Title: A New Cryptography Algorithm for Quantum Images

Authors: Mosayeb Naseri, Mona Abdolmaleky, Amel Laref, Fariborz Parandin, Turgay Celik, Ahmed Farouk, Masoumeh Mohamadi, Hesam Jalalian

PII: S0030-4026(18)30922-7
DOI: https://doi.org/10.1016/j.ijleo.2018.06.113
Reference: IJLEO 61116

To appear in:

Received date: 30-9-2017
Revised date: 20-4-2018
Accepted date: 27-6-2018

A New Cryptography Algorithm for Quantum Images

Mosayeb Naseri[1*], Mona Abdolmaleky[2], Amel Laref[3], Fariborz Parandin[4], Turgay Celik[5], Ahmed Farouk[6], Masoumeh Mohamadi[7], Hesam Jalalian[8]

[1]*Department of Physics, Kermanshah Branch, Islamic Azad University, Kermanshah, IRAN

[2]Department of Electrical Engineering, College of Engineering, Kermanshah Branch, Islamic Azad University, Kermanshah, IRAN

[3]Department of Physics and Astronomy, College of Science, King Saud University, Riyadh, 11451, Saudi Arabia

[4] Department of Electrical Engineering, College of Engineering, Kermanshah Branch, Islamic Azad University, Kermanshah, IRAN

[5] School of Computer Science and Applied Mathematics, University of the Witwatersrand, South Africa

[6] Faculty of Computer and Information Sciences, Mansoura University, Egypt

[7]Department of Information Technology, College of Engineering, Kermanshah Branch, Islamic Azad University, Kermanshah, IRAN

[8]Department of Electrical Engineering, College of Engineering, Kermanshah Branch, Islamic Azad University, Kermanshah, IRAN

Abstract

In this paper, a new bi-step quantum image cryptography algorithm is presented. The proposed scheme is consisted of four different coding algorithms. According to the pixels of original image and corresponding quantum bits of the generated random binary key, the employed coding algorithm is selected. The security of the proposed scheme is increased through randomization of binary image key generation and alteration the pixel values of the original gray-scale values. The simulation of the proposed scheme assures that the final coded image is completely meaningless and it could not be recognized visually through the analysis of the resulted meaningless coded image.

Key Words: Quantum cryptography, Quantum image representation**,** Quantum image processing

1. Introduction

Since the introduction of the first protocol of quantum key distribution proposed by Bennett and Brassard in 1984 [1], a number of different studies have focused on establishing the potential practical application of quantum information processing and computation [2–14].

Processing of digital images is one of the most popular activities in our daily life, and essentially is used to extract useful information from physical objects and environments. Quantum Image Processing (QIP) is recognised as a new discipline focused on the development of novel algorithms for the purposes of storing, processing, and retrieving visual information in a quantum computing framework. The first quantum image processing study was presented by Vlasov in 1997 [15]. Subsequently, Beach *et al*. [16], Venegas-Andraca and Bose [17,18] and other researchers followed the idea of quantum image processing in 2003, with many subsequent studies focused on quantum image processing topics, such as quantum image representation [19–22], quantum image encryption/decryption [23], quantum image segmentation [24], quantum image filtering [25], quantum image storage and retrieval [26], quantum image matching [27], quantum steganography[28–30], quantum watermarking [31–34], and so on [35,36].

One of the simplest and most efficient methods for protecting multimedia data is cryptography. The main aim of image cryptography which commonly is used as pre-processing in many image processing algorithms is to transform a meaningful image into a disordered image. In recent years, many different algorithms for quantum image coding have been proposed. In 2010, for example, Ye [37] proposed ‘image scrambling encryption algorithm of pixel bit based on chaos map’. Later, in 2012, Dalhoum *et al*. [38] proposed digital image scrambling using 2D cellular automata. Later, in 2014, Jiang *et al*. proposed quantum Hilbert image scrambling [39] and the quantum realisation of Arnold and Fibonacci image scrambling [40]. And recently in 2015, Zhou *et al*. [41] proposed ‘quantum image Gray-code and the Bit-plane scrambling algorithm’.

In this paper, a new quantum image cryptography algorithm is proposed. In the suggested algorithm, the original image is coded twice throughout the entire procedure. The applicability of the algorithm is confirmed with the application of software simulation.

In Section 2, various necessary preliminaries are presented. The proposed algorithm is introduced in Section 3. A discussion and analysis are provided in Section 4. Finally, Section 5 presents conclusions and provides a discussion.

2. Preliminaries

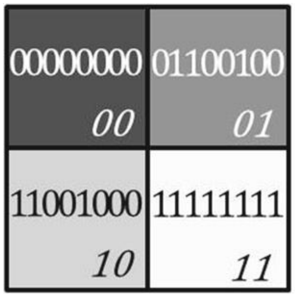
2.1. Quantum image representation models

According to Zhang et al.’s novel enhanced quantum representation (NEQR) for digital images, two entangled qubit sequences are used to store the information of gray-scale value and position of all the pixels [20]. The representation of a $2^N * 2^N$ image with the gray-scale range of 2^q using NEQR is defined as follows

$$|I\rangle = \frac{1}{2^N}\sum_{y=0}^{2^N-1}\sum_{x=0}^{2^N-1}|f(y,x)\rangle \otimes |yx\rangle \tag{1}$$

where $f(y,x) = C_{yx}^{0} \dots C_{yx}^{q-2}C_{yx}^{q-1}$ and $C_{yx}^{k} \in \{0,1\}$.

An example of a 2 × 2 NEQR quantum image and its perspective quantum state is shown in Fig.1.



$$|I\rangle = \frac{1}{2}(|0\rangle \otimes |00\rangle + |100\rangle \otimes |01\rangle + |200\rangle \otimes |10\rangle + |255\rangle \otimes |11\rangle)$$
$$= \frac{1}{2}(|00000000\rangle \otimes |00\rangle + |01100100\rangle \otimes |01\rangle + |11001000\rangle \otimes |10\rangle + |01100100\rangle \otimes |01\rangle)$$

Fig.1. A simple example of an NEQR image[20].

In 2015, Zhou et al. improved the NEQR model [40]. Based on the improved model, representation of a $2^M \times 2^N$ image with the gray-scale range of 2^q is defined as follows

$$|I\rangle = \frac{1}{2^t}\sum_{y=0}^{2^M-1}\sum_{x=0}^{2^N-1}|f(y,x)\rangle \otimes |yx\rangle \qquad (2)$$

where $t = (M+N)/2$, the gray-scale value of the (y,x) pixel is $f(y,x) = C_{yx}^0 \dots C_{yx}^{q-2}C_{yx}^{q-1}$ and $C_{yx}^k \in \{0,1\}$.

2.2 Previous work

2.2.1. Quantum Image Gray-code and Bit-plane Scrambling

In pursuit of improving the NEQR representation of quantum digital images, Zhou et al. proposed a quantum image Gray-code and Bit-plane scrambling algorithm [41]. Using their proposed scheme, a constant algorithm is used to code different images. The proposed scheme is coding the color image based on the value of the pixels. Firstly, the image is divided into 8 bit-planes, where the k_{th} bit-plane ($1 \leq k \leq 8$) is composed of the k_{th} bit of the gray-scale value for each pixel of the image. Subsequently, the whole bit-planes will go through XOR operation together according to the Gray-code scheme. Therefore, the scheme alternates the gray-scale value of all of the pixels and codes the image.

This scheme can be shown as in Eq. (3):

$$Scr(|I\rangle) = \frac{1}{2^t}\sum_{y=0}^{2^M-1}\sum_{x=0}^{2^N-1}Scr(|f(y,x)\rangle) \otimes |yx\rangle = \frac{1}{2^t}\sum_{y=0}^{2^M-1}\sum_{x=0}^{2^N-1}|g(y,x)\rangle \otimes |yx\rangle \qquad (3)$$

where $t = (M+N)/2$, *Scr* denotes the coding, $g(y,x)$ is the gray-scale value of the output pixels of the coding process and $\otimes$ denotes the tensor product.

The main problem of this scheme is that if the attacker realizes the flow of the algorithm, he can easily decode the coded image and retrieve the original one. Therefore, the security of the protocol is not assured against this type of attacks.

2.2.2. Quantum Hilbert Image Scrambling

In Quantum Hilbert Image Scrambling method [38], a $2^n \times 2^n$ image can be coded by utilizing the property of Hilbert Scanning Matrix (H_n) which consists of numbers ranging from 1 to 2^{2n} with a constant order. According to the size of the original image (based on *n*), the H_n matrix and Hilbert curve are realized. Therefore, the original image will be coded by applying both the perspective curve and the geometric transportation as illustrated in Fig.2.

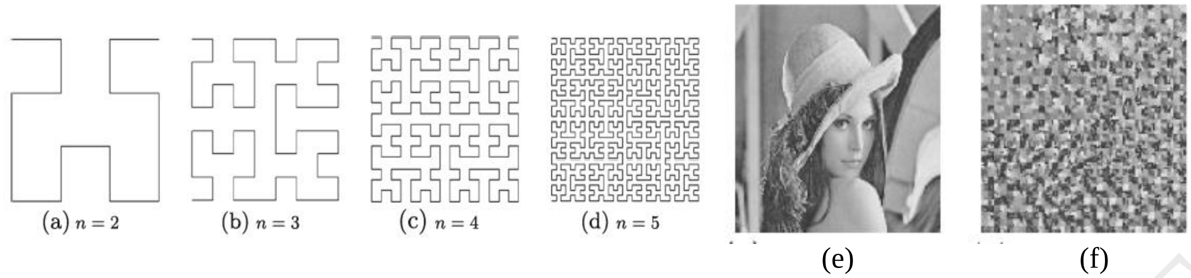


Fig.2. (a) Hilbert curve for n=1, (b) Hilbert curve for n=2, (c) Hilbert curve for n=3, (d) Hilbert curve for n=4, (e) Original image and (f) Coded image[38].

The algorithm suffers from two major artefacts: security and the size of image. Firstly, the transmission of the image is achieved completely by employing the Hilbert curve with a constant Hilbert scanning matrix H_n. Accordingly, the security is imperfect as the attacker, who captures Hilbert scanning matrix, can decrypt the coded image and retrieve the original one without any difficulty. Secondly, since the Hilbert scanning matrix is a square matrix, the algorithm can only be used to code a square image.

3. Proposed algorithm

Now let us introduce our new algorithm for image cryptography. In the present scheme, a random binary image is used as the key. Therefore, the security of the proposed scheme is increased through randomization of binary image key generation and alteration the pixel values of the original image. According to both the pair of original images pixels and its corresponding quantum bits of the generated randomized key, the employed coding algorithm is selected. The chosen coding algorithm is employed to change the gray-scale value and the position of pixels.

Suppose that Alice wants to code a $2^N \times 2^M$ image. Since she is the only one who has the key and knows which algorithm is used to code each pixel, if she communicates the key with someone else in a secured way (such as QKD [2]), the other one can easily decode the coded image using this key image.

Based on the enhanced $NEQR$ representation of quantum images, our proposed algorithm can be expressed as shown in Eq.4.

$$Scr(|I\rangle) = \frac{1}{2^t}\sum_{y=0}^{2^N-1}\sum_{x=0}^{2^M-1}|f'(y,x)\rangle \otimes |yx\rangle \tag{4}$$

where $t = (M+N)/2$, *Scr* denotes the coding, $f'(y,x)$ is the gray-scale value of the yx pixel of the coded image and $\otimes$ denotes the tensor product.

Our proposed cryptography algorithm is composed of two main parts, which are coding and decoding processes. The original image is coded twice according to the first and second random key images. Then, the coded image is stored. By the same way, the decoding process is achieved and the result is the original image as shown in Fig.3.

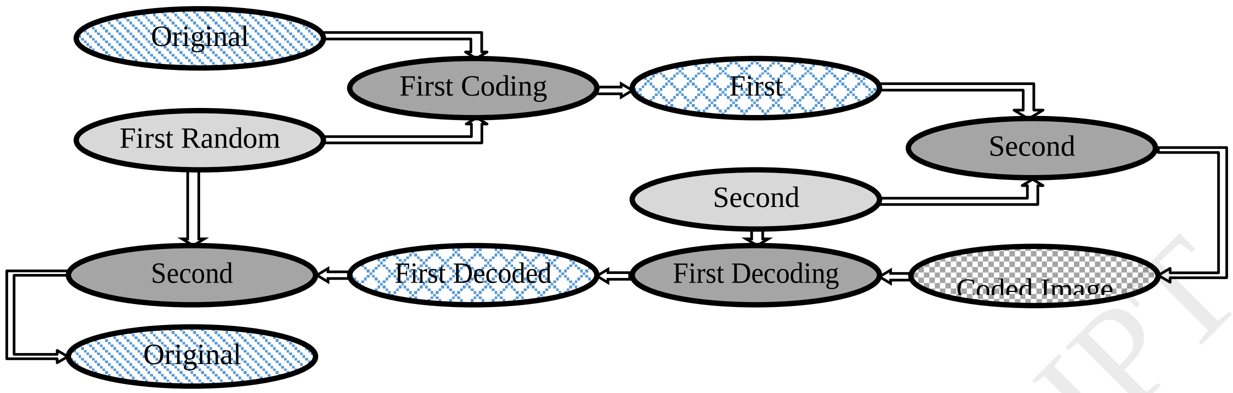


Fig.3. Schematic of proposed algorithm

3.1. Coding process

Our proposed coding process is composed of the following steps:

Step 1: Splitting the image: First of all, Alice splits the original main image into four equal sub-blocks. Then she prepares a $2^N \times 2^M$ random binary image as the coding key and splits it into four equal sub-blocks too (Fig.4). Alice encodes the 1st and 4th sub-blocks of the image according to

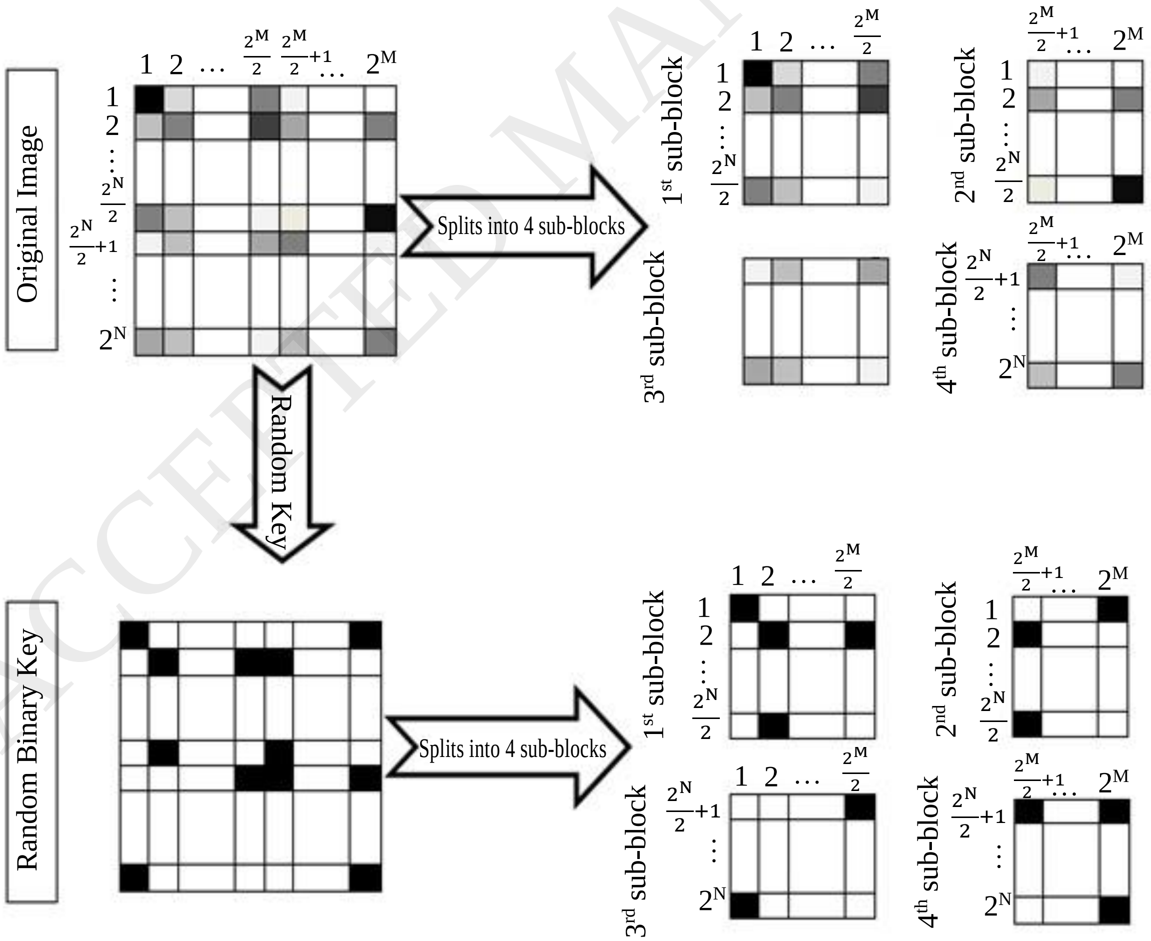
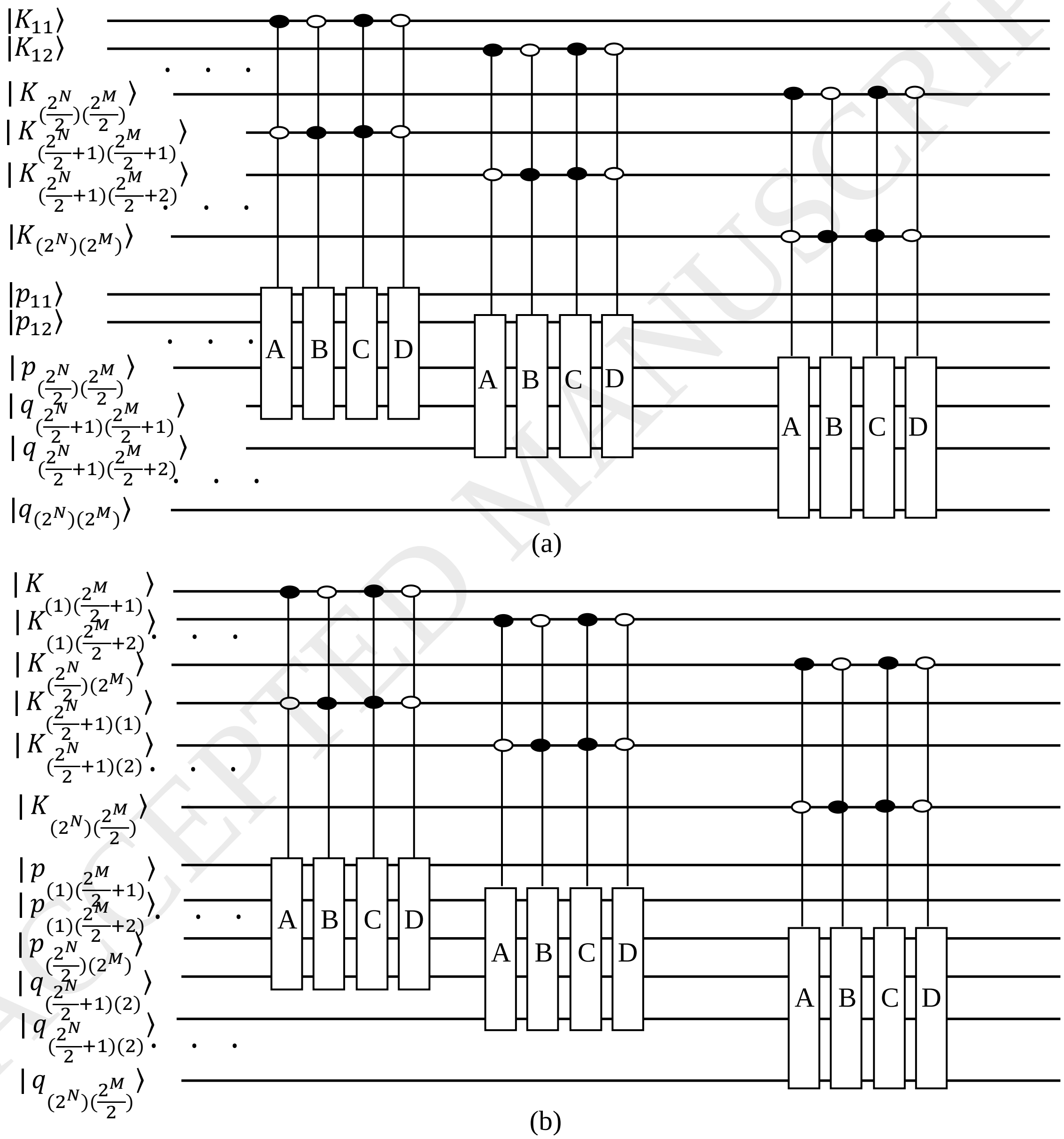


Fig.4. Splitting the original image and random key to 4 sub-blocks

the 1st and 4th sub-blocks of the key, and the 2nd and 3rd sub-blocks of the image according to the 2nd and 3rd sub-blocks of the key.

Step 2: Selecting the coding algorithm based on the key: To code the 1st and 4th (2nd and 3rd) sub-blocks, Alice starts from the first pixel of each sub-block. According to the corresponding qubits of the 1st and 4th (2nd and 3rd) sub-blocks of the binary key, Alice uses one of the A, B, C or D coding algorithms to code these pixels (Table.1, Fig.5). Then she goes on.


$|K_{11}\rangle$
$|K_{12}\rangle$
$|K_{(\frac{2^N}{2})(\frac{2^M}{2})}\rangle$
$|K_{(\frac{2^N}{2}+1)(\frac{2^M}{2}+1)}\rangle$
$|K_{(\frac{2^N}{2}+1)(\frac{2^M}{2}+2)}\rangle$
$|K_{(2^N)(2^M)}\rangle$
$|p_{11}\rangle$
$|p_{12}\rangle$
$|p_{(\frac{2^N}{2})(\frac{2^M}{2})}\rangle$
$|q_{(\frac{2^N}{2}+1)(\frac{2^M}{2}+1)}\rangle$
$|q_{(\frac{2^N}{2}+1)(\frac{2^M}{2}+2)}\rangle$
$|q_{(2^N)(2^M)}\rangle$
A
B
C
D
$|K_{(1)(\frac{2^M}{2}+1)}\rangle$
$|K_{(1)(\frac{2^M}{2}+2)}\rangle$
$|K_{(\frac{2^N}{2})(2^M)}\rangle$
$|K_{(\frac{2^N}{2}+1)(1)}\rangle$
$|K_{(\frac{2^N}{2}+1)(2)}\rangle$
$|K_{(2^N)(\frac{2^M}{2})}\rangle$
$|p_{(1)(\frac{2^M}{2}+1)}\rangle$
$|p_{(1)(\frac{2^M}{2}+2)}\rangle$
$|p_{(\frac{2^N}{2})(2^M)}\rangle$
$|q_{(\frac{2^N}{2}+1)(2)}\rangle$
$|q_{(\frac{2^N}{2}+1)(2)}\rangle$
$|q_{(2^N)(\frac{2^M}{2})}\rangle$

(a)

(b)

Fig.5. Quantum circuit of the key qubits and the selected coding algorithm.

Table.1. Qubits of the key and corresponding coding algorithm.

Qubits of the key	Corresponding coding algorithm	Qubits of the key	Corresponding coding algorithm
10	A	11	C
01	B	00	D

Step 3: Applying the corresponding coding algorithm on pixels: After selecting the coding algorithm based on the qubits of the key, Alice applies one of the 'A', 'B', 'C' or 'D' coding algorithms on the corresponding pixels. To simplify the presentation of algorithms, consider the pixel of the 1st (or 2nd) sub-block as the top pixel with gray scale value of $|p_{ij}\rangle = |p_{ij}^1 p_{ij}^2 \dots p_{ij}^8\rangle$, and the pixel of the 3rd (or 4th) sub-block as the bottom pixel with gray scale value of $|q_{kt}\rangle = |q_{kt}^1 q_{kt}^2 \dots q_{kt}^8\rangle$. Advert that Alice always codes all of the odd qubits before the even qubits. These four coding algorithms are presented in detail below:

Algorithm 'A': If the qubits of the key are 10, Alice applies the algorithm 'A' on corresponding pixels. Using algorithm 'A' to code the pixels, Alice codes every qubit of the gray scale value of them as follows:

For the top pixel: To code the value of qubit p_{ij}^c, if *c* is an odd number, she applies the XOR gate on the qubit p_{ij}^c with the next qubit of the pixel (p_{ij}^{c+1}) and puts the result in p_{ij}^c. If *c* is an even number, she applies XOR gate on the qubit p_{ij}^c with the qubit $k_{(1)(c-1)}$ of the key image and puts the result in p_{ij}^c.

For the bottom pixel: To code the value of qubit q_{kt}^c, if *c* is an odd number, she applies the XOR gate the qubit q_{kt}^c with the next qubit of the pixel (q_{kt}^{c+1}) and puts the result in q_{kt}^c. If *c* is an even number, she applies the NOT gate the qubit q_{kt}^c.

Finally, Alice swaps the coded grayscale values of the top and bottom pixels.

Algorithm 'B': If the qubits of the key are 01, Alice applies the algorithm 'B' on corresponding pixels. Using algorithm 'B' to code the pixels, Alice codes every qubit of the gray scale value of them as follows:

For the top pixel: To code the value of qubit p_{ij}^c, if *c* is an odd number, she applies the NOT gate on the qubit p_{ij}^c. If *c* is an even number, she applies XOR gate on the qubit p_{ij}^c with the previous qubit of the pixel (p_{ij}^{c-1}) and puts the result in p_{ij}^c.

For the bottom pixel: To code the value of qubit q_{kt}^c, if *c* is an odd number, she applies the XOR gate the qubit q_{kt}^c with the qubit $k_{(2^N)(2^M-7+c)}$ of the key image and puts the result in q_{kt}^c. If *c* is an even number, she applies XOR gate on the qubit q_{kt}^c with the previous qubit of the pixel (q_{kt}^{c-1}) and puts the result in q_{kt}^c.

Finally, Alice swaps the coded grayscale values of the top and bottom pixels.

Algorithm 'C': If the qubits of the key are 11, Alice applies the algorithm 'C' on corresponding pixels. Using algorithm 'C' to code the pixels, Alice codes every qubit of the grayscale value of them as follows:

For the top pixel: To code the value of qubit p_{ij}^{c}, if *c* is an odd number, she applies the XOR gate on the qubit p_{ij}^{c} with the p_{ij}^{9-c} qubit of the pixel and puts the result in p_{ij}^{c}. If *c* is an even number, she applies XOR gate on the qubit p_{ij}^{c} with the qubit $k_{(1)(9-c)}$ of the key image and puts the result in p_{ij}^{c}.

For the bottom pixel: To code the value of qubit q_{kt}^{c}, if *c* is an odd number, she applies the XOR gate the qubit q_{kt}^{c} with the qubit q_{kt}^{9-c} of the pixel and puts the result in q_{kt}^{c}. If *c* is an even number, she applies XOR gate on the qubit q_{kt}^{c} with the $k_{(2^{N})(2^{M}+1-c)}$ qubit of the key image and puts the result in q_{kt}^{c}.

Algorithm 'D': If the qubits of the key are 00, Alice applies the algorithm 'D' on corresponding pixels. Using algorithm 'D' to code the pixels, Alice codes every qubit of the gray scale value of them as follows:

For the top pixel: To code the value of qubit p_{ij}^{c}, if *c* is an odd number, she applies the XOR gate on the qubit p_{ij}^{c} with the $k_{(2^{N})(2^{M}+1-c)}$ qubit of the key image and puts the result in p_{ij}^{c}. If *c* is an even number, she applies XOR gate on the qubit p_{ij}^{c} with the qubit p_{ij}^{9-c} of the pixel and puts the result in p_{ij}^{c}.

For the bottom pixel: To code the value of qubit q_{kt}^{c}, if *c* is an odd number, she applies the XOR gate the qubit q_{kt}^{c} with the qubit $k_{(1)(9-c)}$ of the key image and puts the result in q_{kt}^{c}. If *c* is an even number, she applies XOR gate on the qubit q_{kt}^{c} with the q_{kt}^{9-c} qubit of the pixel and puts the result in q_{kt}^{c}.

Fig.6 shows the quantum circuits of these four coding algorithms.

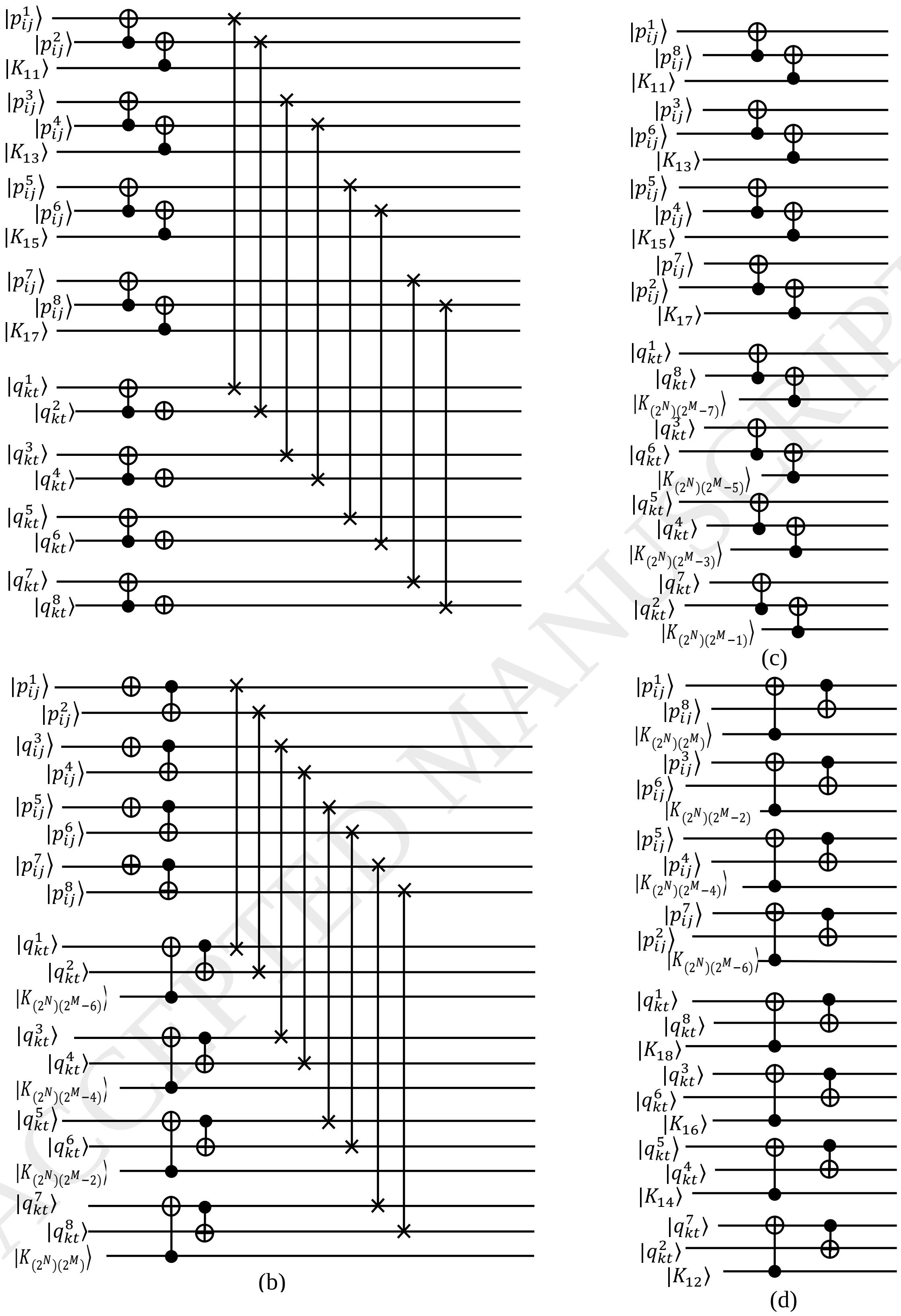


Fig.6. Quantum circuits of (a) the coding algorithm 'A', (b) the oding algorithm 'B', (c) the coding algorithm 'C' and (d) the coding algorithm 'D'.

3.2. Decoding process

The decoding process is composed of several steps similar to the coding process:

Step 1: Splitting the Coded Image: by the same concept of coding process, Alice splits the coded image into four equal sub-blocks which resulted to split the binary key image into four equal sub-blocks. Here again, Alice decodes the 1st and 4th sub-blocks of image according to the 1st and 4th sub-blocks of the key, and the 2nd and 3rd sub-blocks of image according to the 2nd and 3rd sub-blocks of the key respectively.

Step 2: Selecting the Decoding Algorithm based on the Key: To decode the 1st and 4th (2nd and 3rd) sub-blocks of image, Alice starts from the first pixel of each sub-block. According to the corresponding qubit values of the 1st and 4th (2nd and 3rd) sub-blocks of the key, Alice uses one of the 'A', 'B', 'C' or 'D' algorithms to decode these pixels as illustrated in (Table.2, Fig.7). Then, she will go ahead to continue the process.

Table.2. Relation between the Quantum bits of the key and their corresponding decoding algorithm.

Qubits of the key	Corresponding algorithm	Qubits of the key	Corresponding algorithm
10	A	11	C
01	B	00	D

Step 3: Applying the Corresponding Decoding Algorithm on Pixels: After selecting the appropriate decoding algorithm according to the qubits of the key, Alice applies one of the 'A', 'B', 'C' or 'D' decoding algorithms on the corresponding pixels. Consider that Alice always decodes all of the even qubits before the odd one. These four algorithms are presented in details below:

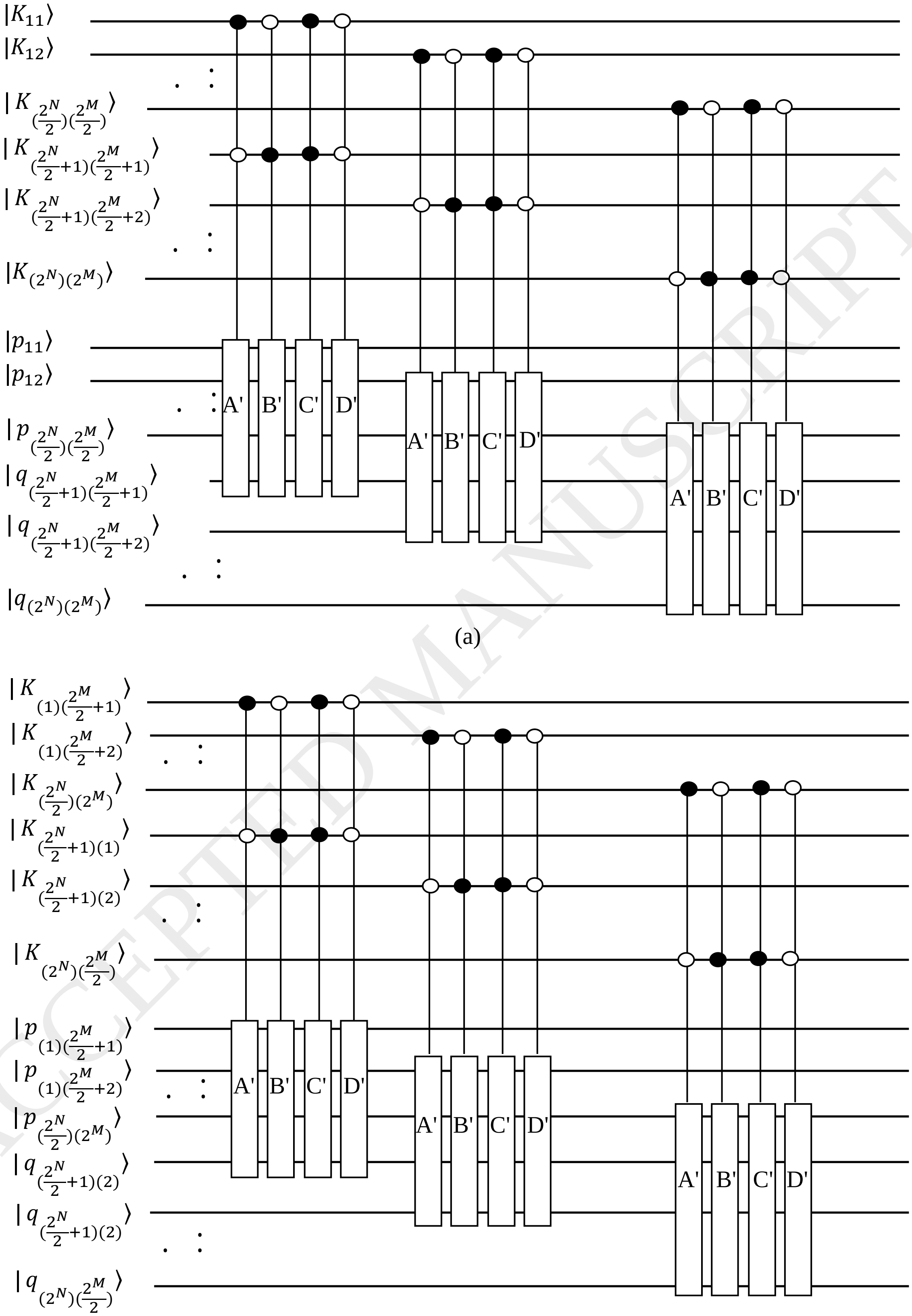
Algorithm 'A': In case of the qubits of the key are 10, then, Alice swaps the coded grayscale values of the top and bottom pixels. After that:

For the top pixel: To decode the value of qubit p_{ij}^{c}, if *c* is an even number, she applies XOR gate on the qubit p_{ij}^{c} with the qubit $k_{(1)(c-1)}$ of the key image and stores the result in p_{ij}^{c}. If *c* is an odd number, she applies the XOR gate on the qubit p_{ij}^{c} with the next qubit of the pixel (p_{ij}^{c+1}) and stores the result in p_{ij}^{c}.

For the bottom pixel: To decode the value of qubit q_{kt}^{c}, if *c* is an even number, she applies the NOT gate the qubit q_{kt}^{c}. If *c* is an odd number, she applies the XOR gate the qubit q_{kt}^{c} with the next qubit of the pixel (q_{kt}^{c+1}) and stores the result in q_{kt}^{c}..

Algorithm 'B': In case of the qubits of the key are 01, at first Alice swaps the coded grayscale values of the top and bottom pixels. After that:

For the top pixel: To decode the value of qubit p_{ij}^{c}, if *c* is an even number, she applies XOR gate on the qubit p_{ij}^{c} with the previous qubit of the pixel (p_{ij}^{c-1}) and stores the result in p_{ij}^{c}. If *c* is an odd number, she applies the NOT gate on the qubit p_{ij}^{c}.


$|K_{11}\rangle$
$|K_{12}\rangle$
$|K_{(\frac{2^N}{2})(\frac{2^M}{2})}\rangle$
$|K_{(\frac{2^N}{2}+1)(\frac{2^M}{2}+1)}\rangle$
$|K_{(\frac{2^N}{2}+1)(\frac{2^M}{2}+2)}\rangle$
$|K_{(2^N)(2^M)}\rangle$
$|p_{11}\rangle$
$|p_{12}\rangle$
$|p_{(\frac{2^N}{2})(\frac{2^M}{2})}\rangle$
$|q_{(\frac{2^N}{2}+1)(\frac{2^M}{2}+1)}\rangle$
$|q_{(\frac{2^N}{2}+1)(\frac{2^M}{2}+2)}\rangle$
$|q_{(2^N)(2^M)}\rangle$
A'
B'
C'
D'
$|K_{(1)(\frac{2^M}{2}+1)}\rangle$
$|K_{(1)(\frac{2^M}{2}+2)}\rangle$
$|K_{(\frac{2^N}{2})(2^M)}\rangle$
$|K_{(\frac{2^N}{2}+1)(1)}\rangle$
$|K_{(\frac{2^N}{2}+1)(2)}\rangle$
$|K_{(2^N)(\frac{2^M}{2})}\rangle$
$|p_{(1)(\frac{2^M}{2}+1)}\rangle$
$|p_{(1)(\frac{2^M}{2}+2)}\rangle$
$|p_{(\frac{2^N}{2})(2^M)}\rangle$
$|q_{(\frac{2^N}{2}+1)(2)}\rangle$
$|q_{(\frac{2^N}{2}+1)(2)}\rangle$
$|q_{(2^N)(\frac{2^M}{2})}\rangle$

(a)

(b)

Fig.7. Quantum circuit of the key qubits and selected decoding algorithm.

For the bottom pixel: To decode the value of qubit q_{kt}^{c}, if *c* is an even number, she applies XOR gate on the qubit q_{kt}^{c} with the previous qubit of the pixel (q_{kt}^{c-1}) and stores the result in q_{kt}^{c}. If *c* is an odd number, she applies the XOR gate the qubit q_{kt}^{c} with the qubit $k_{(2^N)(2^M-7+c)}$ of the key image and puts the result in q_{kt}^{c}.

Algorithm 'C': In case of the qubits of the key are 11:

For the top pixel: to decode the value of qubit p_{ij}^{c}, if *c* is an even number, she applies XOR gate on the qubit p_{ij}^{c} with the qubit $k_{(1)(9-c)}$ of the key image and stores the result in p_{ij}^{c}. If *c* is an odd number, she applies the XOR gate on the qubit p_{ij}^{c} with the p_{ij}^{9-c} qubit of the pixel and stores the result in p_{ij}^{c}.

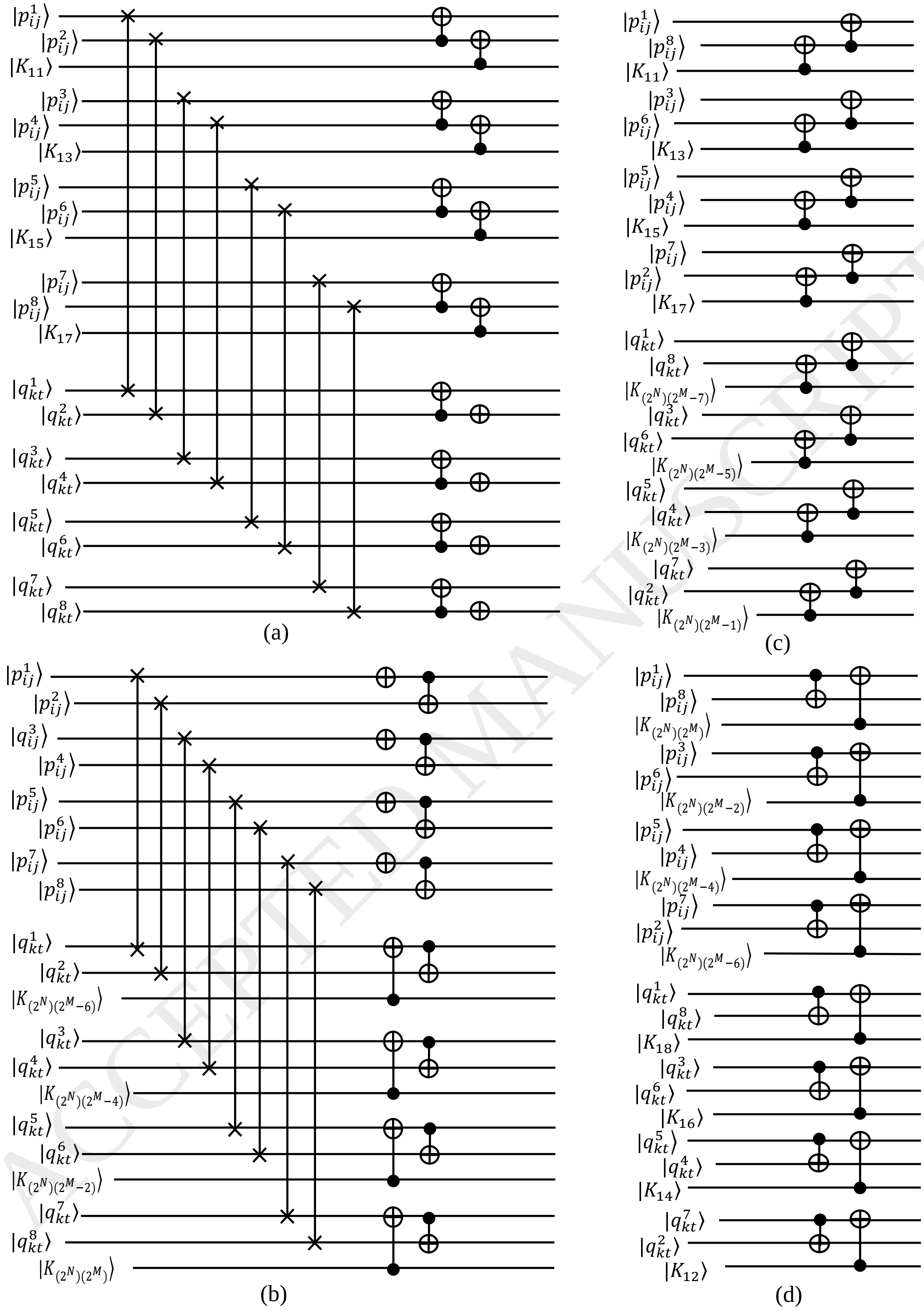
For the bottom pixel: to decode the value of qubit q_{kt}^{c}, if *c* is an even number, she applies XOR gate on the qubit q_{kt}^{c} with the $k_{(2^N)(2^M+1-c)}$ qubit of the key image and stores the result in q_{kt}^{c}. If *c* is an odd number, she applies the XOR gate the qubit q_{kt}^{c} with the qubit q_{kt}^{9-c} of the pixel and stores the result in q_{kt}^{c}.

Algorithm 'D': In case of the qubits of the key are 00:

For the top pixel: to decode the value of qubit p_{ij}^{c}, if *c* is an even number, she applies XOR gate on the qubit p_{ij}^{c} with the qubit p_{ij}^{9-c} of the pixel and stores the result in p_{ij}^{c}. If *c* is an odd number, she applies the XOR gate on the qubit p_{ij}^{c} with the $k_{(2^N)(2^M+1-c)}$ qubit of the key image and stores the result in p_{ij}^{c}.

For the bottom pixel: To decode the value of qubit q_{kt}^{c}, if *c* is an even number, she applies XOR gate on the qubit q_{kt}^{c} with the q_{kt}^{9-c} qubit of the pixel and stores the result in q_{kt}^{c}. If *c* is an odd number, she applies the XOR gate the qubit q_{kt}^{c} with the qubit $k_{(1)(9-c)}$ of the key image and stores the result in q_{kt}^{c}.

Fig.8 shows the quantum circuits of these four decoding algorithms.



(a)

(c)

(b)

(d)

Fig.8. Quantum circuits of (a) decoding algorithm 'A', (b) decoding algorithm 'B', (c) decoding algorithm 'C' and (d) decoding algorithm 'D'.

4. Simulation

Consider a simple meaningful 4×8 image and a binary 4×8 key image as illustrated in Fig.9 (a) and (b) respectively. The pixels of four sub-blocks of the original image according to the corresponding qubits of the key image, are coded using one of 'A', 'B', 'C' or 'D' coding algorithms as illustrated in Table.3. The result of running the coding process once proves that the proposed algorithm codes the original image into a meaningless one as shown in Fig.9 (c). Therefore, we can conclude that a better result will be achieved while running the algorithm twice.

Table.3. Relation among the pixels of the original image, the corresponding qubits of the key, the selected algorithm and the output of the algorithm.

Top pixel	Bottom pixel	Top key qubit	Bottom key qubit	Selected algorithm	Top coded pixel	Bottom coded pixel
1st and 4th sub-blocks						
P_{11}=00011001	Q_{35}=10001001	K_{11}=0	K_{35}=0	D	P'_{11}=00001000	Q'_{35}=11101110
P_{12}=00111001	Q_{36}=10100111	K_{12}=1	K_{36}=0	A	P'_{12}=01011100	Q'_{36}=10001101
P_{13}=01010010	Q_{37}=11001111	K_{13}=0	K_{37}=1	B	P'_{13}=10001001	Q'_{37}=00001101
P_{14}=01101101	Q_{38}=11101101	K_{14}=0	K_{38}=1	B	P'_{14}=10101011	Q'_{38}=00011000
P_{21}=00011001	Q_{45}=10001001	K_{21}=0	K_{45}=1	B	P'_{21}=01000111	Q'_{45}=11000100
P_{22}=00111001	Q_{46}=10100111	K_{22}=0	K_{46}=0	D	P'_{22}=00101000	Q'_{46}=11100000
P_{23}=01010010	Q_{47}=11001111	K_{23}=0	K_{47}=0	D	P'_{23}=11000001	Q'_{47}=10001010
P_{44}=01101101	Q_{48}=11101101	K_{24}=0	K_{48}=1	B	P'_{24}=10101011	Q'_{48}=00011000
2nd and 3rd sub-blocks						
P_{15}=10001001	Q_{31}=00011001	K_{15}=1	K_{31}=0	A	P'_{15}=10110111	Q'_{31}=01101101
P_{16}=10100111	Q_{32}=00111001	K_{16}=1	K_{32}=1	C	P'_{16}=11101000	Q'_{32}=10000101
P_{17}=11001111	Q_{33}=01010010	K_{17}=1	K_{33}=1	C	P'_{17}=10010100	Q'_{33}=10111010
P_{18}=11101101	Q_{34}=01101101	K_{18}=1	K_{34}=0	A	P'_{18}=11010011	Q'_{34}=00011001
P_{25}=10001001	Q_{41}=00011001	K_{25}=1	K_{41}=1	C	P'_{25}=10010010	Q'_{41}=10100001
P_{26}=10100111	Q_{42}=00111001	K_{26}=0	K_{42}=0	D	P'_{26}=10011110	Q'_{42}=01010110
P_{27}=11001111	Q_{43}=01010010	K_{27}=0	K_{43}=1	B	P'_{27}=00111110	Q'_{43}=10111010
P_{28}=11101101	Q_{44}=01101101	K_{28}=0	K_{44}=1	B	P'_{28}=00101011	Q'_{44}=10011000

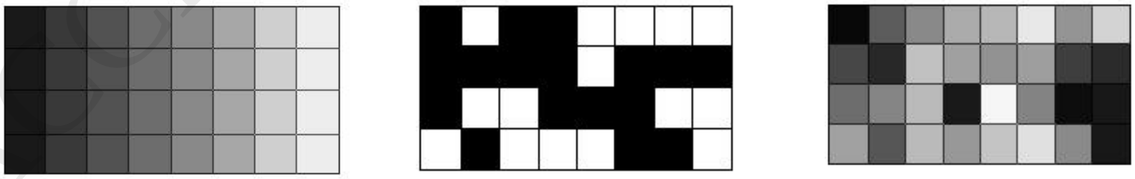


Fig.9. (a) The original image. (b) The binary key image. (c) Once coded image.

Here, we present the results of applying our proposed cryptography algorithm on selected real images as shown in Fig.10. The figure shows the histogram diagrams of encoded images resulting from running the coding process on the original image once and twice respectively. Since the histogram diagram shows the affluence of pixels with each gray-scale value in image, the histogram diagram of the encoded image has to be flatter than the original one which can be quantified using Shannon’s entropy.

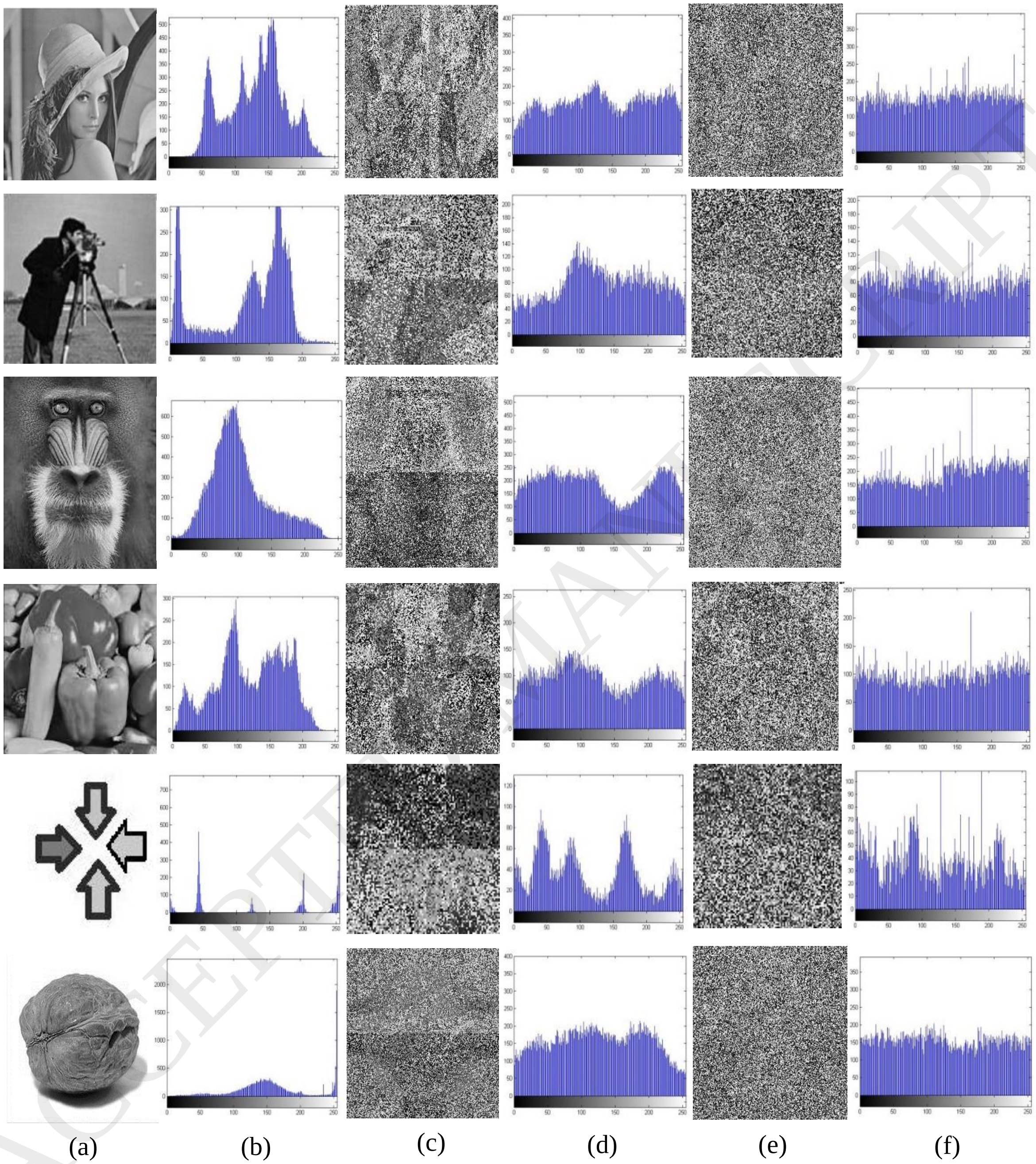


Fig.10. (a) Original image. (b) Histogram of the original image. (c) Once coded image. (d) Histogram of once coded image. (e) Twice coded (final) image. (f) Histogram of final image.

From Fig.10, it can be concluded that the histogram diagram of the final coded image is flatter than both the ones of the original and once coded images respectively. In cases with flat background such as the arrows case, the histogram of the original image is discrete, which means that the original image contains just a few couple of grayscale values and other values doesn’t appear in the image. Nevertheless the histogram of the final coded image is partly flat. Therefore, the appearance of the new grayscale values in the final coded image is enormously.

Entropy is one of good methods to express randomness or uncertainty of a series of random variables. In information theory, the entropy is used to quantify the minimum descriptive complexity of a random variable. For an image, entropy indicates the amount of information can be achieved by the image.

For a random variable X, with n outcomes{x_1, x_2,…,x_n }, the Shannon entropy H(X) is defined as [42,43]:

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_b p(x_i) \qquad (5)$$

Where p(x_i) is probability mass function of outcome x_i. The entropy E_n of image n can be calculated as:

$$E_n = \sum_{i=0}^{255} p(i) * \log_2(\frac{1}{p(i)}) \qquad (6)$$

where p(x_i) is (number of occurrences of pixel ‘i’/total number of pixels). Hence, a perfectly scrambled 8-bit image must have a flat histogram with $E_n = \sum_{i=0}^{255} p(i) * \log_2(\frac{1}{p(i)}) = \sum_{i=0}^{255} \frac{1}{256} * \log_2\left(\frac{1}{\frac{1}{256}}\right) = 8$.

Needless to say that, if all pixels in an image have the same grayscale level or the same intensity of color components, the minimum entropy is achieved. On the other hand, when each pixel of an image presents specific grayscale level or color intensity, the image will exhibit maximum entropy. The higher the value of entropy becomes, less information can be revealed.

The calculated Shannon entropy for the simulated sample images is presented in Table.4.

Table.4. Calculated Shannon entropy for the simulated sample images.

Image	Entropy of the original image	Entropy of once coded image	Entropy of twice coded image
Lena	7.3226	7.9463	7.9616
Camera man	7.1549	7.9198	7.9514
Baboon	7.3463	7.9128	7.9517
Peppers	7.5900	7.9393	7.9547
Arrows	3.2953	7.6589	7.9199
Walnut	5.3504	7.7280	7.9626

As it is illustrated in Table.4, each coding step makes a considerable increase in the values of the image entropy; therefore the proposed coding process imposed a considerable diversity or uncertainty to the original image.

Although in this study each picture is coded twice, but there is no actual limitation on number of iterations and user can code the image more than twice. Needless to say, since in every iteration, a unique random key is generated and stored, larger number of iterations increases the information which would be used to decode the output image to the original one.

5. Conclusion:

A new secure efficient bi-step quantum images cryptography algorithm is proposed. Here, four different coding algorithms are introduced. In this scheme, for the aim of completing the coding task, a randomized binary image key is generated during the procedure. In the coding process, based on both the pair of original images pixels and their corresponding quantum bits of the generated randomized key, the coding algorithm is selected.

In each coding step, the key image not only is used to select the applying coding algorithm for each pixel, but also some qubits of the key is used directly to change the grayscale value of the pixel. This means that if one doesn't have the randomized key images, it is impossible for him to decode the original image correctly.

From the experimental results, it can be seen that the proposed algorithm codes different kinds of images very well and the final coded images were completely meaningless. In each case, the histogram diagram of the final coded image is so much flatter than the histogram diagrams of the original and once coded images. Also, in each case, the calculated entropy of the twice coded image is higher than the calculated entropy of original and once coded images, which confirms that the proposed coding process imposed a considerable uncertainty to the original image.

Acknowledgement

This work is supported by Kermanshah branch, Islamic Azad University, Kermanshah, Iran.

References:

[1] C. H. Bennett, G. Brassard, in: Proceedings of the IEEE International Conference on Computers, Systems and Signal Processing, Bangalore, India (IEEE, New York, 1984), p. 175

[2] Y. S. Zhang, C.F. Li, and G. C. Guo, Quantum key distribution via quantum encryption, Phys. Rev. A 64, 024302 (2001)

[3] N. Zhou, G. Zeng, W. Zeng, F. Zhu, Cross-center quantum identification scheme based on teleportation and entanglement swapping, Optics Communications. 254, 380 (2005)

[4] M. Naseri, Eavesdropping on secure quantum telephone protocol with dishonest server, Opt. Commun. 282, 278-282 (2009)

[5] N. R. Zhou, L. J. Wang, J. Ding, L. H. Gong, Quantum deterministic key distribution protocols based on the authenticated entanglement channel, Physica Scripta 81, 045009 (2010)

[6] M. Naseri, A weak blind signature based on quantum cryptography, Int. J. Phys. Sci. 6, 5051 (2011)

[7] X. B. Chen, G. Xu, Y. Su, et al. Robust variations of secret sharing through noisy quantum channel. Quantum Information & Computation 14 (2014) 0589.

[8] X. B. Chen, Z. Dou, G. Xu, et al, A class of protocols for quantum private comparison based on the symmetry of states. Quantum Information Processing 13 (2014) 85.

[9] J. Li, X. B. Chen, G. Xu, et al. Perfect quantum network coding independent of classical network solutions. IEEE Communications Letters 19 (2015) 115.

[10] S. Y. Kang, X. B. Chen, and Y. X. Yang. Asymmetric Quantum Information Splitting of an Arbitrary N -qubit State via GHZ-like State and Bell States. International Journal of Theoretical Physics 53 (2014) 1848.

[11] G. Xu, X. B. Chen, Z. Duo, et al. A novel protocol for multiparty quantum key management. Quantum Information Processing 14 (2015) 2959.

[12] M. Naseri et al., A scheme for secure quantum communication network with authentication using GHZ-like states and cluster states controlled teleportation, Quantum Inf. Process. 14, 4279-4295(2015)

[13] M. Naseri, Revisiting quantum authentication scheme based on entanglement swapping, Int. J.Theor.Phys. 55, 2428-2435 (2016)

[14] G. Xu, X. B. Chen, Z. Dou, et al. Novel criteria for deterministic remote state preparation via the entangled six-qubit state, Entropy 18 (2016) 267.

[15] Vlasov, Alexander Yu., Quantum computations and images recognition. arXiv preprint quant-ph/9703010 (1997)

[16] Beach, G., Lomont, C., Cohen, C.: Quantum image processing. In: Proceedings of The 2003 IEEE Workshop on Applied Imagery Pattern Recognition, pp. 39–44 (2003)

[17] Venegas-Andraca, S.E., Bose, S.: Quantum computation and image processing: new trends in artificial intelligence. In: Proceedings of the International Conference on Artificial Intelligence IJCAI-03, pp. 1563–1564 (2003)

[18]Venegas-Andraca, S.E., Bose, S.: Storing, processing and retrieving an image using quantum mechanics. In: Proceedings of the SPIE Conference Quantum Information and Computation, pp. 137-147 (2003)

[19] Le, P., Dong, F., Hirota, K., A flexible representation of quantum images for polynomial preparation, image compression, and processing operations. Quantum Inf. Process. 10(1), 63–84 (2011)

[20] 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)

[21] Sun, B., Iliyasu, A., Yan, F., Dong, F., Hirota, K.: An RGB multi-channel representation for images on quantum computers. J. Adv. Comput. Intell. Intell. Info. 17(3), 404–417 (2013)

[22] Abdolmaleky. M et al, Red-Green-Blue Multi-Channel Quantum Representation of Digital Images, Optik-International Journal for Light and Electron Optics http://dx.doi.org/10.1016/j.ijleo.2016.09.123

[23] 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)

[24] Caraiman, S., Manta, I.: Histogram-based segmentation of quantum images. Theor. Comput. Sci. 529, 46–60 (2014)

[25] Caraiman, S., Manta, V.I.: Quantum image filtering in the frequency domain. Adv. Electr. Comput. Eng. 13(3), 77–84 (2013)

[26] P., Dong, F., Hirota, K.: A flexible representation of quantum images for polynomial preparation, image compression, and processing operations. Quantum Inf. Process. 10 (1), 6384 (2011)

[27] Jiang, Nan, Yijie Dang, and Jian Wang, Quantum image matching. Quantum Information Processing 1-30 (2016)

[28] Natori, Shin. Why quantum steganography can be stronger than classical steganography. Quantum Computation and Information. Springer Berlin Heidelberg, 235-240 (2006)

[29] Xu, Shu Jiang, et al. High-efficiency quantum steganography based on the tensor product of Bell states. Science China Physics, Mechanics and Astronomy 56.9: 1745-1754 (2013)

[30] Z. H. Wei, X. B. Chen, X. X. Niu, et al. The Quantum Steganography Protocol via Quantum Noisy Channels. International Journal of Theoretical Physics 54 2505 (2015)

[31] Shaw, Bilal A., and Todd A. Brun. Quantum steganography with noisy quantum channels. Physical Review A 83.2 : 022310. (2011)

[32] Zhang, Wei-Wei, et al. A watermark strategy for quantum images based on quantum fourier transform. Quantum Information Processing 12.2 : 793-803 (2013)

[33] Song, Xian-Hua, et al. A dynamic watermarking scheme for quantum images using quantum wavelet transform. Quantum information processing 12.12: 3689-3706 (2013)

[34] M. Naseri, et al. A new secure quantum watermarking scheme, Optik-International Journal for Light and Electron Optics 139, 77-86 (2017)

[35] Yan F., Iliyasu A.M., Venegas-Andraca S.E., A survey of quantum image representation, Quantum information processing 15, 1-35 (2016)

[36] Yan F., Chen K., Venegas-Andraca S.E., Zhao J., Quantum image rotation by an arbitrary angle, Quantum information processing 16, 282 (2017)

[37] Ye, G.: Image scrambling encryption algorithm of pixel bit based on chaos map. Pattern Recognit. Lett. 31(5), 347-354 (2010).

[38] Dalhoum, A.L.A., Mahafzah, B.A., Awwad, A.A., Aldhamari, I., Ortega, A., Alfonseca, M.: Digital image scrambling using 2D cellular automata. IEEE Multimed. 4, 28-36 (2012).

[39] Jiang, N., Wang, L., Wu, W.Y.: Quantum Hilbert image scrambling. Int. J. Theor. Phys. 53(7), 2463-2484 (2014).

[40] Jiang, N., Wu, W.Y., Wang, L.: The quantum realization of Arnold and Fibonacci image scrambling. Quantum Inf. Process 13(5), 1223-1236 (2014).

[41] Azou, R.G., et al.: Quantum image Gray-code and bit-plane scrambling. Quantum Inf. Process. 14:1717-1734 (2015).

[42] B. Furht , D. Kirovsk, Multimedia Security Handbook, 2005.

[43] Iliyasu A.M., Yan F., K. Hirota, Metric for estimating congruity between quantum images, *Entropy*. *18*, 360 **(2016**)