A Color Image Encryption Scheme Based on Arnold Scrambling and Quantum Chaotic

Hui Liu, Cong Jin
(Corresponding author: Cong Jin)
School of Computer, Central China Normal University
Wuhan, 430079, P.R. China
(Email: jincong@mail.ccnu.edu.cn)
(Received Apr. 20, 2016; revised and accepted June 10, 2016)

Abstract

In recent years, several algorithms of image encryption have been proposed independently. In this paper, an algorithm of image encryption based on general two-dimensional Arnold transform with keys and quantum chaotic map is proposed. First, the key streams are generated by the two-dimensional logistic map as initial conditions and parameters. Second, general Arnold scrambling algorithm with keys is exploited to permute the pixels of color components, R, G and B. Finally a serial of pseudo-random numbers generated by the quantum chaotic map is applied to modify the value of diffused pixels. In order to get the high randomness and the high complexity, the two-dimensional logistic map and quantum chaotic map are coupled with nearest-neighboring coupled-map lattices. Theoretical analyses and computer simulations confirm that the new algorithm has high level of security.

Keywords: Arnold scrambling algorithm, coupled-map lattices, image encryption, quantum chaotic map, two-dimensional logistic map

1 Introduction

1.1 Background

With the rapid growth of the transmission over the Internet, the security of digital image acquires a major concern. So image encryption becomes a hot area and a challenging task. In order to protect personal information, various image encryption algorithms are designed and proposed such as two-dimensional cellular automata-based method [20], Henon chaotic map [10, 21], Chen's hyper-chaotic system [12], Arnold transform [3, 4] and so on. As a classical algorithm Arnold transform has many advantages over others. But an obvious weakness is that it only can be applied to square matrix $N \times N$ and an ideal encryption scheme should not have periodicity. In this paper an excellent method is proposed to solve the problem. Chaotic systems have many good features such as sensitivity to initial conditions and parameters, mixing property, high efficiency and ergodicity. Inspired by the subtle similarity between chaotic systems and cryptosystem, various encryption algorithms based on chaotic map are proposed in the literature. Herein, quantum chaotic system is applied to generate pseudo-random sequence to encrypt color images in the proposed cryptosystem.

1.2 Related Work

Image is one of the most important information representation models and widely used in modern society. An international standard of encryption algorithm is not only suitable for a partial compression algorithm but permutation and diffusion properties. Permutation and diffusion properties are satisfied in cellular automata-based (CA) image system [20]. Ping proposed a novel CA-based multiple image encryption by using a kind of two-dimensional reversible CA, and by using a circular chaining mode of operation. The proposed method allows images to be processed in a 2-D way and makes the statistical information of each plain image in the group hidden in all cipher images.

In order to disturb the high correlation among pixels, the Arnold cat map [3, 4] is a good scrambling tool which has been used widely in various cryptographic and steganographic applications. Chen et al. [3] analyzed the period distribution of the cat map systematically. [4] reported a new image encryption algorithm based on singular value decomposition and Arnold transform. However, in all of these algorithms have two weaknesses, one is that the iteration times are very limited; the other is that the width and height of the plain-image must be identical. Here we propose perfect methods to solve these problems so that the proposed algorithm can be accepted widely.

Chaos-based cryptographic scheme has many brilliant advantages different from other algorithms such as sensitivity to initial conditions and parameters, mixing property, high efficiency non-periodicity and control parame-

ters [7, 15]. In recent years various encryption algorithms based on chaotic map are proposed [26, 27]. Wang and Guo [27] utilized a logistic map for generating a matrix to diffuse the left block of the plain image and then the diffused image was used as the right block of the cipher image. Tang [26] presented an algorithm dividing an input image into overlapping blocks, shuffling image blocks to make initial encryption, exploiting a chaotic map and Arnold transform to generate secret matrices, and achieving final encryption by conducting exclusive OR operations between corresponding elements of each block and a random secret matrix. Jawad [9] enhanced the security level of conventional Blowfish algorithm (BA) for color image encryption by modifying it with new F-function. And the dynamic S-box and XOR operator were generated from the F-function via four-dimensional hyperchaotic map. Lately, in [2] quantum chaos theory becomes a tool that can be used to improve the quality of pseudo-random number generators. The randomness and non-periodicity of quantum chaotic map are successfully verified by statistical complexity and the normalized Shannon entropy. So we apply these characteristics to encrypt the color image for achieving the high randomness and acquiring the non-periodicity that is caused by Arnold transform.

Generally, there are two main stages in the structure of chaos-based algorithm which consists of permutation and diffusion stages. The permutation stage shifts the position of pixels of the plain-image by some chaotic map. General Arnold transform with keys finishes the permutation stage and provides an enough large key space. The diffusion stage modifies the pixels values of shuffled image via chaotic sequences so that a minor change in one pixel of the plain-image causes a totally different cipher-image. Chaotic sequences generated by quantum chaotic map accomplished the diffusion stage and improved the randomness and complexity of the proposed cryptosystem. The diffusion-permutation-based algorithm should have a large key space and the long periodicity of permutation to increase the security. For this purpose, many researchers turn to find some improved chaos-based algorithms with large key spaces and good permutation and diffusion techniques.

1.3 Contribution and Organization

In order to encrypt all color images by Arnold transform algorithm, it is essential to make up the rectangular image into a square. Without loss of generality, we assume that the size of the color plain-image P is $W \times H$, where W is the width of the image, H is the height of the image. Through the method the plain-image is converted into a new image whose size is $N \times N$. Due to the color image that is composed of three color components, we convert three components into three matrices, namely R, G, B. General Arnold transform with keys means that parameters of the matrix A is a set of secret values. We add the matrix $(k\mu, k\nu)^T$ as secret values during the process that Arnold transform is iterated n times. The experiment proves that the chaos character is better when $n = 6$. So we get three different matrices $(k\mu_i, k\nu_i)^T$ $(i = 1, 2, 3)$ as keys to improve the high randomness and enlarge the key space. And then quantum chaotic map [1, 6, 24] is applied to generate three matrices X, Y, Z of size $N \times N$ to encrypt three matrices R, G and B. In this process, the initial condition of quantum chaotic map is a pseudo-random number, which is altered with the time of iteration. For the high complexity and the high randomness, in this paper chaotic maps are coupled with nearest-neighboring coupled-map (NCML), which extremely increases the security and sensitivity of the proposed algorithm.

The major contribution of the proposed algorithm include following points:

1) Provide a method (Equation (10)) to map an arbitrary value into a given interval to meet the demands of two-dimensional logistic map and quantum chaotic map;

2) Add matrices $(k\mu_i, k\nu_i)^T (i = 1, 2, 3)$ as keys into general Arnold transform to enlarge the key space and improve the randomness;

3) Key generator is an address mapping table, which is generated by two-dimensional logistic map. According to session keys we obtain initial conditions and parameters so that improve the sensitivity of the key generator.

The rest of this paper is organized in the following manners: Section 2 introduce the basic theory of the proposed cryptosystem. Section 3 the proposed cryptosystem is explained detailed. Section 4 simulation results and security analysis are proposed. Finally the conclusion is drawn in Section 5.

2 Basic Theory of the Proposed Cryptosystem

2.1 Two-dimensional Logistic Map

In this paper two-dimensional logistic map is applied whose definition is as follows: The two-dimensional logistic map is described as [14, 29]:

$$\begin{aligned} \varphi_1(x_n) &= \mu_1 x_n(1 - x_n) + \nu_1 y_n^2 \\ \varphi_1(y_n) &= \mu_2 y_n(1 - y_n) + \nu_2(x_n^2 + x_n y_n) \end{aligned} \tag{1}$$

when $2.75 < \mu_1 \leq 3.4$, $2.75 < \mu_2 \leq 3.45$, $0.15 < \nu_1 \leq 0.21$ and $0.13 < \nu_2 \leq 0.15$, the system can generate pseudo-numbers in the region (0,1]. All parameters are generated by key generator.

2.2 General Arnold Transform with Keys

We set that the location of the plain-image pixel is (x, y), the location of the cipher-image pixel is (x', y'). The

definition of general Arnold transform is given in [25]:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = A \begin{bmatrix} x \\ y \end{bmatrix} (modN), A = \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} \quad (2)$$

where we set $N = 256$. When $a = b = 1$, Equation (2) is a classical two-dimensional Arnold map. In order to improve security of the cryptosystem, parameters a and b are used as secret keys, which are generated by key generator. Because Arnold transform is a bijection transform, the result of iterating Equation (2) k times still is a bijection transform. In other words, after the process of iteration for k times, point (x, y) become (x', y') and (x', y') is the one and only one point. Due to the result of orthogonal transformation is a limited discrete set, we can add a matrix $(k\mu,k\nu)^T$ as a set of secret keys to enlarge the key space. So we get general Arnold transform with keys whose definition is as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = A^n \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} k\mu \\ k\nu \end{bmatrix} (modN), A = \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} \quad (3)$$

where n is iteration times of the matrix A. According to the inverse transformation of Equation (3), the corresponding decryption algorithm is shown as follows:

$$\begin{bmatrix} x \\ y \end{bmatrix} = A^{-n} \begin{bmatrix} x' - k\mu \\ y' - k\nu \end{bmatrix} (modN), A^{-n} = \begin{bmatrix} ab+1 & -a \\ -b & 1 \end{bmatrix} \quad (4)$$

2.3 Quantum Chaotic Map

Dissipative quantum systems are often described in where the system is coupled to a path of harmonic oscillators to construct a quantum logistic map [1, 6, 24] with quantum corrections. In [1], authors analyze the effects of quantum corrections and state $\alpha = <\alpha>+\delta\alpha$, where $\delta\alpha$ shows a quantum fluctuation about $<\alpha>$. Furthermore, they prove that the very lowest-order quantum corrections can yield the chaotic map as follows:

$$\begin{aligned} \varphi_2(x'_n) &= r(x'_n - |x'_n|^2) - ry'_n \\ \varphi_2(y'_n) &= -y'_n \exp(-2\beta) + \exp(-\beta)r[(2 - x'_n - x'^*_n)y'_n \\ &\quad - x'_n z'^*_n - x'^*_n z'_n] \\ \varphi_2(z'_n) &= -z'_n \exp(-2\beta) + \exp(-\beta)r[2(1 - x'^*_n)z'_n \\ &\quad - 2x'_n y'_n - x'_n] \end{aligned} \quad (5)$$

where $x' = <\alpha>$, $y' = <\delta\alpha \dagger \delta\alpha>$, $z' = <\delta\alpha\ \delta\alpha>$, and β is dissipation parameter. Generally y, x'_n , y'_n and z'_n are complex numbers with $x'_n{}^*$ being the complex conjugate of x'_n and similarly for z'_n. However, if we set the initial value to be real number, then all successive value will also be real. According to [2], the range of the parameters as follows: $0 \le x'_n \le 1$, $0 \le y'_n \le 0.1$, $0 \le z'_n \le 0.2$, $x'_n{}^* = x'_n$, $z'_n{}^* = z'_n$. They conclude that the best value of the control parameter (r) and dissipation parameter (β) are $r = 3.99$, and $\beta \ge 6$. So we set $r = 3.99$, $\beta = 6$, and iterate Equation (5) with real initial parameters x'_0, y'_0,z'_0, $x'_0{}^*$ and $z'_0{}^*$.

2.4 Nearest-neighboring Coupled-map Lattices

The two-dimensional logistic map and the quantum chaotic map proposed in Sections 2.1 and 2.3 are independently coupled with NCML [5, 11] as follows:

$$z_{n+1}(j) = (1 - \varepsilon)\varphi(z_n(j + 1)) + \varepsilon\varphi(z_n(j + 1)) \quad (6)$$

where $n = 0, 1,..., L$-1 is the time index; $j = 1, 2, ... , T$ is the lattice state index; function φ represents a chaotic map such as φ_1, φ_2; $\varepsilon\in(0,1)$ is a coupling constant; L is the length of the plain-text; and T is maximum value of lattice state index. Here, T is chosen as 2 and 3 for the two-dimensional logistic map and the quantum chaotic map, while the other parameter is selected as ε= 0.001 to have good chaotic properties [5, 11]. Moreover, the periodic boundary condition, i.e., $z_n(j+T) = z_n(j)$ is imposed into this system.

Applying Equation (1) to Equation (6), the coupling of two-dimensional logistic map is defined as follows:

$$\begin{aligned} x_{n+1} &= (1 - \varepsilon)\varphi(x_n) + \varepsilon\varphi(y_n) \\ y_{n+1} &= (1 - \varepsilon)\varphi(y_n) + \varepsilon\varphi(x_n) \end{aligned} \quad (7)$$

and by applying Equation (2) to Equation (6), the coupling of quantum chaotic map is defined as follows:

$$\begin{aligned} x'_{n+1} &= (1 - \varepsilon)\varphi(x'_{n+1}) + \varepsilon\varphi(y'_{n+1}) \\ y'_{n+1} &= (1 - \varepsilon)\varphi(y'_{n+1}) + \varepsilon\varphi(z'_{n+1}) \\ z'_{n+1} &= (1 - \varepsilon)\varphi(z'_{n+1}) + \varepsilon\varphi(x'_{n+1}) \end{aligned} \quad (8)$$

Iterating Equation (7) and Equation (8), the required key streams for the proposed cryptosystem are produced.

3 Proposed Cryptosystem

In this section, we combine the generation process with the image processing, the permutation process and the diffusion process. The architecture of the overall image encryption cryptosystem using the proposed algorithm is shown in Figure 1.

3.1 Generation of the Initial Conditions and Parameters

The proposed cryptosystem utilizes a 128-bit external secret key, K, which is divided into 8-bit blocks, k_i, referred to as session keys. The 128-bit external secret key is given by:

$$K = k_1, k_2, ..., k_{16}. \quad (9)$$

In order to increase the security of the proposed algorithm, we apply the two-dimensional logistic map Equation (1) and nearest-neighboring coupled-map lattices Equation (6) so that the initial conditions and parameters of the system are extremely sensitive to the changes in even a single bit in the 128-bit secret key. The detailed process of key generator is described as follows:

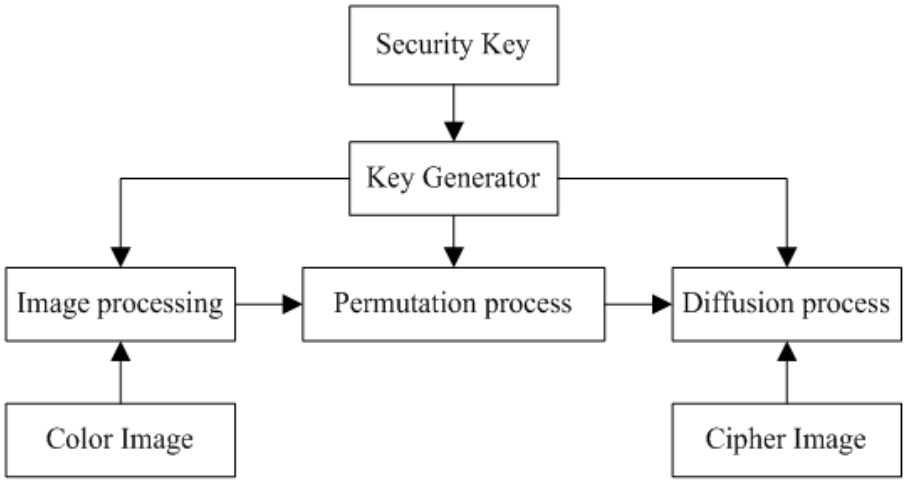


Figure 1: Overall architecture of the proposed cryptosystem

Step 1: Apply k_1, k_2, k_3, k_4 to generate μ_1, μ_2, ν_1, ν_2 respectively. We have known that when $2.75 < \mu_1 \leq 3.4$, $2.75 < \mu_2 \leq 3.45$, $0.15 < \nu_1 \leq 0.21$ and $0.13 < \nu_2 \leq 0.15$ the two-dimensional logistic map generates chaos. We set $a < t_i \leq b$, the initial conditions and parameters of system are derived as follows:

$$t_i = (\frac{k_i}{256} \times 100) mod[(b-a) \times 100] \div 100 + a \quad (10)$$

where we set $\mu_1 = t_1$, $\mu_2 = t_2$, $\nu_1 = t_3$, $\nu_2 = t_4$. So for the different k_i we can get different t_i and make sure that μ_1, μ_2, ν_1, ν_2 are in the region that the system generate chaos.

Step 2: Apply k_5, k_6, ... , k_{16} as initial condition to generate other key values. t_{max} = max([k_5, k_6, ... , k_{16}]). t_{min} = min([k_5, k_6, ... , k_{16}]). t_{ssv} = min([k_5, k_6, ... , k_{16}] - t_{min}). We set $x_0 = t_{min} \div 256$, $y_0 = t_{ssv} \div 256$, and iterate Equation (7) for ceil($t_{max} \div 2$) times with μ_1, μ_2, ν_1, ν_2, x_0, y_0 and then save their output in a new vector E whose size is $2 \times$ceil($t_{max} \div 2$). Apply the following Equation (11):

$$t_i = E_{k_i} \quad (11)$$

where $i = 5, 6, \ldots, 16$ and t_i are in the region (0, 1].

Step 3: In order to improve randomness and complexity of the encryption algorithm and broaden the key space, According to Equation (4) three sets of secret keys, a_i, b_i and $(k\mu_i, k\nu_i)^T$ are required to encrypt three component of the color image R, G, B respectively. Without loss of generality, we assume that the size of the color plain-image P is $W \times H$. Apply the transformation as the following equation to t_5, t_6, t_7:

$$\begin{aligned} a_{i-4} &= [floor(t_i \times W \times H) mod 256]/16 \\ b_{i-4} &= [floor(t_i \times W \times H) mod 256] mod 16 \end{aligned} \quad (12)$$

where a_i, b_i (i = 1, 2, 3) are the first four digits and the last four digits of eight-digit binary number respectively.

Apply the transformation as following equations to a_8, a_9, a_{10}:

$$ku_{i-7} = floor(t_i \times W \times H) mod 256 \quad (13)$$

Apply the transformation as following equations to a_{11}, a_{12}, a_{13}:

$$ku_{i-10} = floor(t_i \times W \times H) mod 256 \quad (14)$$

Step 4: Recalling as mention in Section 2.3, $y'_n \in [0,0.1]$, $z'_n \in [0,0.2]$. Applying Equation (10) analogously initial parameters x'_0, y'_0, z'_0 are derived as follows:

$$\begin{aligned} x'_0 &= t_{14} \\ y'_0 &= [(t_{15} \times 10) mod 1] \div 10 \\ z'_0 &= [(t_{15} \times 10) mod 2] \div 10 \end{aligned} \quad (15)$$

To this end, all initial conditions and parameters are generated. The proposed chaotic algorithm is greatly sensitive to secret key so that even a change in the secret key causes completely different results; as a result, the proposed algorithm with total complexity of 2^{128} can resist against any key sensitivity attack and any bruteforce attack.

3.2 Proposed Encryption Algorithm

Due to Arnold transform is not adapt to image $N \times N$, it is essential to transform image $W \times H$ into $N \times N$. we give the following equation to meet the demand:

$$N = \max([W, H]) \quad (16)$$

where set N is a bigger value between W and H. When $W = H$, $N = W = H$. In other words, if the image is

square, it remains unchanged; otherwise it will be amplified. Pixel values of increased part of the image are filled with random numbers, which are generated by the random function. It not only improves the randomness of the cryptosystem, but also if we can not get the real width and height of plain-image before decryption, we can not finish the decryption. We assume that the color plain-image P of $W \times H$ becomes P' of $N \times N$ by the transformation above. In this process we convert the matrix P with red green and blue components into three matrices R, G and B. Taking an example of the matrix R, the detailed encryption algorithm is described as follows:

Permutation process:

The process applies pseudo-random key streams generated by Equation (12), Equation (13) and Equation (14) according to Section 3.1 to permute pixels of the color image. Substituting a_1, b_1 and $(k\mu_1, k\nu_1)^T$ into Equation (3) and iterate it for n times. According to the experiment we find that when $n = 6$ the proposed cryptosystem performs better. Apply the same permutation process into G and B respectively, the plain-image becomes a cipher-image after n times iteration, namely, Matrices R, G and B all becomes R',G' and B'.

Diffusion process:

Step 1: Set L=$N \times N$ and generate the initial condition $(x'_0,\ y'_0,\ z'_0)$ according to Section 3.1 and iterate Equation (8) m+L times and discard the former m values to avoid harmful effects. Where m also can be as a secret key, we set $m = 13$ for convenience. Discarding the first m result and Sorting these L values as X= $\{x_{m+1}, x_{m+2}, \ldots, x_{m+L}\}$, Y= $\{y_{m+1}, y_{m+2}, \ldots, y_{m+L}\}$ and Z= $\{z_{m+1}, z_{m+2}, \ldots, z_{m+L}\}$.

Step 2: Transforming three matrices R',G' and B' into vectors $\overrightarrow{R'}$=$\{r_1, r_2, \ldots, r_L\}$,$\overrightarrow{G'}$ = $\{g_1, g_2, \ldots, g_L\}$, and $\overrightarrow{B'}$=$\{b_1, b_2, \ldots, b_L\}$ respectively.

Step 3: Applying the encryption transformation as the following equations:

$$\begin{aligned} C_{ri} &= ((floor(r_{m+i} \times W \times H \times k_6 \times k_7 \times k_9 \times k_{10} \\ &\quad \times k_{12} \times k_{13}) mod 256) \oplus r_i \\ C_{gi} &= ((floor(g_{m+i} \times W \times H \times k_5 \times k_7 \times k_8 \times k_{10} \\ &\quad \times k_{11} \times k_{13}) mod 256) \oplus g_i \\ C_{bi} &= ((floor(b_{m+i} \times W \times H \times k_5 \times k_6 \times k_8 \times k_9 \\ &\quad \times k_{11} \times k_{12}) mod 256) \oplus b_i \end{aligned} \quad (17)$$

where set initial values $i = 1$.Set $i = i$+1 and then iterating this step until $i \leq L$ we can get three matrices C_r, C_g and C_b.

Remark 1. M *mod* N *involves modulo operation giving a integer result between 0 and* N.

Remark 2. *ceil(a) returns the smallest integer value that is bigger than or equal to the value of a.*

Remark 3. max*(*[k_1, k_2, ... , k_n]*) returns the biggest value among all of them.*

Remark 4. min*(*[k_1, k_2, ... , k_n]*) returns the smallest value among all of them.*

Obviously the generation of the key stream depends on the 128-bit external secret key, K, and the width W, the height H of plain-image. The generation of initial conditions and parameters are derived by the two-dimensional logistic map and the nearest-neighboring coupled-map lattices. And the key stream is chosen from an array of chaotic sequence, which makes sure that cryptosystem has a high complexity, sensitivity and randomness. In the encryption process, the Arnold transform with keys is applied to permute the pixels of color components. And the quantum chaotic map is exploited to generate the key streams to modify the value of diffused pixels.

3.3 Proposed Decryption Algorithm

The decryption process is similar to the encryption one, achieved in the reverse order. In decryption process transforming matrices C_r, C_g and C_b into three vectors $\overrightarrow{C_r}$= $\{r_1, r_2, \ldots, r_L\}$,$\overrightarrow{C_g}$ = $\{g_1, g_2, \ldots, g_L\}$, and $\overrightarrow{C_b}$= $\{b_1, b_2, \ldots, b_L\}$ respectively. the detail decryption algorithm is described as follows:

Step 1: Apply the external 128-bit secret key used in the encryption process. According to Section 3.1 generate the initial conditions and parameters.

Step 2: Substituting the initial condition $(x'_0,\ y'_0,\ z'_0)$ and iterating Equation (8) m+L times, discarding the former m values to avoid harmful effects, where $m = 13$.

Step 3: Sorting these values X= $\{x_{m+1}, x_{m+2}, \ldots, x_{m+L}\}$, Y= $\{y_{m+1}, y_{m+2}, \ldots, y_{m+L}\}$ and Z= $\{z_{m+1}, z_{m+2}, \ldots, z_{m+L}\}$. Setting $i = 1$ and iterate Equation (17)) until $i = L$ we can get three vectors $\overrightarrow{C'_r}$, $\overrightarrow{C'_g}$ and $\overrightarrow{C'_b}$.

Step 4: We convert these vectors into three matrices ${R'}_r$, ${G'}_g$ and ${B'}_b$ whose size are all $N \times N$. Substituting parameters a_i, b_i and the initial condition $(k\mu_i, k\nu_i)^T$ $(i = 1, 2, 3)$, and then using the encryption algorithm Equation (4) we get R', G', B' of the image, According to the width W and the height H of the plain-image we tailor R', G', B' and get plain values of R, G and B. In this way the encryption process finished.

4 Performance and Security Analysis

We have done many measures to check the security and performance of the proposed cryptosystem. These measures consist of statistical analysis, key sensitivity analy-

Table 1: The related correlation coefficient between plain-image and cipher-image

Scan direction	Lena					
	Plain-image			Cipher-image		
	R	G	B	R	G	B
Horizontal	0.972978	0.954127	0.938846	0.001418	0.000082	-0.002191
Horizontal	0.981110	0.951084	0.934597	-0.007127	0.000587	0.000086
Vertical	0.958757	0.934720	0.915541	0.000700	0.000647	0.004526

sis, key space analysis, speed performance. Each of these measures is shown in detail in the following subsections.

4.1 Statistical Analysis

4.1.1 Histogram of Encrypted Image

An ideal cipher-image should has a uniform frequency distribution. From Figures 2, 3, 4 and 5, it is obvious that the histogram of cipher-image are independent of the type of plain-image such as binary, gray level and are nearly uniform and significantly different from the histogram of the original images. Hence it dose not provide any useful statistic data in the cipher-image to trigger any statistical attacks to the algorithm.

4.1.2 Correlation of Two Adjacent Pixels

In order to get the correlation of two adjacent pixels we have selected 3000 pairs of two adjacent pixels from plain-image and cipher-image randomly for the experiment and have calculated the correlation coefficients as follows:

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

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

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

$$r_{xy} = \frac{Cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \quad (18)$$

The x and y represent gray-level values of two adjacent pixels. The distribution of two horizontally adjacent pixels of R, G and B components of plain-image and cipher-image Lena is shown in Figure 6.

Table 1 shows that the correlation between adjacent pixels of the cipher-image is much smaller than that of plain-image, so we claim that the adjacent pixels of the plain-image are uncorrelated by the proposed cryptosystem effectively from different directions.

In color images there are the high correlation between adjacent pixels of R, G and B components. The proposed cryptosystem encrypt pixels of color components so that make them affect one another. Table 2 and Table 3 show the results of the same position correlations and related adjacent position correlations between R, G and B components of plain-image and cipher-image.

Table 2: Similar position correlations between R, G and B components

Scan direction	R-G	R-B	G-B
Plain-image	0.929848	0.797885	0.949200
Cipher-image	0.000279	0.005105	0.004628

Table 3: Adjacent position correlation between R, G and B components

Scan direction	R-G	R-B	G-B
Plain-image	0.896510	0.756614	0.891265
Cipher-image	0.002288	0.006150	0.001227

4.2 Key Sensitivity Analysis

When one bit of the security key is altered, there are obviously differences between two cipher-images. The number of pixels change rate ($NPCR$) and the unified average changing intensity ($UACI$) for the two encrypted images are applied to measure the number of pixels change rate.

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

$$UACI = \frac{1}{N \times N}[\sum_{i,j} \frac{|C(i,j) - C'(i,j)|}{255}] \times 100\% \quad (19)$$

where N is the height (width) of the encrypted image. We get two encrypted images C and C', whose secret keys are different in only one bit. We also define a two-dimensional array D, which has the same size as C. If $C(i,j) = C'(i,j)$, then $D(i,j) = 0$, otherwise $D(i,j) = 1$. To resist against security key attack, $NPCR$ and $UACI$ values should be large enough for an ideal cipher system. When the secret key is altered from 207 21 42 61 122 203 97 76 101 5 7 241 139 28 98 17 to 208 21 42 61 122 203 97 76 101 5 7 241 139 28 98 17 the differences is made greatly.

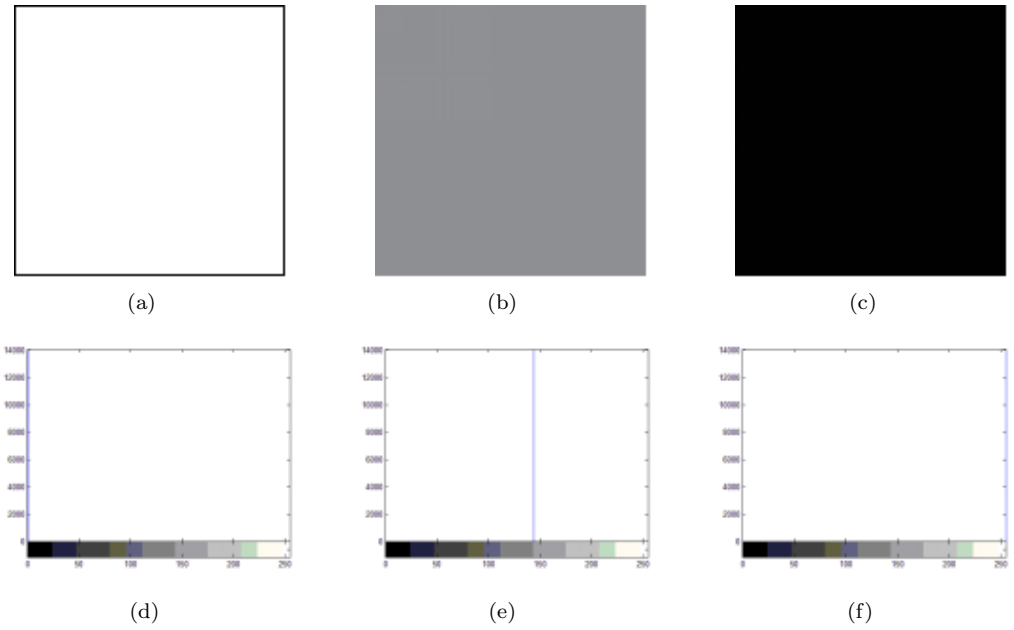


Figure 2: (a) Original white image, (b) the original monolithic gray-level image, (c) the original black image, (d) the histogram of the white image, (e) the histogram of the monolithic gray-level image, (f) the histogram of the black image

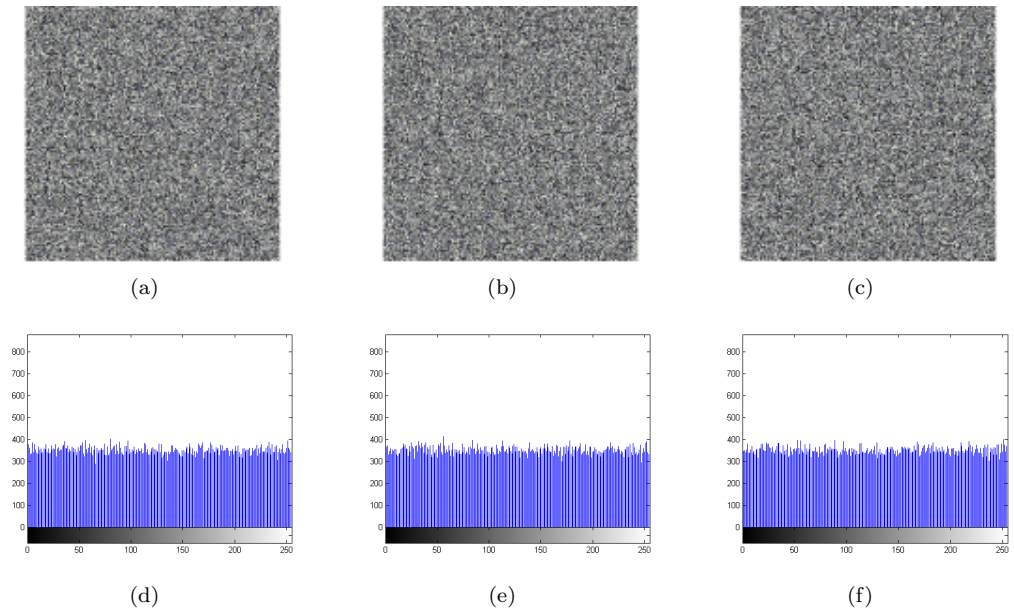
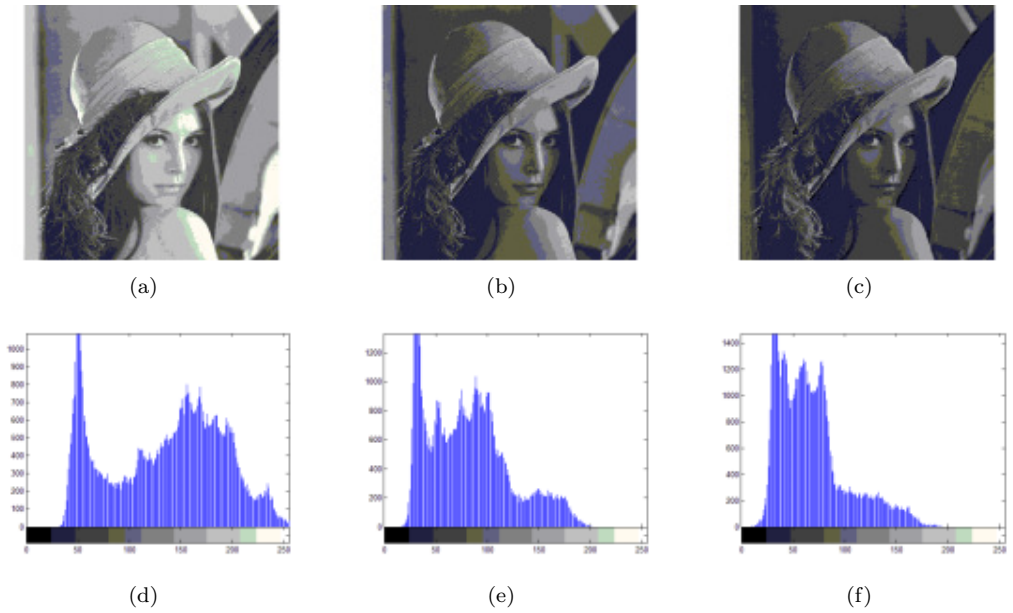


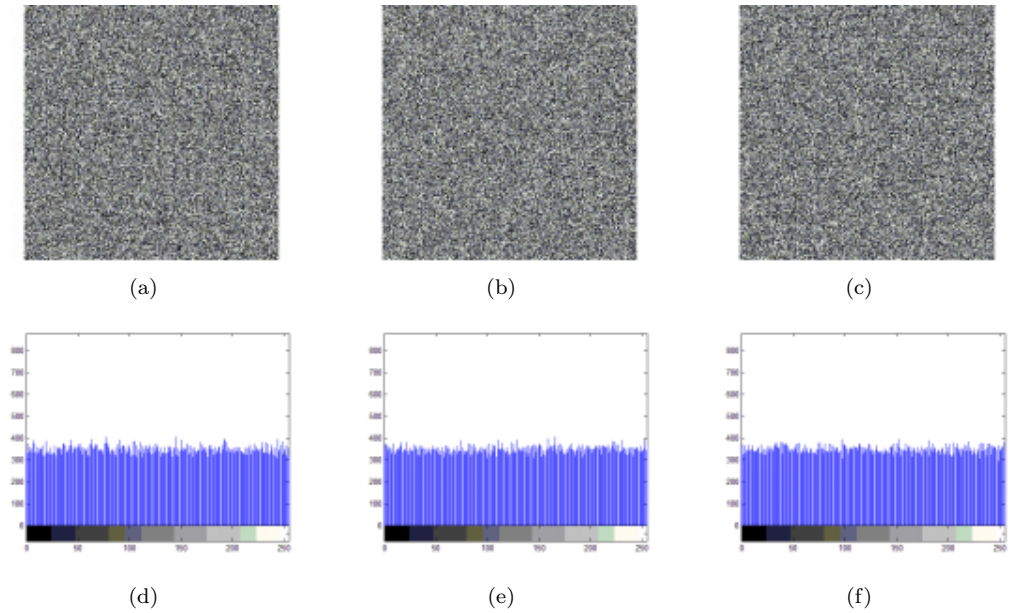
Figure 3: (a) Cipher of white image, (b) the cipher of monolithic gray-level image, (c) the cipher of the black image, (d) the histogram of the encrypted white image, (e) the histogram of the encrypted monolithic gray-level image, (f) the histogram of the encrypted black image



(a) (b) (c)

(d) (e) (f)

Figure 4: (a) Plain-image Lena-R, (b) the plain-image Lena-G, (c) the plain-image Lena-B, (d) the histogram of the plain-image Lena-R, (e) the histogram of the plain-image Lena-G, (f) the histogram of the plain-image Lena-B



(a) (b) (c)

(d) (e) (f)

Figure 5: (a) The encrypted image Lena-R, (b) the encrypted image Lena-G, (c) the encrypted image Lena-B, (d) the histogram of the encrypted image Lena-R, (e) the histogram of the encrypted image Lena-G, (f) the histogram of the encrypted image Lena-B

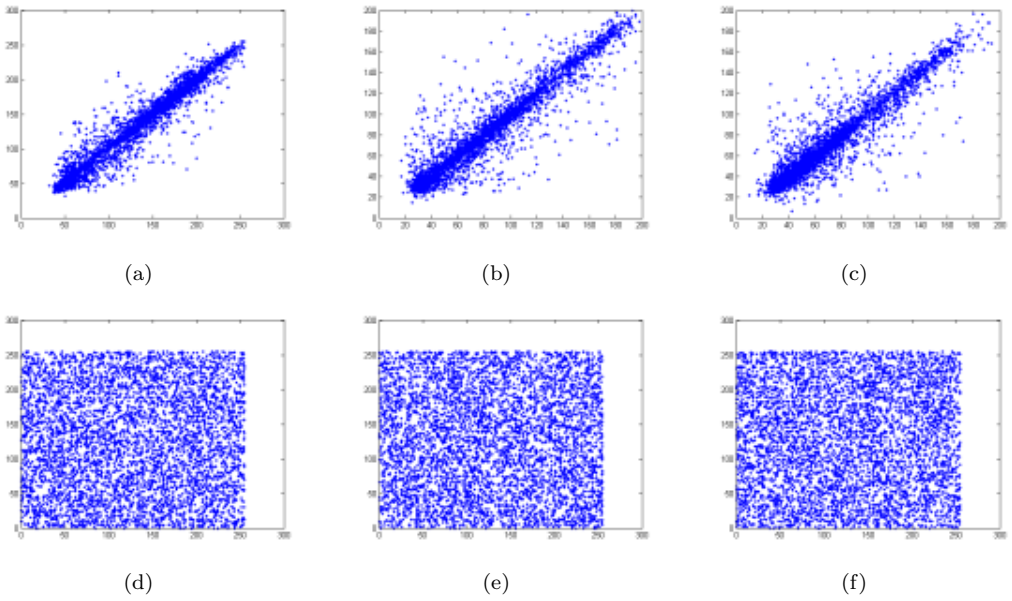


Figure 6: Distribution of two horizontally adjacent pixels in the plain-image of Lena in the (a) red, (b) green and (c) blue components. The distribution of two horizontally adjacent pixels in the cipher-image of Lena in the (d) red, (e) green and (f) blue components. (Color figure online)

Table 4 shows the average $NPCR_{R,G,B}$ and $UACI_{R,G,B}$ values and compares this proposed algorithm with other schemes in terms of the key sensitivity. The proposed algorithm is sensitive dependent on initial conditions and parameters.

Table 4: Comparison of the average $NPCR_{R,G,B}$ and $UACI_{R,G,B}$

Algorithm	Average ($NPCR$)	Average ($UACI$)
Proposed	0.996896	0.334402
[1]	0.000041	0.003320
[6]	0.996355	0.334188
[16]	0.996028	0.334289
[13]	0.000384	0.000433
[28]	0.996358	0.334428
[23]	0.996828	0.334898

4.3 Key Space Analysis

An ideal encryption scheme should have a enough large key space to defend brute-force attack. The size of the key space should be bigger than 2100 to provide a high level of security from the crytography of view [17, 22]. Due to the secret key is 128-bit long, the key space is 2^{128}. We can conclude that the proposed algorithm is large enough to resist all kinds of brute-force attacks.

4.4 Speed Performance

Apart from the security considerations, some other aspects on image cryptosystem algorithm are also important, particularly the running speed for real time Internet multimedia applications. In fact the actual execution time of a cryptosystem depends on many factors, such as CPU structure, OS, memory size, programming skill and so on. We have analyzed the speed of the proposed image encryption technique on an Intel Core I3 CPU 2.3 GHz and 3.99 GB of RAM running on Windows XP and MATLAB 7.1 programming. For accuracy each set of the timing tests was executed several times for considerable number of images and then the average obtained was reported. In Table 5 we can see the comparison results for the proposed scheme and other schemes. Table 5 shows that the proposed algorithm is very fast compared to the other schemes.

Table 5: Comparison of encryption speeds for the proposed scheme and different schemes

Algorithm	Speed (Mbit/s)
Proposed	9.89
[6]	8.11
[16]	5.15
[23]	9.12
[19]	9.39
[18]	8.16
[8]	1.45

5 Conclusions

This paper has realized the quantum image encryption and decryption and protected the information. Image information is ciphered by the proposed encryption algorithm based on general Arnold transform with keys and quantum chaotic map. By improving the Arnold transform algorithm, we not only enlarge the key space to resist against any key sensitivity and any brute-force attack, but also raise the running speed of the process of the encryption. The experiment shows that only one time general Arnold transform with keys has a good result. In order to enhance the sensitivity of the cryptosystem, the generator of the initial conditions and parameters apply the addressing map to get corresponding value. Quantum chaotic map possesses perfect chaotic character, which is used to change the pixel values of the plain-image and eliminate the periodicity generated by the algorithm of general Arnold transform with keys.

The experimental results demonstrate that the proposed method can achieve the high security level to resist various attacks and possesses the high encryption speed (speed $>$ 9.89Mbit/s). Accordingly the proposed algorithm is suitable to practical uses to protect the digital image information over the Internet.

Acknowledgments

This work was supported by the fundamental research funds for the central universities (Grant No. CCNU15GF007).

References

[1] A. Akhshani, A. Akhavan, S. C. Lim, and Z. Hassan, "An image encryption scheme based on quantum logistic map," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 12, pp. 4653–4661, 2012.

[2] A. Akhshani, A. Akhavan, A. Mobaraki, S. C. Lim, and Z. Hassan, "Pseudo random number generator based on quantum chaotic map," *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 1, pp. 101–111, 2014.

[3] F. Chen, K. W. Wong, X. Liao, and T. Xiang, "Period distribution of generalized discrete arnold cat map," *Theoretical Computer Science*, vol. 552, no. 4, pp. 13–25, 2014.

[4] L. Chen, D. Zhao, and F. Ge, "Mage encryption based on singular value decomposition and arnold transform in fractional domain," *Optics Communications*, vol. 219, no. 6, pp. 98–103, 2013.

[5] M. Ding and W. Yang, "Stability of synchronous chaos and on-off intermittency in coupled map lattices," *Physical Review E*, vol. 56, no. 4, pp. 4009–4016, 1997.

[6] A. A. A. El-Latif, L. Li, N. Wang, Q. Han, and X. Niu, "A new approach to chaotic image encryption based on quantum chaotic system, exploiting color spaces," *Signal Processing*, vol. 93, no. 11, pp. 2986–3000, 2013.

[7] J. Fridrich, "Symmetric ciphers based on two dimensional chaotic maps," *International Journal of Bifurcation & Chaos*, vol. 8, no. 6, pp. 1259–1284, 1998.

[8] T. G. Gao and Z. Q. Chen, "A new image encryption algorithm based on hyper-chaos," *Physics Letters A*, vol. 372, no. 4, pp. 394–400, 2008.

[9] L. M. Jawad and G. Sulong, "Chaotic map-embedded blowfish algorithm for security enhancement of colour image encryption," *Nonlinear Dynamics*, vol. 81, no. 4, pp. 1–15, 2015.

[10] J. Khan, J. Ahmad, and S. O. Hwang, "An efficient image ecryption scheme based on: Henon map, skew tent map and s-box," *International Conference on Modeling, Simulation and Applied Optimization*, vol. 10, pp. 1–6, 2015.

[11] M. Khan, T. Shah, and S. I. Batool, "Texture analysis of chaotic coupled map lattices based image encryption algorithm," *3D Research*, vol. 15, no. 3, pp. 1–5, 2015.

[12] D. L, "Color image encryption algorithm based on chua's circuit and chen's hyper-chaotic system," *Journal of Information & Computational Science*, vol. 12, pp. 1021–1028, 2015.

[13] S. Liu, J. Sun, and Z. Xu, "An improved image encryption algorithm based on chaotic system," *Journal of Computers*, vol. 4, no. 11, pp. 1091–1100, 2009.

[14] M. Machkour, A. Saaidi, and M. L. Benmaati, "A novel image encryption algorithm based on the two-dimensional logistic map and the latin square image cipher," *3D Research*, vol. 6, no. 4, pp. 1–18, 2015.

[15] R. Matthew, "One the derivation of a chaotic encryption algorithm," *Cryptologia*, vol. 8, pp. 29–42, 1989.

[16] S. Mazloom and M. A. Eftekhari, "Color image encryption based on coupled nonlinear chaotic map.," *Chaos Solitons Fractals*, vol. 42, no. 3, pp. 1745–1754, 2009.

[17] B. Norouzi, S. M. Seyedzadeh, S. Mirzakuchaki, and M. R. Mosavi, "A novel image encryption based on row-column, masking and main diffusion processes with hyper chaos," *Multimedia Tools & Applications*, vol. 74, no. 3, pp. 781–811, 2015.

[18] V. Patidar, N. K. Pareek, G. Purohit, and K. K. Sud, "Modified substitution-diffusion image cipher using chaotic standard and logistic maps," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 15, pp. 2755–2765, 2010.

[19] V. Patidar, N. K. Pareek, and K. K. Sud, "A new substitution-diffusion based image cipher using chaotic standard and logistic maps," *Communications in Nonlinear Science and Numerical Simulation*, vol. 14, no. 7, pp. 3056–3075, 2009.

[20] P. Ping, Z. J. Wang, and F. Xu, "A two-dimensional cellular automata based method for multiple image," *International Conference on Computer Science & Service System*, vol. 112, pp. 101–104, 2014.

[21] N. S. Raghava, A. Kumar, and A. C. A. Deep, "Improved lsb method for image steganography using henon chaotic map," *Open Journal of Information Security & Applications*, vol. 1, no. 1, pp. 34–42, 2014.

[22] B. Schneier, *Applied Cryptography: Protocol, Algorithms, and Source Code in C*, New York: John Wiley & Sons, 2015.

[23] S. M. Seyedzadeh and S. Mirzakuchaki, "A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map," *Signal Processing*, vol. 92, no. 5, pp. 1202–1215, 2012.

[24] S. M. Seyedzadeh, B. Norouzi, M. R. Mosavi, and S. Mirzakuchaki, "A novel color image encryption algorithm based on spatial permutation and quantum chaotic map," *Communications in Nonlinear Science and Numerical Simulation*, vol. 81, no. 1-2, pp. 1–19, 2015.

[25] X. H. Sun, *Image Encryption Algorithms and Practices with Implementations in C#*, Beijing: Science Press, 2013.

[26] Z. Tang, X. Zhang, and W. Lan, "Efficient image encryption with block shuffling and chaotic map," *Multimedia Tools & Applications*, vol. 74, no. 15, pp. 1–20, 2015.

[27] X. Wang and K. Guo, "A new image alternate encryption algorithm based on chaotic map," *Nonlinear Dynamics*, vol. 76, no. 4, pp. 1943–1950, 2014.

[28] X. Wang, L. Teng, and X. Qin, "A novel color image encryption algorithm based on chaos," *Signal Processing*, vol. 92, no. 4, pp. 1101–1108, 2012.

[29] X. Y. Wang, Y. Q. Zhang, and Y. Y. Zhao, "A novel image encryption scheme based on 2-D logistic map and dna sequence operations," *Nonlinear Dynamics*, vol. 82, no. 3, pp. 1269–1280, 2015.

Hui Liu is a M.S. candidate at the school of computer, Central China Normal University. His research interests include: information security, quantum chaotic.

Cong Jin received the M.S. degrees in applied mathematics from Harbin Institute of Technology, China. She received the Ph.D. in Institute for Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology, China. From 1993 to 2003, she was a Lecturer and then become as a full professor at the Hubei University, China. From 2003 to now, she is a full professor of the school of computer, Central China Normal University, China. She has published more than 150 papers on information security, signal processing, and algorithm design and analysis. Her main research interests include computer network security, digital image processing, and software reliability prediction, etc.