

Quantum image encryption based on generalized Arnold transform and double random-phase encoding

Nan Run Zhou · Tian Xiang Hua · Li Hua Gong ·
Dong Ju Pei · Qing Hong Liao

Received: 24 August 2014 / Accepted: 13 January 2015 / Published online: 28 January 2015
© Springer Science+Business Media New York 2015

Abstract A quantum realization of the generalized Arnold transform is designed. A novel quantum image encryption algorithm based on generalized Arnold transform and double random-phase encoding is proposed. The pixels are scrambled by the generalized Arnold transform, and the gray-level information of images is encoded by the double random-phase operations. The keys of the encryption algorithm include the independent parameters of coefficients matrix, iterative times and classical binary sequences, and thus, the key space is extremely large. Numerical simulations and theoretical analyses demonstrate that the proposed algorithm with good feasibility and effectiveness has lower computational complexity than its classical counterpart.

Keywords Generalized Arnold transform · Double random-phase encoding · Quantum image encryption

1 Introduction

With the rapid development of multimedia technology, more and more important information is embodied in images and videos and the security of private images becomes

N. R. Zhou (✉) · T. X. Hua · L. H. Gong · D. J. Pei · Q. H. Liao
Department of Electronic Information Engineering, Nanchang University, Nanchang 330031, China
e-mail: znr21@163.com; nrzhou@ncu.edu.cn

L. H. Gong
Key Laboratory of Photoelectronics and Telecommunication of Jiangxi Province, Nanchang 330022, China

D. J. Pei
School of Computer and Information Engineering, Jiangxi Agricultural University, Nanchang 330045, China

a serious issue. For private images in conventional computers, there have been many good image encryption algorithms [1–20]. Double random-phase encoding is a classical method of optical encryption [2]. The phase encoding has been developed and employed in some encryption schemes [3–8]. An image encryption method based on compressive sensing and double random-phase encoding was proposed, and the data volume for encryption was lowered due to the dimensional decrease properties of compressive sensing [7]. Zhang et al. proposed an image encryption algorithm by combining fractional Fourier transform with pixel scrambling operation based on double random-phase encoding [8]. However, some studies have shown that the double random-phase encoding scheme is vulnerable to various attacks including known-plaintext attack [9, 10], chosen-ciphertext attack [11] and chosen-plaintext attack [12]. Image scrambling, as a common way to encrypt image data, is to hide image content from illegal user. Arnold transform is an effective image pixel scrambling tool called as the “cat’s mapping” [13]. Arnold transform can scramble the matrix-pixel sequence by encoding a single parameter and reduce the length of the key. A novel image scrambling and watermarking scheme based on the orbits of Arnold transform were proposed, where Arnold transform disordered the pixel positions to obtain a totally visual difference from the original images [14]. An efficient image encryption algorithm with the generalized Arnold map was proposed, which can resist statistical analyses, chosen-plaintext attacks and known-plaintext attacks [15]. Arnold transform is applied widely in digital image scramble. To enhance security of the encryption algorithm, some encryption schemes have been designed by combining Arnold transform with other transforms [16–20]. Liu et al. [16] designed a double image encryption algorithm based on Arnold transform and discrete fractional angular transform. A novel color image encryption method by combining discrete fractional random transform with Arnold transform in the intensity–hue–saturation color space was proposed, where Arnold transform yields good scrambling results and its periodicity ensures the implementation of decryption is accurate and easy [20].

Quantum computation has been applied in many fields of information sciences [21]. The rapid development of quantum computation and quantum computer attracts people to investigate quantum data security. The quantum images as an important part of quantum information will make the applications of quantum computers more widely and comprehensively in the future. A series of methods to represent quantum images were proposed [22–27]. Moreover, a novel enhanced quantum representation (NEQR) for digital images was proposed, which improves the flexible representation of quantum images (FRQI) [28]. Consequently, some new quantum algorithms were developed to secure quantum images [29–37]. For example, Yang et al. [36] proposed a novel gray image encryption scheme based on quantum Fourier transform (QFT) and double random-phase encoding technique, which is heuristic to introduce more optical information processing techniques into quantum scenarios. Jiang et al. [38] proposed the Arnold and Fibonacci scrambling quantum circuits based on FRQI, which does not take advantage of the particularities of “mod 2^n ,” multiply by 2 and subtraction in binary arithmetic. Later, a simplified scheme was presented in [39] to cut down the network complexity apparently. However, there are no quantum versions of some basic classical image transforms, such as generalized Arnold transform, fractional Fourier transform, fractional Mellin transform and so on.

We will design a quantum version of the generalized Arnold transform and will propose a quantum image encryption algorithm by combining generalized Arnold transform with double random-phase encoding technology. The algorithm is composed of two stages, i.e., diffusion and confusion. In the diffusion stage, the random-phase operations are controlled by the classical binary sequence, which can change different angles of the color information for different positions. However, in Yang et al's scheme [36], the angle of every positions in quantum image is changed with a same angle with the phase operation, which cannot guarantee the changes for different positions. In the confusion stage, the generalized Arnold transform is applied to shuffle the positions of image pixels. It changes not only the gray values of pixels but also the locations of pixels. Unlike Yang et al's scheme, the proposed quantum image encryption algorithm is expected with good diffusion and confusion performances. Moreover, the generalized Arnold transform is introduced into the encryption algorithm to increase the number of keys and then enhances the security. Numerical simulations and theoretical analysis are presented to illustrate the feasibility and effectiveness of the proposed algorithm.

The rest of this paper is organized as follows. In Sect. 2, the flexible representation for quantum images and the double random-phase encoding are reviewed. The quantum realization of the generalized Arnold image scrambling is designed in Sect. 3. The proposed quantum image encryption and decryption algorithm is given in Sect. 4. Section 5 is devoted to classical simulation analysis and performance comparison. Finally, a conclusion is drawn in Sect. 6.

2 Flexible representation for quantum images and double random-phase encoding technique

2.1 Flexible representation for quantum images

Classical image is represented by a matrix with the same size of the image, i.e., the number of pixels. In a classical gray image, each pixel consists of the grayscale value and the position information. Inspired by the pixel representation for images in classical computers, a flexible representation for quantum images on quantum computers was proposed [23]. For a quantum image, the color information and the corresponding position information of every pixels are stored into the corresponding quantum states, respectively. According to the flexible representation for quantum images, suppose M is a classical image of size $2^n \times 2^n$, $|M\rangle$ is the storage of the whole quantum states for a grayscale image, the quantum image representation can be expressed as:

$$|M\rangle = \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} |g(y, x)\rangle |yx\rangle$$

$$|g(y, x)\rangle = \cos \theta_i |0\rangle + \sin \theta_i |1\rangle, \theta_i \in \left[0, \frac{\pi}{2}\right], i = yx = 0, 1, \dots, 2^{2n} - 1 \quad (1)$$

where $\theta = (\theta_0, \theta_1, \dots, \theta_{2^{2n}-1})$ is the vector of angles encoding colors, $|g(y, x)\rangle$ encodes the color information of quantum image, $|i\rangle = |yx\rangle = |y\rangle |x\rangle$

$= |y_{n-1}y_{n-2}\dots y_0\rangle |x_{n-1}x_{n-2}\dots x_0\rangle$ encodes the corresponding positions of the quantum image, $|y_{n-1}y_{n-2}\dots y_0\rangle$ encodes the first n -qubit along the vertical location information while $|x_{n-1}x_{n-2}\dots x_0\rangle$ encodes the second n -qubit along the horizontal location information and n is the number of quantum bits required for encoding.

2.2 Double random-phase encoding technique

The double random-phase encoding technique was proposed by Refregier and Javidi in 1995 [2]. The technique can encrypt an original image by using two statistically independent random-phase masks in the input and Fourier planes, respectively. If two random-phase masks are used to encrypt the image in the input and Fourier planes, respectively, the encrypted image would be generalized to a stationary white noise of statistical properties with time shift invariant. If only the random-phase mask is used to encrypt the original image in the input plane, the encrypted image would be a non-stationary white noise of statistical properties changing over time. If one only uses the random-phase mask to encrypt image in the Fourier plane, the encrypted image can easily be deciphered.

Assume $f(x, y)$ is the plaintext image, while $g(x, y)$ is the cipher one. Let (x, y) and (μ, v) denote the spatial plane and the Fourier plane coordinates, respectively, $\phi(x, y)$ and $\varphi(\mu, v)$ denote two white noise sequences in input phase and Fourier phase, which are uniformly distributed from 0 to 1. The random-phase masks $\exp[j2\pi\phi(x, y)]$ and $\exp[j2\pi\varphi(\mu, v)]$ as the keys are generated by two white noise sequences. The encoding and decoding procedures are shown as follows.

$$g(x, y) = \text{FFT}^{-1} \{ \text{FFT} \{ f(x, y) \exp[j2\pi\phi(x, y)] \} \exp[j2\pi\varphi(\mu, v)] \} \quad (2)$$

$$f(x, y) = \text{FFT}^{-1} \{ \text{FFT} \{ g(x, y) \} \exp[-j2\pi\varphi(\mu, v)] \} \exp[-j2\pi\phi(x, y)] \quad (3)$$

where FFT and FFT^{-1} represent the Fourier transform and inverse Fourier transform, respectively.

3 Realization of generalized Arnold transform

3.1 Quantum representation of generalized Arnold transform

Arnold transform was proposed by Arnold [13] in the research of ergodic theory, it was also called cat map. Dyson et al. [40] quoted the transform as an image scrambling method in 1992. The two-dimensional generalized Arnold transform in the form of matrix is defined as

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & t \\ m & tm + 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \pmod{N} = C \begin{bmatrix} x \\ y \end{bmatrix} \pmod{N} \quad (4)$$

The inverse transformation is:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & t \\ m & tm + 1 \end{bmatrix}^{-1} \begin{bmatrix} x' \\ y' \end{bmatrix} \pmod{N} = \begin{bmatrix} tm + 1 & -t \\ -m & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} \pmod{N} \quad (5)$$

where $x, y, x', y' \in \{0, 1, \dots, N - 1\}$, t and m are positive integers, x and y are the pixel coordinates of the original image, N is the size of the square image, x' and y' are the pixel coordinates of the generalized Arnold scrambled image. The generalized Arnold transform in the form of coordinates can be expressed as

$$\begin{cases} x' = (x + ty) \pmod{N} \\ y' = (mx + (tm + 1)y) \pmod{N} \end{cases} \quad (6)$$

The generalized Arnold transform has the features of chaotic mapping, which changes the positions of two pixels. The generalized Arnold transform focuses on manipulating the information about the position of each pixel in the image. Corresponding to the classical image, the quantum representation of the generalized Arnold transform can be described as

$$\begin{cases} |x'\rangle = |(x + ty) \pmod{2^n}\rangle \\ |y'\rangle = |(mx + (tm + 1)y) \pmod{2^n}\rangle \end{cases} \quad (7)$$

3.2 Quantum circuit architecture of generalized Arnold transform

An explicit construction of several elementary quantum networks, i.e., plain adder, adder modulo N , controlled multiplier modulo N and exponentiation modulo N are designed in [41]. The plain adder is a quantum network that can calculate the sum of two numbers. Inputs are encoded in a binary form in the computational basis of selected qubits usually called a quantum register. The addition of two quantum registers $|a\rangle$ and $|b\rangle$ can be written as $|a, b\rangle \rightarrow |a, a + b\rangle$. The plain adder network is illustrated in Fig. 1a. The adder modulo N is a quantum network that can calculate the modular sum of two numbers. The modular addition of two quantum registers $|a\rangle$ and $|b\rangle$ can be expressed as $|a, b\rangle \rightarrow |a, (a + b) \pmod{N}\rangle$. The adder modulo N network is demonstrated in Fig. 1b. However, the plain adder network and the adder modulo N network require the inputs are two n qubits binary numbers. A quantum circuit ADDER-MOD 2^n defined in [39] can accomplish $(a + b) \pmod{2^n}$ simply by ignoring the carry bit from ADDER module. The ADDER-MOD 2^n network is shown in Fig. 1c. In the generalized quantum Arnold transform, the states $|x'\rangle$ and $|y'\rangle$ are independent of each other, which can be realized by connecting several quantum circuit ADDER-MOD 2^n . Hence, the quantum ADDER-MOD 2^n network is fundamental to realize the generalized Arnold transform in quantum computer.

Assuming that x and y are both n qubits binary numbers, $x = x_{n-1}x_{n-2} \dots x_0$, $y = y_{n-1}y_{n-2} \dots y_0$, $x_i, y_i \in \{0, 1\}$, $i = n - 1, n - 2, \dots, 0$. According to the nature of the modulo, $(x + y) \pmod{2^n} = (x \pmod{2^n} + y \pmod{2^n}) \pmod{2^n} = (x + y \pmod{2^n}) \pmod{2^n}$, so $(x + 2y) \pmod{2^n} = (x + 2y \pmod{2^n}) \pmod{2^n}$. The realization of $|x'\rangle$ is divided into t steps, as shown in Fig. 2. The ADDER-MOD 2^n network is used to obtain $(ty + x) \pmod{2^n}$ from the first step to the t -th step.

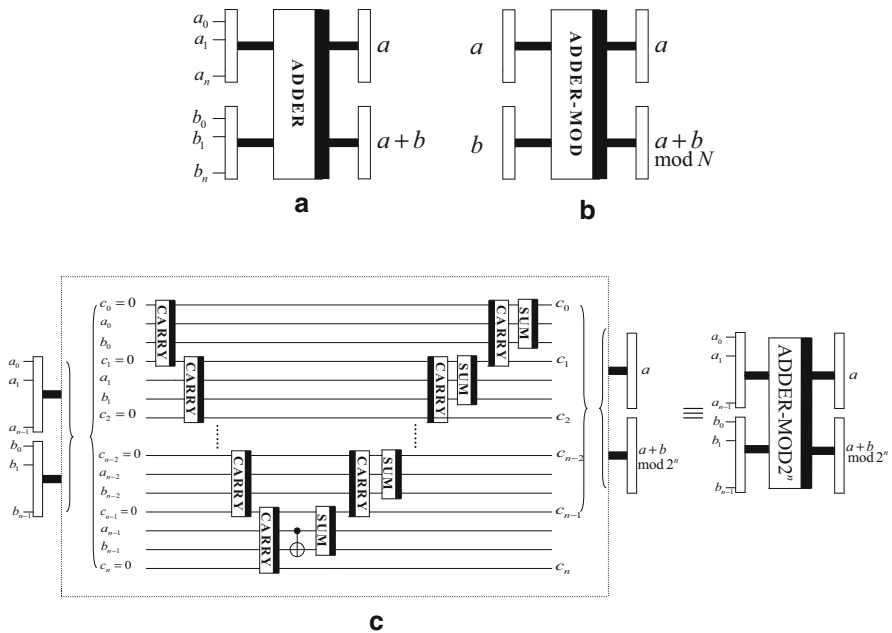


Fig. 1 **a** Plain adder network, **b** adder modulo N network, **c** ADDER-MOD 2^n network

$$\begin{aligned}
 |y, x\rangle &\rightarrow |y, (y + x) \bmod 2^n\rangle \rightarrow \dots \rightarrow \\
 &|y, ((t - 1)y + x) \bmod 2^n\rangle \rightarrow |y, (ty + x) \bmod 2^n\rangle
 \end{aligned}
 \tag{8}$$

The input is the position information $|x\rangle$ and $|y\rangle$ of original image, and the output is the position information $|x'\rangle$ of Arnold scrambled image.

The realization of $|y'\rangle$ is divided into $tm + m + 1$ steps, as shown in Fig. 3. From the first step to the $(m - 1)$ -th step, the ADDER-MOD 2^n network is used to obtain $mx \bmod 2^n$, in the m -th step, x is replaced by y , from the $(m + 1)$ -th step to the last step, the ADDER-MOD 2^n network is employed to obtain $(mx + (tm + 1)y) \bmod 2^n$.

$$\begin{aligned}
 |x, x\rangle &\rightarrow |x, 2x \bmod 2^n\rangle \rightarrow \dots \rightarrow |x, mx \bmod 2^n\rangle \rightarrow |y, mx \bmod 2^n\rangle \\
 &\rightarrow |y, (y + mx) \bmod 2^n\rangle \rightarrow \dots \rightarrow |y, (tm y + mx) \bmod 2^n\rangle \\
 &\rightarrow |y, ((tm + 1)y + mx) \bmod 2^n\rangle
 \end{aligned}
 \tag{9}$$

Thus, the $|y'\rangle$ network outputs the position information $|y'\rangle$ of Arnold scrambled image with inputs $|x\rangle$ and $|y\rangle$.

From Eq. (5), it is clear that the position information $|x\rangle$ and $|y\rangle$ is regained only depending on $|x'\rangle$ and $|y'\rangle$ of the scrambled image. Hence, the inverse scrambling circuits are necessary. The inverse scrambling networks are designed to regain $|x\rangle$ and $|y\rangle$. A theorem in [39] was related as follows.

$$(x - y) \bmod 2^n = (x + (\bar{y} + 1)) \bmod 2^n
 \tag{10}$$

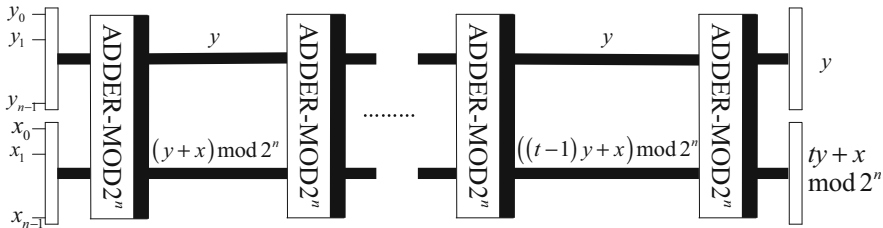


Fig. 2 $|x'\rangle$ network

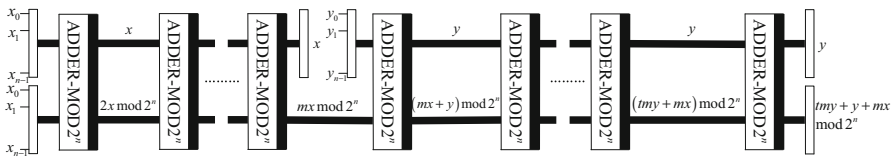


Fig. 3 $|y'\rangle$ network

where $\bar{y} = \bar{y}_{n-1}\bar{y}_{n-2} \dots \bar{y}_0$, $\bar{y}_i = 1 - y_i$, $i = n - 1, n - 2, \dots, 0$. According to Eq. (10), we obtain $((tm + 1)x' - ty') \bmod 2^n = ((tm + 1)x' + t\bar{y}') \bmod 2^n$. So the realization of $|x\rangle$ is divided into $t + tm + 3$ steps, as shown in Fig. 4. The ADDER-MOD 2^n network is used to obtain $((tm + 1)x') \bmod 2^n$ from the first step to the tm -th step, in the $(tm + 1)$ -th step, x' is replaced by \bar{y}' , the ADDER-MOD 2^n network is used to retrieve $((tm + 1)x' + t\bar{y}') \bmod 2^n$ from the $(tm + 2)$ -th step to the $(tm + t + 1)$ -th step, in the $(tm + t + 2)$ -th step, x' is replaced by \bar{y}' , and in the last step, an ADDER-MOD 2^n network is involved.

$$\begin{aligned}
 |x', x'\rangle &\rightarrow \dots \rightarrow |x', (tm + 1)x' \bmod 2^n\rangle \rightarrow |\bar{y}', (tm + 1)x' \bmod 2^n\rangle \\
 &\rightarrow \dots \rightarrow |\bar{y}', ((tm + 1)x' + t\bar{y}') \bmod 2^n\rangle \\
 &\rightarrow |t, ((tm + 1)x' + t\bar{y}') \bmod 2^n\rangle \\
 &\rightarrow |t, ((tm + 1)x' + t\bar{y}' + t) \bmod 2^n\rangle
 \end{aligned}
 \tag{11}$$

Due to $(-mx' + y') \bmod 2^n = (m\bar{x}' + m + y) \bmod 2^n$, the realization of $|y\rangle$ is divided into $m + 2$ steps, as depicted in Fig. 5. From the first step to the $(m - 1)$ -th step, the ADDER-MOD 2^n network is exploited to obtain $m\bar{x}' \bmod 2^n$, in the m -th step, x' is replaced with y' , the ADDER-MOD 2^n operation is used to obtain $(m\bar{x}' + y') \bmod 2^n$ in the $(m + 1)$ -th step, in the $(m + 2)$ -th step, y' is replaced with m , and in the last step, ADDER-MOD 2^n network is necessary to obtain $(m\bar{x}' + y' + m) \bmod 2^n$.

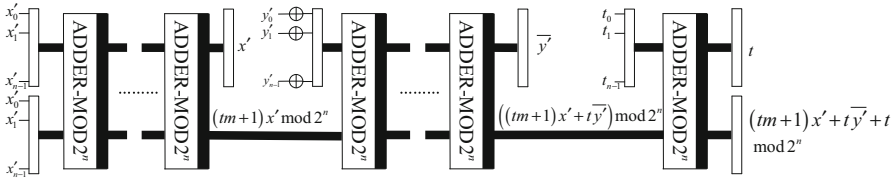


Fig. 4 |x⟩ network

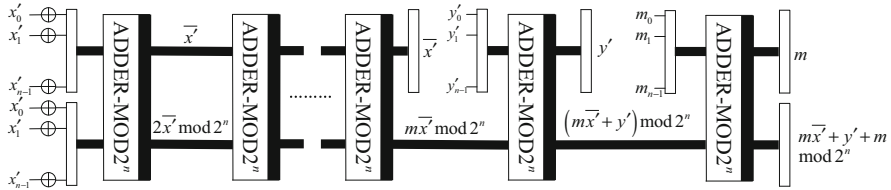


Fig. 5 |y⟩ network

$$\begin{aligned}
 |x', x'\rangle &\rightarrow \dots \rightarrow |\bar{x}', m\bar{x}' \bmod 2^n\rangle \rightarrow |y', m\bar{x}' \bmod 2^n\rangle \rightarrow |y', (m\bar{x}' + y') \bmod 2^n\rangle \\
 &\rightarrow |m, (m\bar{x}' + y') \bmod 2^n\rangle \rightarrow |m, (m\bar{x}' + y' + m) \bmod 2^n\rangle \quad (12)
 \end{aligned}$$

4 Quantum image encryption and decryption algorithm

4.1 Quantum image encryption algorithm

Assume that plaintext quantum image is $|M\rangle = \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} |g(y, x)\rangle |yx\rangle$, where $|g(y, x)\rangle = \cos \theta_i |0\rangle + \sin \theta_i |1\rangle$, $\theta_i \in [0, \frac{\pi}{2}]$, $i = yx = 0, 1, \dots, 2^{2n} - 1$. The proposed image encryption algorithm consists of the following steps:

Step 1. Perform generalized Arnold transform operation on $|M\rangle$ for k times to obtain $|Q_1\rangle$, where $|x_A\rangle$ and $|y_A\rangle$ represent the horizontal and the vertical location information of the final scrambled quantum image $|Q_1\rangle$, respectively. A represents the generalized Arnold image scrambling, and $|Q\rangle$ represents the scrambled quantum image for once. The quantum version of generalized Arnold transform is defined as

$$\begin{aligned}
 |Q\rangle &= A(|M\rangle) = \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} |g(y, x)\rangle A(|yx\rangle) \\
 &= \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} |g(y, x)\rangle A(|y\rangle) A(|x\rangle) \\
 &= \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} |g(y, x)\rangle |y'x'\rangle \quad (13)
 \end{aligned}$$

where

$$\begin{cases} |x'\rangle = A(|x\rangle) = |(x + ty) \bmod 2^n\rangle \\ |y'\rangle = A(|y\rangle) = |[mx + (tm + 1)y] \bmod 2^n\rangle \end{cases} \tag{14}$$

Step 2. Perform quantum random-phase operation on $|Q\rangle$ in the spatial domain to encode each color angle to a new angle. Random-phase gate U_i is controlled by a classical binary number k_i , where $k_i \in \{0, 1\}, i = 0, 1, \dots, 2^{2n} - 1$. Binary sequence $K = k_0k_1 \dots k_{2^{2n}-1}$ is the key.

$$T_i = (U_i)^{k_i} = \begin{cases} U_i, & k_i = 1; \\ I, & k_i = 0. \end{cases} \quad i = 0, 1, \dots, 2^{2n} - 1. \tag{15}$$

$$U_i = \begin{bmatrix} 1 & 0 \\ 0 & e^{j2\pi\varphi_i} \end{bmatrix} \tag{16}$$

where φ_i is a real number and distributed uniformly between 0 and 1. Unitary transform T_i is used to construct a $2n + 1$ qubits-based unitary transform B_i .

$$B_i = I \otimes \sum_{y=0}^{2^n-1} \sum_{\substack{x=0 \\ yx \neq i}}^{2^n-1} |yx\rangle \langle yx| + T_i \otimes |i\rangle \langle i| \tag{17}$$

The controlled phase matrix B_i is a unitary matrix since $B_i B_i^\dagger = I^{\otimes 2n+1}$. By applying a $2n + 1$ qubits unitary transform B on quantum image $|Q_1\rangle, |Q_2\rangle$ is obtained.

$$\begin{aligned} B(|Q_1\rangle) &= \prod_{i=0}^{2^{2n}-1} B_i(|Q_1\rangle) \\ &= \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} T_{yx} (\cos \theta_{yx} |0\rangle + \sin \theta_{yx} |1\rangle) |y_A x_A\rangle \\ &= \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} |f(y, x)\rangle |y_A x_A\rangle = |Q_2\rangle \end{aligned} \tag{18}$$

Step 3. Execute QFT on $|Q_2\rangle$, then perform quantum random-phase operation in the Fourier transform domain. Random-phase gate U'_i is controlled by a binary number d_i , where $d_i \in \{0, 1\}, i = 0, 1, \dots, 2^{2n} - 1$. Binary sequence $D = d_0d_1 \dots d_{2^{2n}-1}$ is another key.

$$H_i = (U'_i)^{d_i} = \begin{cases} U'_i, & d_i = 1; \\ I, & d_i = 0. \end{cases} \quad i = 0, 1, \dots, 2^{2n} - 1. \tag{19}$$

$$U'_i = \begin{bmatrix} 1 & 0 \\ 0 & e^{j2\pi\psi_i} \end{bmatrix} \tag{20}$$

where ψ_i is a real number and distributed uniformly between 0 and 1. Unitary transform H_i is used to construct a $2n + 1$ qubits-based unitary transform C_i .

$$C_i = I \otimes \sum_{y=0}^{2^n-1} \sum_{\substack{x=0 \\ yx \neq i}}^{2^n-1} |yx\rangle \langle yx| + H_i \otimes |i\rangle \langle i| \tag{21}$$

The controlled phase matrix C_i is a unitary matrix since $C_i C_i^\dagger = I^{\otimes 2n+1}$. Apply a $2n + 1$ qubits unitary transform C on QFT ($|Q_2\rangle$).

$$\begin{aligned} |Q_3\rangle &= C (\text{QFT} (|Q_2\rangle)) = \prod_{i=0}^{2^{2n}-1} C_i (\text{QFT} (|Q_2\rangle)) \\ &= \prod_{i=0}^{2^{2n}-1} C_i \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} \text{QFT} (|f(y, x)\rangle |y_A x_A\rangle) \\ &= \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} H_{yx} \text{QFT} (|f(y, x)\rangle |y_A x_A\rangle) \end{aligned} \tag{22}$$

Step 4. Perform the inverse quantum Fourier transform (IQFT) on $|Q_3\rangle$.

$$\begin{aligned} |Q_4\rangle &= \text{IQFT} (|Q_3\rangle) \\ &= \text{IQFT} \left(\frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} H_{yx} \text{QFT} (|f(y, x)\rangle |y_A x_A\rangle) \right) \end{aligned} \tag{23}$$

4.2 Quantum image decryption algorithm

The key involved in the encryption process is composed of the independent parameters t and m of coefficients matrix, iterative times k , the classical binary sequences $K = k_0 k_1 \dots k_{2^{2n}-1}$ and $D = d_0 d_1 \dots d_{2^{2n}-1}$. According to the encryption, the decryption process is as follows.

Step 1. Perform QFT on $|Q_4\rangle$.

$$\text{QFT} (|Q_4\rangle) = \text{QFT} (\text{IQFT} (|Q_3\rangle)) = |Q_3\rangle \tag{24}$$

Step 2. Perform the decryption operation on $|Q_3\rangle$ with the key D .

$$\begin{aligned} C^{-1} (|Q_3\rangle) &= \prod_{i=0}^{2^{2n}-1} C_i^\dagger (|Q_3\rangle) \\ &= \prod_{i=0}^{2^{2n}-1} C_i^\dagger \left(\frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} H_{yx} \text{QFT} (|f(y, x)\rangle |y_A x_A\rangle) \right) \end{aligned}$$

$$\begin{aligned}
 &= \frac{1}{2^n} \sum_{x=0}^{2^n-1} \sum_{y=0}^{2^n-1} H_{yx}^{-1} H_{yx} \text{QFT}(|f(y, x)\rangle |y_A x_A\rangle) \\
 &= \frac{1}{2^n} \sum_{x=0}^{2^n-1} \sum_{y=0}^{2^n-1} \text{QFT}(|f(y, x)\rangle |y_A x_A\rangle) = \text{QFT}(|Q_2\rangle) \tag{25}
 \end{aligned}$$

where C_{yx}^\dagger is the Hermitian conjugate of C_{yx} .

Step 3. Execute the IQFT to obtain $|Q_2\rangle$ and then perform the decryption operation on $|Q_2\rangle$ with the key K .

$$\begin{aligned}
 B^{-1}(|Q_2\rangle) &= \prod_{i=0}^{2^{2n}-1} B_i^\dagger(|Q_2\rangle) \\
 &= \prod_{i=0}^{2^{2n}-1} B_i^\dagger\left(\frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} |f(y, x)\rangle |y_A x_A\rangle\right) \\
 &= \frac{1}{2^n} \sum_{x=0}^{2^n-1} \sum_{y=0}^{2^n-1} T_{yx}^{-1} T_{yx} (|g(y, x)\rangle |y_A x_A\rangle) \\
 &= \frac{1}{2^n} \sum_{x=0}^{2^n-1} \sum_{y=0}^{2^n-1} |g(y, x)\rangle |y_A x_A\rangle = |Q_1\rangle \tag{26}
 \end{aligned}$$

Step 4. Perform the inverse generalized Arnold transform operation A^{-1} on quantum image $|Q_1\rangle$ for k times. The quantum version of inverse generalized Arnold transform is defined as

$$\begin{aligned}
 |M\rangle &= A^{-1}(|Q\rangle) = \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} |g(y, x)\rangle A^{-1}(|y'x'\rangle) \\
 &= \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} |g(y, x)\rangle A^{-1}(|y'\rangle) A^{-1}(|x'\rangle) \\
 &= \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} |g(y, x)\rangle |yx\rangle \tag{27}
 \end{aligned}$$

where

$$\begin{cases} |x\rangle = A^{-1}(|x'\rangle) = |((tm + 1)x' - ty') \bmod 2^n\rangle \\ |y\rangle = A^{-1}(|y'\rangle) = |(-mx' + y') \bmod 2^n\rangle \end{cases} \tag{28}$$

5 Numerical simulation and discussion

The experiments are limited to classical simulations on a classical computer with MATLAB, since the lack of quantum hardware. The simulations are based on linear algebraic constructions. The quantum states and the quantum operations are simulated by complex vectors and unitary matrices, respectively. The final step is the measurement in quantum computation, which converts the quantum information into the classical form as probability distribution. In a classical computer, the quantum images are transformed into large matrices, and the simulations of the transformation are implemented by using linear algebraic constructions equivalent to the quantum circuit elements.

To achieve classical numerical simulation, the simulations on encryption process are completed in two stages. In the confusion stage, the simulation is implemented by the corresponding classical generalized Arnold transform. In the diffusion stage, classical binary sequences are used to control the random-phase gates, which makes the angles of the color information different for different positions. Thus, the simulation is implemented differently from the corresponding classical double random-phase encoding. The binary sequences are represented by a matrix with the same size of the image on the software of MATLAB. All the elements of the matrix are only 0 and 1, and the matrix is used to control the random-phase mask (random-phase matrix). If the matrix element is 1 in one position, the random-phase matrix element is replaced by 1 in the corresponding position. Then, the random-phase encoding can be implemented by the random-phase matrix and the image matrix point multiplication.

According to the definition of the generalized Arnold transform, the independent parameters of coefficient matrices t and m are any positive integers, which makes the determinant of coefficient matrix to be 1. The period of the generalized Arnold transform is connected with image size. The pixel size of all the images is 512×512 . Thus, the period of the generalized Arnold transform can be computed as $T = 384$. If the iterative times is not exactly a multiple of the period, the generalized Arnold transform can scramble the positions of image pixels. Therefore, one has a chance to select these parameters randomly to some degree. The parameters in simulation are set as: $t = 600$, $m = 300$ and $k = 45$. The binary sequences and random-phase matrices are generated by the random number generation function in MATLAB R2012a (version 7.14.0.739). The plaintext image is Lena shown in Fig. 6a, and the corresponding cipher image is shown in Fig. 6b.

5.1 Statistical analysis

Statistical analysis has been performed with the proposed quantum image encryption algorithm to demonstrate its confusion and diffusion properties.

5.1.1 Correlation of adjacent pixels

Generally, each pixel in the plaintext image is highly correlated with its adjacent pixels in horizontal, vertical or diagonal directions. To test the correlations of adjacent pixels in Lena and encrypted Lena, 8,000 pairs of two adjacent pixels are randomly

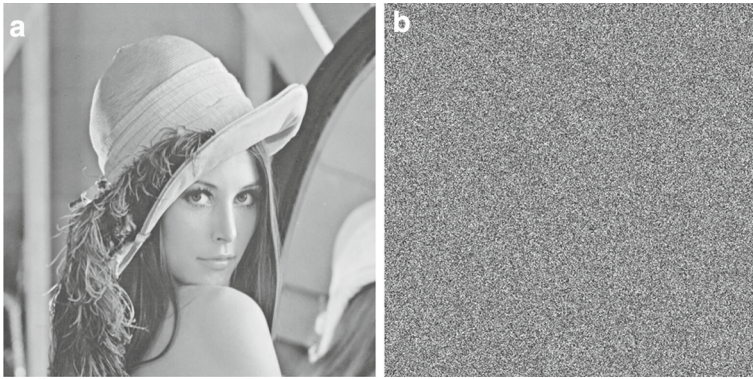


Fig. 6 **a** Plaintext image Lena, **b** cipher image

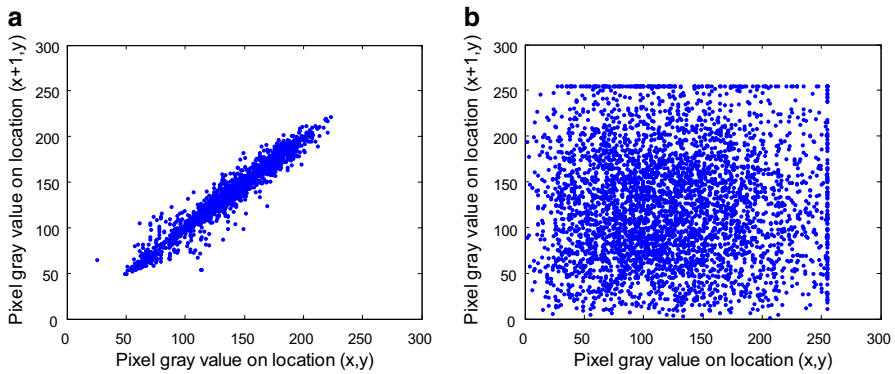


Fig. 7 Correlation distributions of two horizontally adjacent pixels: **a** original image Lena and **b** encrypted image

selected from horizontal, vertical and diagonal directions, respectively. The correlation coefficient can be calculated by

$$C_{xy} = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2 \sum_{i=1}^N (y_i - \bar{y})^2}} \tag{29}$$

where $\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$ and $\bar{y} = \frac{1}{N} \sum_{i=1}^N y_i$. Figures 7, 8, 9 give the correlation distribution of horizontally, vertically and diagonally adjacent pixels in the plaintext image “Lena” and its corresponding cipher image. The results of correlation coefficients for the original images and their corresponding encrypted images are compiled in Table 1. The correlation of the plaintext is close to 1 in each direction of each component, while the correlation of the encrypted image is close to 0 in each direction. That is to

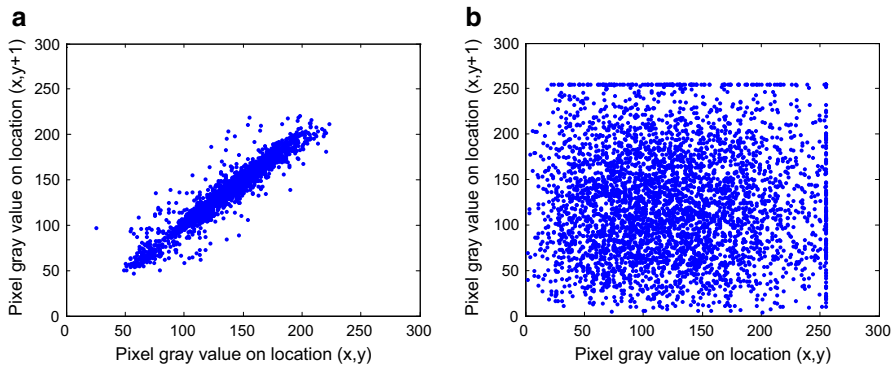


Fig. 8 Correlation distributions of two vertically adjacent pixels: **a** original image Lena and **b** encrypted image

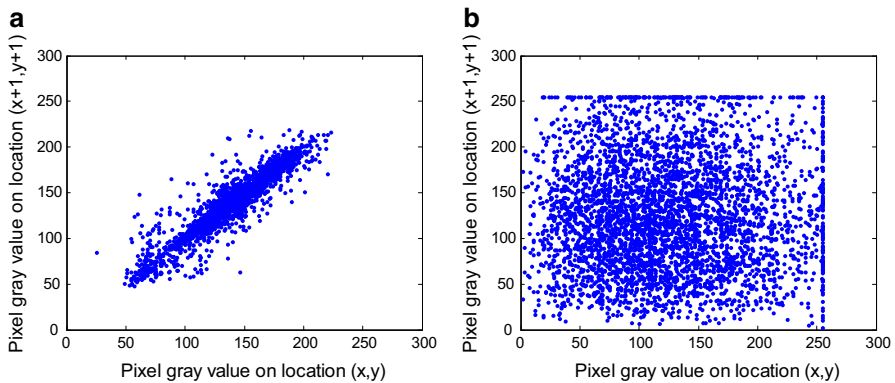


Fig. 9 Correlation distributions of two diagonally adjacent pixels: **a** original image Lena and **b** encrypted image

Table 1 Correlation coefficients of adjacent pixels

Correlation coefficient	Horizontal	Vertical	Diagonal
Original Lena	0.9770	0.9617	0.9412
Encrypted Lena	-0.0050	0.0018	0.0069
Original Baboon	0.7198	0.8351	0.6889
Encrypted Baboon	-0.0130	0.0023	0.0012
Original Peppers	0.9752	0.9670	0.9459
Encrypted Peppers	0.0137	-0.0115	0.0104
Original plane	0.9625	0.9698	0.9452
Encrypted plane	-0.0116	0.0023	0.0039

say, the proposed algorithm removes the tight correlation among adjacent pixels of the original image successfully. The results demonstrate that the attackers cannot obtain useful information according to the statistical analysis.

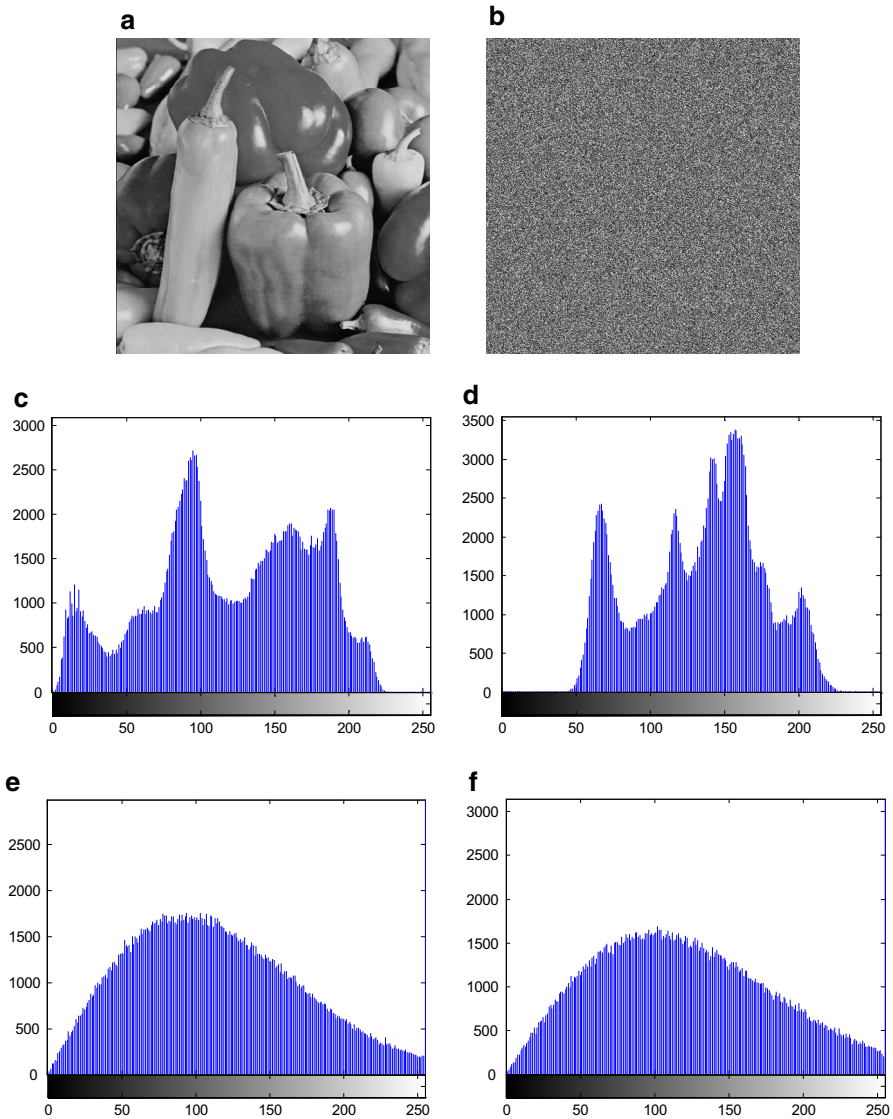


Fig. 10 **a** Peppers; **b** encrypted Peppers; **c** histogram of Peppers; **d** histogram of Lena; **e** histogram of encrypted Peppers; and **f** histogram of encrypted Lena

5.1.2 Histogram analysis

The histogram is one of the most important statistical characteristics of an image and represents the frequency of all the gray-level values from all over the image. Figure 10a is the gray image Peppers which is obviously different from the image Lena shown in Fig. 6a, and it comes to the same conclusion by comparing their histograms shown in Fig. 10c, d, respectively. The two images are encrypted under the same conditions. The encrypted result of peppers is shown in Fig. 10b. Figure 10e, f are the histograms

Table 2 Information entropy of original and encrypted images

Images (512 × 512)	Information entropy (H)	
	Original images	Encrypted images
Lena	7.2185	7.9877
Peppers	7.5925	7.8699
Baboon	7.1391	7.8726
Boat	7.1914	7.9735
Camera	7.0097	7.9410
Bridge	5.7056	7.8222

for encrypted Lena and encrypted Peppers, and they are quite similar. After a number of parallel experiments, it can be concluded that the ciphertext of different original images have similar histograms. Thus, the attackers cannot obtain useful information according to the statistical properties.

5.1.3 Information entropy

Entropy is a statistical measure of randomness to characterize the texture of an image. The entropy $H(s)$ of a message source s can be calculated as

$$H(s) = - \sum_{i=0}^{2^N-1} p(s_i) \log_2 p(s_i) \quad (30)$$

where $p(s_i)$ represents the probability of symbol s_i and the entropy is expressed in bits. The ideal entropy value for an encrypted image should be 8 bits [42]. For a cryptosystem able to resist the entropy attacks, the entropy of the ciphertext should be close to the ideal value [43,44]. The entropy of the six original images and their corresponding encrypted images are computed and listed in Table 2. From Table 2, one can see that the entropy of encrypted images is very close to the theoretical value. The increase in entropy reflects that the distribution of gray scale becomes more even. Therefore, the encryption algorithm is secure against the entropy attack.

5.2 Key space analysis

The key space of a good image encryption algorithm should be large enough to make brute-force attack invalid. In the proposed algorithm, assuming key space for generalized Arnold transform is S_1 and key space for double random-phase encoding is S_2 then the key space of the entire algorithm is $S_1 \times S_2$. The keys are composed of the independent parameters t and m of coefficients matrix, iterative times k , binary sequences $K = k_0k_1 \dots k_{2^{2n}-1}$ and $D = d_0d_1 \dots d_{2^{2n}-1}$. The key space for the generalized Arnold transform is estimated to be $S_1 \approx 10^8$. The key space for the binary sequences is about $S_2 = 4^{512 \times 512}$. The total key space is a very huge number, and thus, the proposed algorithm can resist the brute-force attack.

5.3 Key sensitivity analysis

Key sensitivity is an essential factor for any good cryptosystem, which ensures the security of the cryptosystem against the brute-force attack. According to the properties of the generalized Arnold transform, the parameters of coefficients matrix t and m are any positive integers, so we can select the parameters randomly to some extent. The period of the generalized Arnold transform is connected with image size. In the simulation, the iteration number in the encryption process is $k = 45$, and thus, the iteration number in the decryption process should be multiple of $T - k = 339$. The attacker cannot obtain the correct original image if he decrypts the image with the wrong iteration number of the generalized Arnold transform. To analyze the key sensitivity, six groups of keys are used to decrypt the cipher image. The simulation results are shown in Fig. 11a–f. Figure 11a is the decrypted image with the correct keys. Figure 11b gives the decrypted image with an incorrect independent parameter t , while the other keys are all correct. Figure 11c shows the decrypted image with an incorrect independent parameter m , while the other keys are all intact. Figure 11d shows the decrypted image with a wrong iterative times, while the other keys are all right. Figure 11e gives the decrypted image with an incorrect binary sequence K , while the other keys are all correct. Figure 11f shows the decrypted image with an incorrect binary sequence D , while the other keys are all intact. From the results, it is shown that the image can be reconstructed correctly iff the decryption keys are all right. For a large key space, it is difficult to reconstruct the plain image if the key distribution is unknown.

5.4 Performance comparison

5.4.1 Diffusion and confusion

Since Yang et al. presented a novel gray-level image encryption scheme based on QFT and double random-phase encoding technique, it is meaningful to compare the proposed algorithm with Yang et al.'s scheme. This is the reason why we introduced the generalized Arnold transform into quantum image encryption. Moreover, we also compared the proposed algorithm with the method only utilizing the Arnold transform. The correlations for the proposed algorithm, Yang et al.'s scheme and the Arnold transform method are shown in Table 3. It can be seen that the correlation of the proposed quantum encryption algorithm is much weaker than the other two methods.

The security of the proposed encryption system depends on not only the number of keys, but also the cryptosystem structure. Generally, for a secure encryption algorithm, the cryptosystem structure meets the principle of confusion and diffusion in cryptography. The encryption algorithm based on QFT and double random-phase encoding can be considered as a kind of encryption of the gray-level information. It is well known that Arnold transform has the characteristics of chaotic mapping. However, it can only scramble the position information of quantum image. The proposed encryption algorithm implements quantum image encryption by combining generalized Arnold transform with double random-phase encoding technology, which helps to realize the diffusion and confusion of the image information. What's more, the parameters and

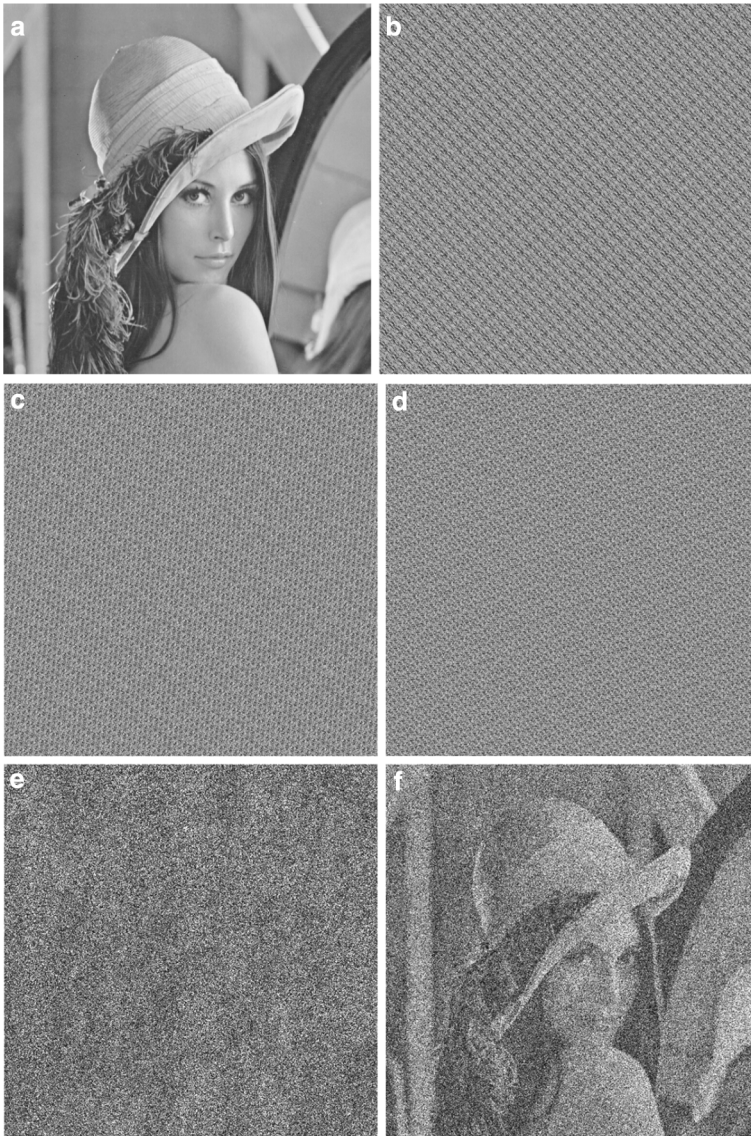


Fig. 11 Decrypted images with: **a** correct keys; **b** incorrect independent parameter $t = 601$; **c** incorrect independent parameter $m = 301$; **d** wrong iterative times $k = 46$; **e** incorrect binary sequence K ; **f** incorrect binary sequence D

iteration times of generalized Arnold transform are the keys, which enlarges the key space and consequently enhances the security further.

5.4.2 Computational complexity

Assume that M is a $2^n \times 2^n$ original image. There are 2^{2n} pixels in the original image. The computational complexity of the proposed encryption algorithm depends

Table 3 Correlation of three algorithms for encrypted Lena

Encrypted Lena	Horizontal	Vertical	Diagonal
Proposed algorithm	0.0048	-0.0056	0.0028
Yang et al.'s scheme	-0.0136	-0.0199	0.0172
Arnold transform method	0.0536	-0.0735	-0.0529

very much on what is considered to be elementary gate. We choose the Control-NOT gate, NOT gate and phase gate to be basic units. The Toffoli gate can be realized by six Control-NOT gates [41]. The numbers of elementary gates in basic carry and sum operations are 13 and 2, respectively. The plain adder includes $2n - 1$ carry operations, n sum operations and one Control-NOT gate. Consequently, the elementary gates of the plain adder are $28n - 12$. Because the architecture of ADDER-MOD 2^n is same as the plain adder, the elementary gates of ADDER-MOD 2^n are $28n - 12$. So, the generalized Arnold transform needs $(tm + t + m)(28n - 12)$ basic gates. The QFT operation needs $\frac{n(n-1)}{2}$ basic gates. The complexity of random-phase operation is $O(n)$. Thus, the total computational complexity of the encryption algorithm is $O(n^2)$. By analyzing the corresponding classical image encryption algorithm, the computational complexity of the generalized Arnold transform is $O(2^{2n})$. The classical random-phase encoding is realized by using 2^{2n} multiplication operations. The computational complexity of the classical Fourier transform operation is $O(n2^{2n})$. Therefore, the total computational complexity is $O(n2^{2n})$. Therefore, the proposed encryption algorithm takes advantage over its classical counterparts in terms of computational complexity.

6 Conclusion

A quantum version of generalized Arnold transform is defined, and its quantum circuit is suggested. By combining generalized Arnold transform with double random-phase encoding, a quantum image encryption algorithm is proposed. The encryption process can be realized by performing generalized Arnold transform and double random-phase operations on positions information and gray-level information of the quantum image, respectively. The independent parameters, the iterative times of the generalized Arnold transform and the classical binary sequences are used as the keys, therefore, the key space of the proposed algorithm is very large. Simulation results show the validity and the reliability of the image encryption algorithm. Moreover, the proposed image encryption algorithm has lower computational complexity than its classical counterparts.

Acknowledgments This work is supported by the National Natural Science Foundation of China (Grant Nos. 61462061 and 61262084), the Foundation for Young Scientists of Jiangxi Province (Jinggang Star) (Grant No. 20122BCB23002), the Research Foundation of the Education Department of Jiangxi Province (Grant Nos. GJJ14138 and GJJ13057) and the Open Project of Key Laboratory of Photoelectronics and Telecommunication of Jiangxi Province (Grant No. 2013003).

References

1. Zhang, Y., Xiao, D.: Cryptanalysis of S-box-only chaotic image ciphers against chosen plaintext attack. *Nonlinear Dynam.* **72**(4), 751–756 (2013)
2. Refregier, P., Javidi, B.: Optical image encryption using input plane and Fourier plane random encoding. *SPIE's 1995 International Symposium on Optical Science, Engineering, and Instrumentation. International Society for Optics and Photonics*, 62–68 (1995)
3. Situ, G., Zhang, J.: Double random-phase encoding in the Fresnel domain. *Opt. Lett.* **29**(14), 1584–1586 (2004)
4. Unnikrishnan, G., Joseph, J., Singh, K.: Optical encryption by double-random phase encoding in the fractional Fourier domain. *Opt. Lett.* **25**(12), 887–889 (2000)
5. Zhou, X., Lai, D., Yuan, S., Li, D.H., Hu, J.P.: A method for hiding information utilizing double-random phase-encoding technique. *Opt. Laser Technol.* **39**(7), 1360–1363 (2007)
6. Tao, R., Xin, Y., Wang, Y.: Double image encryption based on random phase encoding in the fractional Fourier domain. *Opt. Express* **15**(24), 16067–16079 (2007)
7. Lu, P., Xu, Z., Lu, X., Liu, X.: Digital image information encryption based on compressive sensing and double random-phase encoding technique. *Optik* **124**(16), 2514–2518 (2013)
8. Liu, Z., Li, S., Liu, W., Wang, Y., Liu, S.: Image encryption algorithm by using fractional Fourier transform and pixel scrambling operation based on double random phase encoding. *Opt. Lasers Eng.* **51**(1), 8–14 (2013)
9. Peng, X., Zhang, P., Wei, H., Yu, B.: Known-plaintext attack on optical encryption based on double random phase keys. *Opt. Lett.* **31**(8), 1044–1046 (2006)
10. Frauel, Y., Castro, A., Naughton, T.J., Javidi, B.: Resistance of the double random phase encryption against various attacks. *Opt. Express* **15**(16), 10253–10265 (2007)
11. Carnicer, A., Montes-Usategui, M., Arcos, S., Juvells, I.: Vulnerability to chosen-ciphertext attacks of optical encryption schemes based on double random phase keys. *Opt. Lett.* **30**(13), 1644–1646 (2005)
12. Peng, X., Wei, H., Zhang, P.: Chosen-plaintext attack on keyless double-random phase encoding in the Fresnel domain. *Opt. Lett.* **31**(22), 3261–3263 (2006)
13. Arnold, V.I., Avez, A.: *Ergodic problems of classical mechanics*. Benjamin, New York (1968)
14. Ye, R.S.: A novel image scrambling and watermarking scheme based on orbits of Arnold transform. *Conference on Circuits, Communications and Systems, Pacific-Asia*, 485–488 (2009)
15. Ye, G., Wong, K.W.: An efficient chaotic image encryption algorithm based on a generalized Arnold map. *Nonlinear Dynam.* **69**(4), 2079–2087 (2012)
16. Liu, Z., Gong, M., Dou, Y., Liu, F., Liu, S., Ashfaq Ahmad, M., Liu, S.: Double image encryption by using Arnold transform and discrete fractional angular transform. *Opt. Lasers Eng.* **50**(2), 248–255 (2012)
17. Chen, W., Quan, C., Tay, C.J.: Optical color image encryption based on Arnold transform and interference method. *Opt. Commun.* **282**(18), 3680–3685 (2009)
18. Chen, L., Zhao, D., Ge, F.: Image encryption based on singular value decomposition and Arnold transform in fractional domain. *Opt. Commun.* **291**, 98–103 (2013)
19. Liu, Z., Liu, S., Chen, H., Liu, T., Li, P., Xu, L., Dai, J.: Image encryption by using gyrator transform and Arnold transform. *J. Electron. Imaging* **20**(1), 013020–013026 (2011)
20. Guo, Q., Liu, Z., Liu, S.: Color image encryption by using Arnold and discrete fractional random transforms in IHS space. *Opt. Lasers Eng.* **48**(12), 1174–1181 (2010)
21. Nielsen, M.A., Chuang, I.L.: *Quantum computation and quantum information*. Cambridge University Press, Cambridge (2010)
22. Venegas-Andraca, S.E., Ball, J.L.: Processing images in entangled quantum systems. *Quantum Inf. Process.* **9**(1), 1–11 (2010)
23. Le, P.Q., Dong, F.Y., Hirota, K.: A flexible representation of quantum images for polynomial preparation, image compression, and processing operations. *Quantum Inf. Process.* **10**(1), 63–84 (2011)
24. Sun, B., Le, P.Q., Ilyyasu, A.M., Yan, F., Garcia, J.A., Dong, F., Hirota, K.: A multi-channel representation for images on quantum computers using the RGB α color space. *Intelligent Signal Processing (WISP), 2011 IEEE 7th International Symposium on Floriana*, 160–165 (2011)
25. Le, P.Q., Ilyyasu, A.M., Garcia, J.A., Dong, F., Hirota, K.: Representing visual complexity of images using a 3D feature space based on structure, noise, and diversity. *JACIII* **16**(5), 631–640 (2012)
26. Zhang, Y., Lu, K., Gao, Y., Xu, K.: A novel quantum representation for log-polar images. *Quantum Inf. Process.* **12**(9), 3101–3126 (2013)

27. Yuan, S., Mao, X., Xue, Y., Chen, L., Xiong, Q., Compare, A.: SQR: a simple quantum representation of infrared images. *Quantum Inf. Process.* **13**(6), 1353–1379 (2014)
28. Zhang, Y., Lu, K., Gao, Y., Wang, M.: NEQR: a novel enhanced quantum representation of digital images. *Quantum Inf. Process.* **12**(8), 2833–2860 (2013)
29. Akhshani, A., Akhavan, A., Lim, S.C., Hassan, Z.: An image encryption scheme based on quantum logistic map. *Commun. Nonlinear Sci. Numer. Simulat.* **17**(12), 4653–4661 (2012)
30. Zhou, R.G., Wu, Q., Zhang, M.Q., Shen, C.Y.: Quantum image encryption and decryption algorithms based on quantum image geometric transformations. *Int. J. Theor. Phys.* **52**(6), 1802–1817 (2013)
31. Zhang, W.W., Gao, F., Liu, B., Wen, Q.Y., Chen, H.: A watermark strategy for quantum images based on quantum Fourier transform. *Quantum Inf. Process.* **12**(2), 793–803 (2013)
32. Zhou, N., Liu, Y., Zeng, G., Xiong, J., Zhu, F.: Novel qubit block encryption algorithm with hybrid keys. *Physica A.* **375**(2), 693–698 (2007)
33. Abd El-Latif, A.A., Li, L., Wang, N., Han, Q., Niu, X.: A new approach to chaotic image encryption based on quantum chaotic system, exploiting color spaces. *Signal Process.* **93**(11), 2986–3000 (2013)
34. Song, X., Wang, S., El-Latif, A.A.A., Niu, X.: Dynamic watermarking scheme for quantum images based on Hadamard transform. *Multimedia Syst.* **20**(4), 379–388 (2014)
35. Jiang, N., Wang, L., Wu, W.Y.: Quantum Hilbert image scrambling. *Int. J. Theor. Phys.* **53**(7), 2463–2484 (2014)
36. Yang, Y.G., Xia, J., Jia, X., Zhang, H.: Novel image encryption/decryption based on quantum Fourier transform and double phase encoding. *Quantum Inf. Process.* **12**(11), 3477–3493 (2013)
37. Song, X.H., Wang, S., Abd El-Latif, A.A., Niu, X.M.: Quantum image encryption based on restricted geometric and color transformations. *Quantum Inf. Process.* **13**(8), 1765–1787 (2014)
38. Jiang, N., Wu, W.Y., Wang, L.: The quantum realization of Arnold and Fibonacci image scrambling. *Quantum Inf. Process.* **13**(5), 1223–1236 (2014)
39. Jiang, N., Wang, L.: Analysis and improvement of the quantum Arnold image scrambling. *Quantum Inf. Process.* **13**(7), 1545–1551 (2014)
40. Dyson, F.J., Falk, H.: Period of a discrete cat mapping. *Am. Math. Mon.* **99**(7), 603–614 (1992)
41. Vedral, V., Barenco, A., Ekert, A.: Quantum networks for elementary arithmetic operations. *Phys. Rev. A* **54**(1), 147–153 (1996)
42. Chen, J.X., Zhu, Z.L., Fu, C., Yu, H.: A fast image encryption scheme with a novel pixel swapping-based confusion approach. *Nonlinear Dynam.* **77**(4), 1191–1207 (2014)
43. Ahmed, H., Kalash, H., Allah, O.: Implementation of rc5 block cipher algorithm for image cryptosystems. *Int. J. Inf. Technol.* **3**(4), 245–250 (2007)
44. Enayatifar, R.: Image encryption via logistic map function and heap tree. *Int. J. Phys. Sci.* **6**(2), 221–228 (2011)