most Significant Bits) of every pixel can be selected in the coding process.

Here is a demo for better understanding the encryption process. Suppose the original media in the binary form is $B = \{111110, 010101, 110011, 100001, 010110, 011110\}$. Then, its pseudo-codon sequence is $C = \{332,111,303,201,112,132\}$. The bases of the arbitrary N-nary system are $D(i), i = 1, \ldots, 6$. The pseudo-codon sequence after substitution is $C' = \{223,012,013,331,203,211\}$. The additional information in arbitrary N-nary generated by the hyper-chaotic system is $A = \{2,1,3,1,1,1\}$. The whole process encrypt B into B'. Figure 3 demonstrates the substitution process. Table 5 offers parameters appeared in the substitution.

Table 4 Permuted pseudo-codon table

1st base	2nd base								3rd base
	0		1		2		3		
0	000	A	010	B	122	C	220	D	0
	322		202		021		031		1
	131		033		300		311		2
	223		113		231		120		3
1	100	H	110	G	303	F	130		0
	321		221		121		002	E	1
	233		302		212		132		2
	020		023		123		211		3
2	200		210		013		230	L	0
	330		032	I	313		022		1
	111	J	101		222	K	232	M	2
	012		030		003		312		3
3	011	N	310		320	O	201	P	0
	301		133		213		331		1
	112	Q	332	R	102	S	001	T	2
	203		103		323		333		3



M	A (i=6)	J (i=5)	F (i=4)	O (i=3)	Q (i=2)	E (i=1)
j=0	000	111	303	201	112	002
j=1	322	012	121	331	203	132
j=2	131		212			211
j=3	223		123			
j=4			013			
j=5			313			

Fig. 3 Substitution for encryption

Table 5 Different sequences in the encryption

B	111110	010101	110011	100001	010110	011110
C	332	111	303	201	112	132
C'	223	012	013	331	203	211
B'	101011	000110	000111	111101	100011	100101
D	4	2	6	2	2	3
A	2	1	3	1	1	1

4 Experiments and analysis

We have tested different digital media in the experiments period. Images with size 512 × 512 from USC-SIPI image database and miscellaneous gray-level images are selected for the demonstration both in the steganography and in the encryption. All experiments are performed on the MATLAB 2012a platform running on a personal computer with CPU of AMD Phenom (tm) II X4 810 Processor 2.6 GHz, memory of 4 GB, and operating system of Windows 7 × 64 Ultimate Edition.

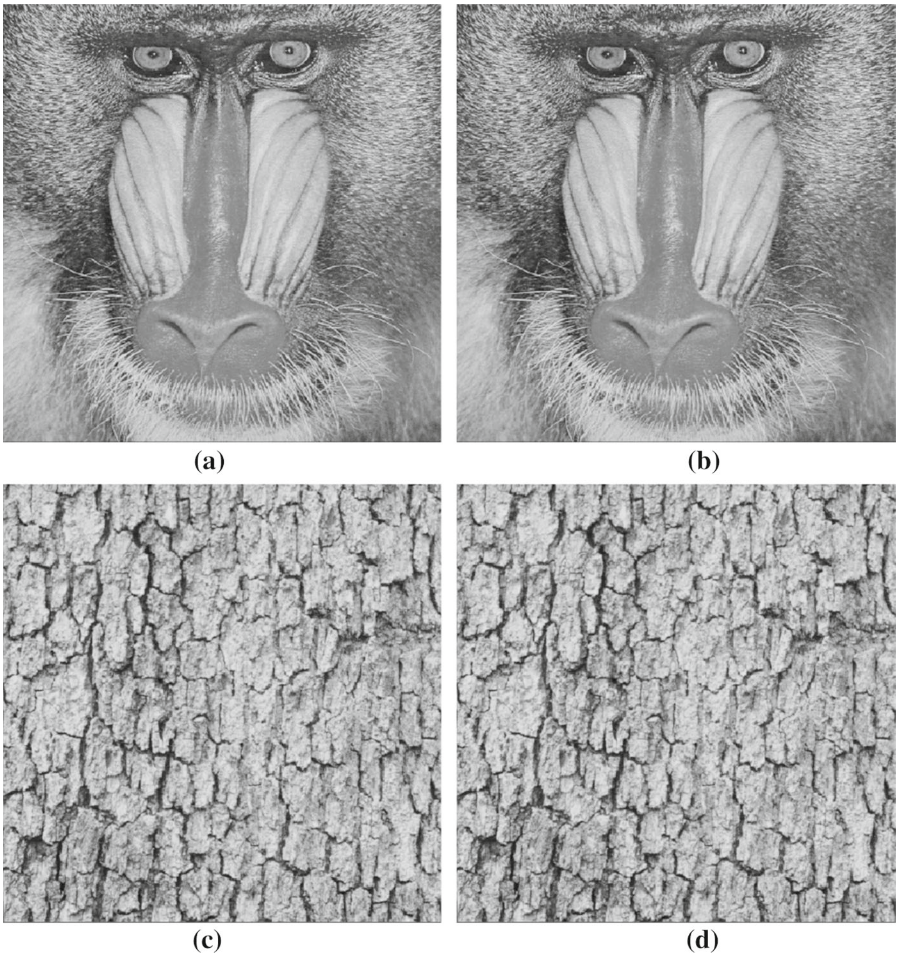


Fig. 4 Image ‘Baboon’ and texture image after embedding. **a** 6LSB embedding with PSNR = 43.4713, **b** 2LSB embedding with PSNR = 50.6073, **c** 6LSB embedding with PSNR = 44.1440, **d** 2LSB embedding with PSNR = 50.1622

Table 6 Results of 2LSBs embedding according to Table 3

Image	Airplane	Baboon	Boat	Lena	Peppers	Tiffany
ER(bpp)	0.5883	0.5898	0.5889	0.5898	0.5913	0.5882
PSNR(dB)	49.8712	50.6073	50.2333	49.4485	49.9727	49.5426

4.1 Steganography schemes

4.1.1 Experiments and comparisons

Random binary bits act as the additional information in the experiments. Images ‘Baboon’ and a texture image after steganography are presented in Fig. 4. The last two LSBs and the last six LSBs are selected to be encoded into pseudo-codon sequence for steganography, respectively. The embedding rates of the sub-image (a) and (b) are 1.7278 bpp (bit per pixel) and 0.5898 bpp, respectively, while the embedding rate of sub-image (c) and (d) are 1.7556 bpp and 0.5930 bpp, respectively.

The payload varies as the cover media changes, because different images are encoded into different pseudo-codon sequences. The substitution process may be different very much. Besides, the payload varies when the randomly generated pseudo-codon table varies. When the pseudo-codon table is just Table 3, the embedding rates and corresponding PSNR of different standard images are demonstrated in Table 6.

Compared with those LSB based steganography schemes [1,29–31], the proposed scheme offers more

Table 7 Comparisons of traditional 2LSBs method and 2LSBs steganography by us

Image	Airplane	Baboon	Boat	Lena	Peppers	Tiffany
ER(bpp)	0.3	0.3	0.3	0.3	0.3	0.3
PSNR(dB)						
Traditional	52.3878	52.4339	52.2734	52.3576	52.4145	52.4184
Ours	52.7312	52.9913	52.7304	52.7862	53.0851	53.1859

security, and a higher payload sometimes. Through the random generation of pseudo-codon table, the encryption of secure data is done in accompany with the embedding process. One can never know the arbitrary N-nary system without the pseudo-codon table, which is randomly constructed. The substitution of pseudo-codons causes random changes to the cover media, which is hardly detected. Besides, additional information is often hidden without changing the original pseudo-codons, as in Fig. 2. Therefore, the corresponding pixels keep the same in the image. It causes less distortion compared with traditional LSB methods when the same amount of data is hidden, as shown in Table 7.

Higher payload and lower distortion has been achieved compared with those traditional steganography schemes such as difference expansion-based methods [3,4] and histogram-based methods [5,32], whose embedding rate is less than 1 bpp. Besides, the texture of the image does not affect the payload of the steganography. The DNA steganography schemes based on complementary rule [33,34] hide data through the substitution of the nucleosides with their complementary nucleosides. It needs reference DNA sequences in the data extraction process. However, scheme proposed by us achieves blind extraction that is very important in steganography. Besides, in [34], the look-up table is needed in the data extraction. It needs large quantity additional information in the extraction process.

4.1.2 Analysis

If the pseudo-codon table is just like a special codon table, as depicted in Table 10, the randomness is eliminated. It achieves a trade-off between the payload (embedding capacity) and the distortion in the steganography. If the six LSBs of every pixel are encoded into pseudo-codons, the steganography scheme according to Table 8 becomes into the basic LSB steganography. It substitutes the last two bits of every pixel to present the additional information. Therefore, traditional LSB steganography is a special case of the proposed scheme. Obviously, the embedding rate is 0.67 bpn (bit per nucleoside) if codons are grouped according to Table 8. Moreover, the corresponding embedding rate of steganography in images comes to 2 bpp.

If the genetic codon table is just the real one, as depicted in Table 2, the core of proposed scheme can be utilized in the DNA steganography that similar to scheme proposed by Tai [35]. The proposed scheme can be used for any digital cover media as long as it can be encoded, DNA sequence is of course one of them. In this scheme, information is embedded into codons through a substitution of codons that can be mapped to the same amino acids. Those substituted codons are utilized for the authentication of plant variety rights. The steganography scheme in DNA proposed by Tai is used in marking plants for authentication. Therefore, it depends on the real genetic codon table, which is fixed and public. However, scheme proposed in this paper depends on the pseudo-genetic codon table that is generated randomly. The grouping strategy is secret and can be generated one-time pad. It is obvious that scheme proposed by us not only has all the advantages of Tai' scheme but also is much safer.

4.2 Image encryption

4.2.1 Experiments and comparisons

Image 'Lena' with size 512 × 512 is encrypted and decrypted to demonstrate the results. Randomly select three bits from the four MSBs of every pixel in the image and encode them for the encryption. They are encoded and substituted. The original image, image after encryption, image decrypted with right keys, and image decrypted with wrong keys are presented in Fig. 5, respectively. The encrypted image is of ran-

Table 8 Special pseudo-codon table

1st base	2nd base								3rd base
	0		1		2		3		
0	000	A	010	B	020	C	030	D	0
	001		011		021		031		1
	002		012		022		032		2
	003		013		023		033		3
1	100	H	110	G	120	F	130	E	0
	101		111		121		131		1
	102		112		122		132		2
	103		113		123		133		3
2	200	I	210	J	220	K	230	L	0
	201		211		221		231		1
	202		212		222		232		2
	203		213		223		233		3
3	300	M	310	N	320	O	330	P	0
	301		311		321		331		1
	302		312		322		332		2
	303		313		323		333		3

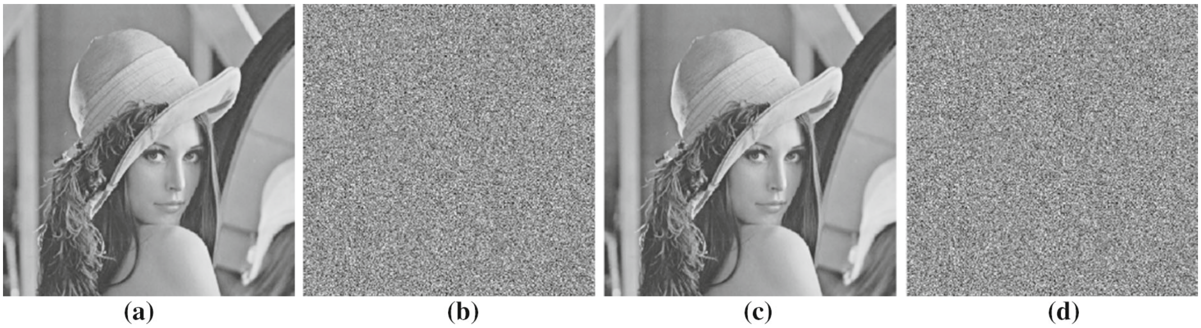

(a) (b) (c) (d)

Fig. 5 Image ‘Lena’ and its encryption and decryption. **a** Original image, **b** image after encryption, **c** right decryption, **d** wrong decryption

dom noise-like distribution. The encryption keys are: key1 = [101], key2: the initial value of the hyper-chaotic system [11, 2, 8, 1], the parameters of the hyper-chaotic system $a_2 = 36, b_2 = 3, c_2 = 28, d_2 = -16, k_2 = 0.2$. In the wrong decryption, the initial values of the hyper-chaotic system in key2 are slightly changed from [11, 2, 8, 1] to [11, 2, 8.00000001, 1]. The hyper-chaotic system is very sensitive to the initial values and parameters, which guarantees the keys sensitivity in the encryption scheme. Histograms of original “Lena” image and of the image after encryption are presented in Fig. 6. The encryption scheme destroys the statistical features in the histogram.

All these steps imposed on the image in the encryption have permuted the selected three MSBs of every pixel sufficiently. It is a bit level encryption scheme based on the complicated permutation and substitution in some degree. The hyper-chaotic systems are incorporated, which is much safer compared with other existing bit level permutation schemes [14,17,18].

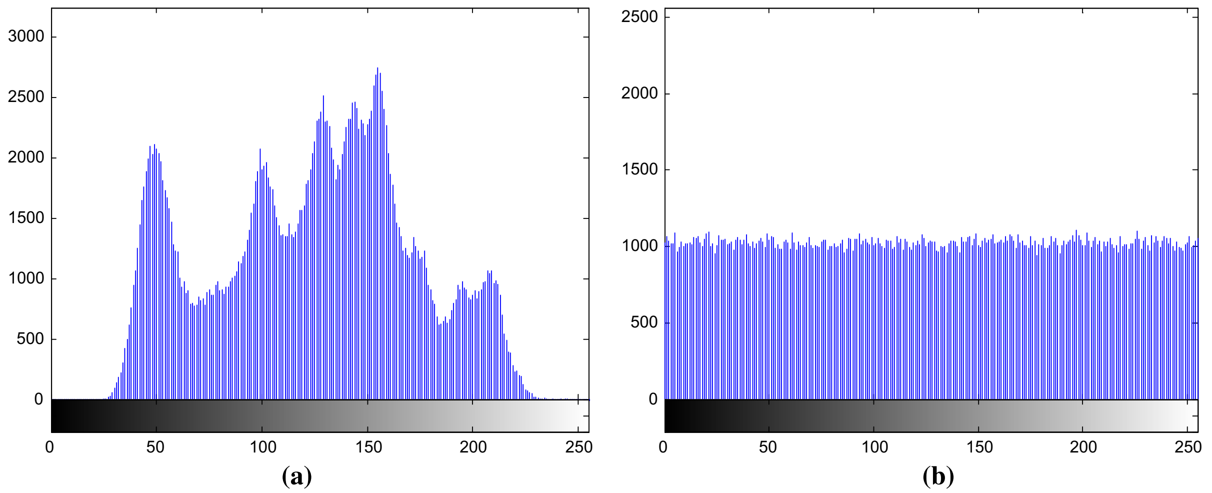


Fig. 6 Histograms of Image "Lena". **a** Histogram of original "Lena", **b** histogram after encryption

Due to the hyper-chaotic systems are used in the encryption, the algorithm offers large key space. There are three keys key1, key2, key3 in the whole algorithm. The first one key1 has 8 different selections. Another two keys key2 and key3 offer the hyper-chaotic random rational numbers. The secret key comprises random fractional numbers with t bits in every hyper-chaotic system, and the hyper-chaotic system has been used for two times. Therefore, the key space can be roughly calculated without considering extra parameters: $8 \times ((10^t)^4)^2$, where $t = 13, 14, 15$ is always selected. It is large enough to resist brute-force attack.

The information entropy defined in theory of information measures the uncertainty of a random variable. When used in the image encryption, it measures the distribution of pixels in one image. Greater information entropy represents higher uniformity. The ideal value of information entropy is approaching to 8 for an encrypted gray value image. It is defined as follows:

$$H(m) = \sum_{i=1}^{L} P(m_i) \log_2 \frac{1}{P(m_i)}, \tag{3}$$

where m_i is the gray value of the ith pixel, $P(m_i)$ represents the occurrence probability of m_i. There are correlations between adjacent pixels in nature images, which make the nature images look comfortable. A basic and important task for image encryption is to destroy such correlation. The correlation between two adjacent pixels is calculated by the following formulas:

$$E(x) = \frac{1}{N} \sum_{i=1}^{N} x_i, \tag{4}$$

$$D(x) = \frac{1}{N} \sum_{i=1}^{N} (x_i - E(x))^2 \tag{5}$$

$$\mathrm{cov}(x, y) = \frac{1}{N} \sum_{i=1}^{N} E(x - E(x)(y - E(y))) \tag{6}$$

$$r_{xy} = \frac{\mathrm{cov}(x, y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{7}$$

We randomly select 4096 pairs of adjacent horizontal pixels, adjacent vertical pixels and adjacent diagonally pixels to test the correlations of the same image before encryption and after encryption. Fig. 7 presents the correlation of image 'Lena' from Fig. 5. There are strong correlations between adjacent pixels in the natural image 'Lena', as depicted in Fig. 7a–c. However, such correlations disappear in the encrypted 'Lena' image, as in Fig. 7d–f.

The information entropy and correlation coefficients of image 'Lena' before and after encryption are demonstrated in Table 9.

Clearly, the information entropy after encryption is quite close to the ideal value 8 after encryption. The correlation coefficients of original image are close to 1, which demonstrate the similarity features of natural images. However, correlation coefficients after encryption are very close to zero, which means these similarities between adjacent pixels are destroyed. Comparisons of information entropy with other schemes are

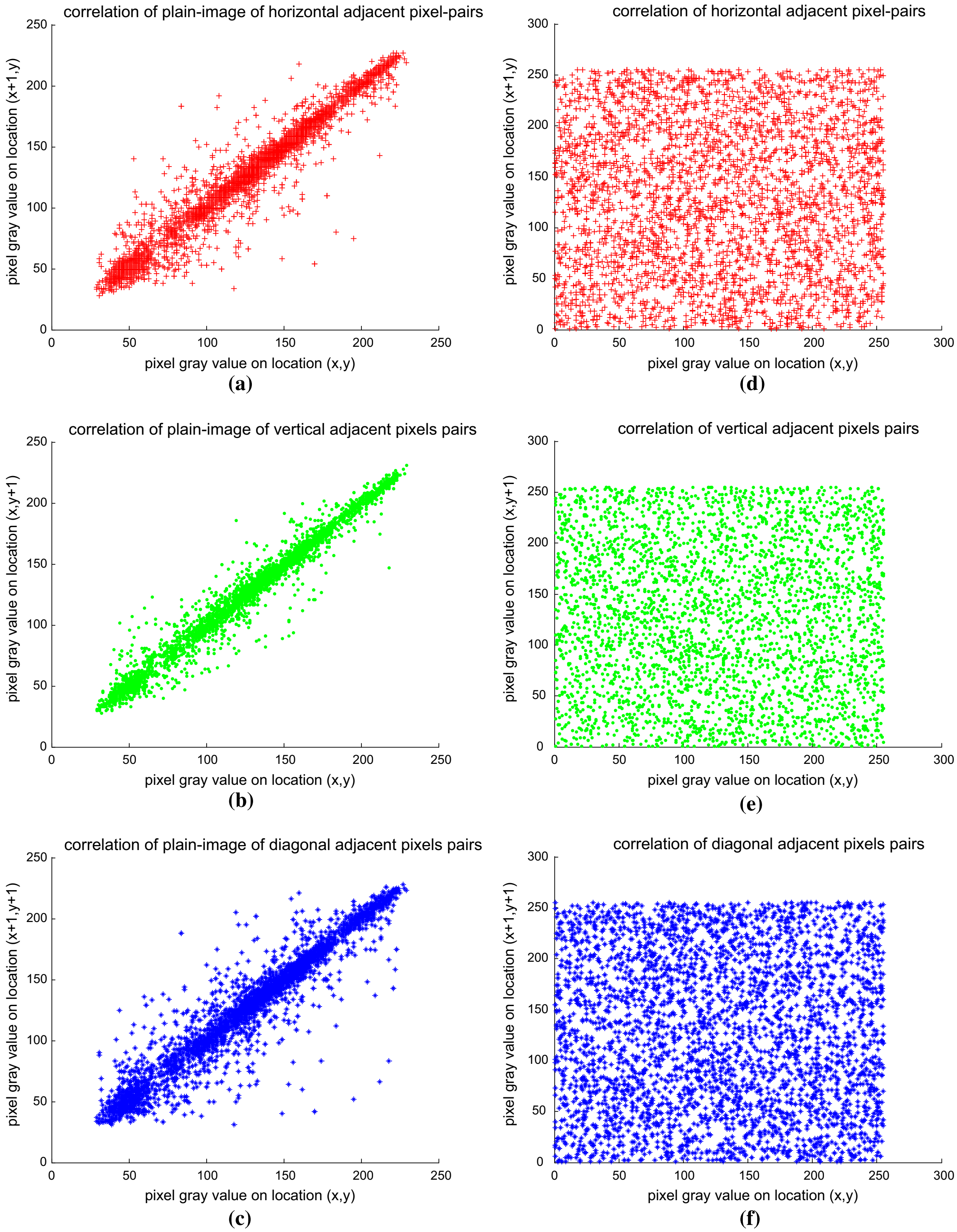


Fig. 7 Correlations of two adjacent pixels in the plain-image and in the cipher-image of ‘Lena’. **a** Adjacent horizontal pixels (original), **b** adjacent vertical pixels (original), **c** adjacent diagonal pixels (original), **d** adjacent horizontal pixels (encrypted), **e** adjacent vertical pixels (encrypted), **f** adjacent diagonal pixels (encrypted)

Table 9 Encryption results of "Lena" image with size 512 × 512

Lena image	Information entropy	Correlation coefficients		
		Horizontal	Vertical	Diagonal
Original	7.4450613278190	0.9928	0.9631	0.9729
Encrypted	7.9969682810312	0.0103	−0.0061	0.084

Table 10 Information entropy comparisons with existing schemes of 'Lena' image

Schemes	256 × 256			128 × 128		
Entropy	Plain-image	Cipher-image	Liu's [20]	Plain-image	Cipher-image	Liu's [21]
Entropy	7.5683	7.992	7.9874	7.2099	7.9881	7.9877

Table 11 Comparisons of UACI and NPCR of image 'Lena' with size 256 × 256

Schemes	Ours	Lin's [16]	Liu's [20]	Pareek's [36]
UACI	0.3289	0.3334	0.2814	0.3179
NPCR	0.9960	0.9368	0.9960	N/A

presented in Table 10. Compared with other bit-level encryption schemes based on DNA [20,21], the proposed scheme achieves better results.

The NPCR (Number of Pixels Change Rate) and UACI (Unified Average Changing Intensity) are two parameters to measure the sensitivity of an encryption scheme. The NPCR means the percentage of different pixels between two encrypted images, while the UACI calculates the average difference between the pixels of two encrypted images:

$$C(i,j) = \begin{cases} 0, & \text{if} \quad C_1(i,j) = C_2(i,j) \\ 1, & \text{if} \quad C_1(i,j) \neq C_2(i,j) \end{cases}, \tag{8}$$

$$\text{NPCR} = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} C(i,j)}{M \times N} \times 100\,\%, \tag{9}$$

$$\text{UACI} = \frac{1}{M \times N}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{|C_1(i,j) - C_2(i,j)|}{255} \times 100\,\%, \tag{10}$$

where C_1 and C_2 represent pixels of two encrypted images with size $M \times N$.

Randomly select one pixel in the image and substitute it with another random gray value. The average NPCR and UACI of 50 images are 99.6021 and 33.0994 %, respectively. They are very close to the ideal value, which means that the encryption is hardly decrypted by differential attacks. The comparisons of UACI and NPCR of image 'Lena' with other encryption schemes are presented in Table 11. Due to the one-time pad design in the encryption and the sensitivity of the hyper-chaotic system, better performances have been achieved. The initial values of the hyper-chaotic system can be generated from the original media according to a simple hash function. Compared with bit-level encryption scheme [16], DNA based encryption scheme [20], and traditional diffusion-substitution scheme [36], it achieves better UACI and/or NPCR because of the bit-level chaotic substitution.

4.2.2 Analysis

The proposed encryption scheme is a chaotic bit-level scheme. It has the advantages of other chaotic and/or bit-level schemes. Besides, it takes only some MSBs of every pixel in the image, which means no more than half of the binary bits are involved in the substitution. Moreover, the basic operation to achieve the encryption is substitution rather than the traditional time-consuming computing, such as exclusive OR operation. Therefore, it is more efficient. The one-time pad feature offered by the scheme is also necessary for a novel encryption scheme.

The coding and substitution frame offers some new features both in the steganography and in the encryption, which both contribute the secure communication. It offers more security, high payload with low distortion to steganography by the random substitution. Moreover, it offers bit level, chaotic, one-time-pad, and timesaving features to the encryption. Different from existing steganography and encryption schemes designed for special media such as images, audios, videos and so on, the proposed scheme is based on binary bits coding. Therefore, other kinds of digital media can also adopt this frame to achieve steganography and encryption concerning its own features.

5 Conclusions

Inspired by central dogma in molecular biology, this paper designed a coding and substitution frame. Based on the proposed frame, implementations of steganography and encryption are presented, both of which have demonstrates their superiority compared with traditional secure communication algorithms. The basic process of the frame is coding and substitution according to some rules, which is simple but useful. Besides, the hyper-chaotic random number generator and the construction of arbitrary N-nary system proposed in this paper can also be used in many other areas in signal processing.

Acknowledgments The work was partly supported by the Key Program of National Science Fund of Tianjin, China (Grant NO. 11JCZDJC16000).

References

1. van Schyndel, R.G., Tirkel, A.Z., Osborne, C.F.: A digital watermark. In: Proceedings of the IEEE International Conference on Image Processing (1994), pp. 86–90
2. Fridrich, J., Goljan, M., Du, R.: Invertible authentication watermark for JPEG images. In: IEEE Proceedings of the International Conference on Information Technology: Coding and Computing (2001), pp. 223–227
3. Tian, J.: Reversible data embedding using a difference expansion. IEEE Trans. Circuits Syst. Video Technol. **13**(8), 890–896 (2003)
4. Alattar, A.M.: Reversible watermark using the difference expansion of a generalized integer transform. IEEE Trans. Image Process. **13**(8), 1147–1156 (2004)
5. Ni, Z., Shi, Y.-Q., Ansari, N., Su, W.: Reversible data hiding. IEEE Trans. Circuits Syst. Video Technol. **16**(3), 354–362 (2006)
6. Anees, A., Siddiqui, A.M., Ahmed, J., Hussain, I.: A technique for digital steganography using chaotic maps. Nonlinear Dyn. **75**(4), 807–816 (2014). doi:10.1007/s11071-013-1105-3
7. Shannon, C.E.: Communication theory of secrecy systems. Bell Syst. Tech. J. **28**(4), 656–715 (1949)
8. Gao, T., Chen, Z.: A new image encryption algorithm based on hyper-chaos. Phys. Lett. A **372**(4), 394–400 (2008)
9. Huang, C., Nien, H.: Multi chaotic systems based pixel shuffle for image encryption. Opt. Commun. **282**(11), 2123–2127 (2009)
10. Yoon, J.W., Kim, H.: An image encryption scheme with a pseudorandom permutation based on chaotic maps. Commun. Nonlinear Scie. Numer. Simul. **15**(12), 3998–4006 (2010)
11. Huang, X.: Image encryption algorithm using chaotic Chebyshev generator. Nonlinear Dyn. **67**(4), 2411–2417 (2012)
12. Ye, G., Wong, K.-W.: An efficient chaotic image encryption algorithm based on a generalized Arnold map. Nonlinear Dyn. **69**(4), 2079–2087 (2012)
13. Wang, X.-Y., Wang, T.: A novel algorithm for image encryption based on couple Chaotic systems. Int. J. Mod. Phys. B **26**(30), 1250175(1250171)-1250175(1250179) (2012)
14. Ye, G.: Image scrambling encryption algorithm of pixel bit based on chaos map. Pattern Recognit. Lett. **31**(5), 347–354 (2010)
15. Zhu, Z.-L., Zhang, W., Wong, K.-W., Yu, H.: A chaos-based symmetric image encryption scheme using a bit-level permutation. Inf. Sci. **181**(6), 1171–1186 (2011)
16. Lin, T., Xingyuan, W.: A bit-level image encryption algorithm based on spatiotemporal chaotic system and self-adaptive. Opt. Commun. **285**(20), 4048–4054 (2012)
17. Zhang, W., Wong, K.-W., Yu, H., Zhu, Z.-I.: An image encryption scheme using lightweight bit-level confusion and cascade cross circular diffusion. Opt. Commun. **285**(9), 2343–2354 (2012)
18. Zhang, W., Wong, K.-W., Yu, H., Zhu, Z.-I.: A symmetric color image encryption algorithm using the intrinsic features of bit distributions. Commun. Nonlinear Sci. Numer. Simul. **18**(3), 584–600 (2012)
19. A, A.V.I.A.: Ergodic problems of classical mechanics. Math. Phys. Monogr. Ser. (1968)
20. Liu, H., Wang, X.: Image encryption using DNA complementary rule and chaotic maps. Appl. Soft Comput. **12**(5), 1457–1466 (2012)
21. Liu, L., Zhang, Q., Wei, X.: A RGB image encryption algorithm based on DNA encoding and chaos map. Comput. Electr. Eng. **38**(5), 1240–1248 (2012)
22. Zhao, J., Wang, S., Chang, Y., Li, X.: A novel image encryption scheme based on an improper fractional-order chaotic system. Nonlinear Dyn. **80**(4), 1721–1729 (2015). doi:10.1007/s11071-015-1911-x
23. Adleman, L.M.: Molecular computation of solutions to combinatorial problems. Science **266**, 1021 (1994)
24. Lipton, R.J.: DNA solution of hard computational problems. Science **268**(5210), 542–545 (1995)
25. Ouyang, Q., Kaplan, P.D., Liu, S., Libchaber, A.: DNA solution of the maximal clique problem. Science **278**(5337), 446–449 (1997)
26. Liu, Q., Wang, L., Frutos, A.G., Condon, A.E., Corn, R.M., Smith, L.M.: DNA computing on surfaces. Nature **403**(6766), 175–179 (2000)
27. Crick, F.: Central dogma of molecular biology. Nature **227**(5258), 561–563 (1970)
28. Gao, T., Chen, Z., Yuan, Z., Chen, G.: A hyperchaos generated from Chen's system. Int. J. Mod. Phys. C **17**(04), 471–478 (2006)

29. Chan, C.-K., Cheng, L.-M.: Hiding data in images by simple LSB substitution. Patt. Recogn. **37**(3), 469–474 (2004)
30. Yang, C.-H., Weng, C.-Y., Wang, S.-J., Sun, H.-M.: Adaptive data hiding in edge areas of images with spatial LSB domain systems. IEEE Trans. Inf. Forensics Secur. **3**(3), 488–497 (2008)
31. Liao, X., Wen, Q.-Y., Zhang, J.: A steganographic method for digital images with four-pixel differencing and modified LSB substitution. J. Vis. Commun. Image Represent. **22**(1), 1–8 (2011)
32. Tulpan, D., Regoui, C., Durand, G., Belliveau, L., Léger, S.: HyDEn: a hybrid steganocryptographic approach for data encryption using randomized error-correcting DNA codes. BioMed Res. Int. **2013**, 1–11 (2013)
33. Shiu, H., Ng, K.-L., Fang, J.-F., Lee, R.C., Huang, C.-H.: Data hiding methods based upon DNA sequences. Inf. Sci. **180**(11), 2196–2208 (2010)
34. Taur, J.-S., Lin, H.-Y., Lee, H.-L., Tao, C.-W.: Data hiding in DNA sequences based on table lookup substitution. Int. J. Innov. Comput. Inf. Control **8**(10A), 6585–6598 (2012)
35. Tai, W.-L., Wang, C.N., Sheu, P.C.Y., Tsai, J.J.P.: Data hiding in DNA for authentication of plant variety rights. J. Electron. **11**(1), 38–43 (2013)
36. Pareek, N.K., Patidar, V., Sud, K.K.: Diffusion-substitution based gray image encryption scheme. Digital Signal Process. **23**(3), 894–901 (2013)