International Journal of Theoretical Physics (2023) 62:52
https://doi.org/10.1007/s10773-023-05303-1

A Novel Quantum Image Steganography Algorithm Based on Double-Layer Gray Code

Jin-Liang Yao[1] · Hong-Mei Yang[1] · Dong-Huan Jiang[2] · Bin Yan[3] · Jeng-Shyang Pan[1] · Meng-Xi Wang[1]

Received: 15 July 2022 / Accepted: 1 February 2023 / Published online: 3 March 2023

Abstract
In the development of quantum image steganography, the visual effect of the image has always been the focus of scholars' research. Based on the classical Gray code algorithm and the least significant bit (LSB) algorithm, a novel quantum image steganography algorithm is proposed in this paper. Firstly, the Arnold scrambling method is used to scramble the information image. In the meantime, the scrambled information image is expanded to the same size as the carrier image by using the quantum expansion method. Secondly, the double-layer Gray code rule is applied to the carrier pixels. This operation can reduce the change rate of LSB of carrier pixels. At the same time, the key image used in the extracting phase is generated. The key image improves the security of the algorithm. The classical Gray code algorithm needs to change 50% bits of the LSB of the carrier pixels when embedding. The proposed algorithm in this paper only needs to change 25% bits of the LSB of the carrier pixels when embedding. The PSNR value of the algorithm proposed in this paper can be around 54dB in the visual effect experiment, which is around 3dB higher than that of the classical Gray code algorithm. The robustness experiment is also better compared with other algorithms.

Keywords Quantum image steganography · Gray code · Least significant bit · Quantum circuit · Quantum expansion

1 Introduction

Steganography is a technology that hides information in a carrier. It has a very promising application and can improve the security of information delivery. For example, digital steganography technology can be used instead of encryption technology to transmit business contracts. Steganography focuses more on hiding the transmission of information compared to digital watermarking techniques. In contrast, digital watermarking techniques focus on

✉ Hong-Mei Yang
yanghongmei@sdust.edu.cn

Extended author information available on the last page of the article.

protecting the copyright of the image. Meanwhile, digital watermarking techniques are more concerned with robustness.

The quantum computer was first proposed in 1982 by Richard Feynman, winner of the Nobel Prize in Physics. Based on the superposition state principle of quantum mechanics, the computation speed of the quantum computer is much faster than that of the classical computer. In the development of quantum image processing, classical images first need to be represented on the quantum computer. The main methods of representation are as follows: qubit lattice [1], real ket [2], entangled image [3], flexible representation of quantum images (FRQI) [4], a novel enhanced quantum representation (NEQR) [5], normal arbitrary quantum superposition state (NAQSS) [6], improved neqr (INEQR) [7] and generalized quantum image representation (GQIR) [8]. On the basis of these quantum image representation methods, researchers have investigated many aspects of quantum image processing. Such as quantum image scaling [9, 10], quantum image scrambling [11, 12], quantum image watermarking [13–15], quantum image steganography [16, 17], and quantum image matching [18].

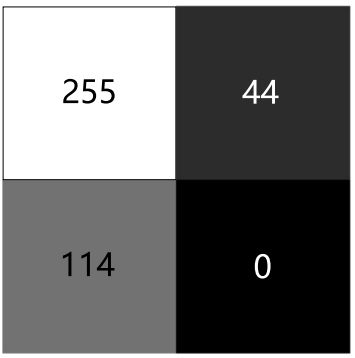
In this paper, quantum image steganography is mainly studied. The least significant bit (LSB) steganography algorithm is the most basic one among the commonly used algorithms. Based on the LSB algorithm, Nan Jiang et al. proposed block LSB algorithm [19] has a good visual effect but the embedding capacity is poor. The algorithm proposed by Luo et al. [20] has larger embedding capacity but poor visualization. The algorithm proposed by Zhou et al. [21] uses more auxiliary graphics to enhance the visual effect, but the embedding capacity and robustness need to be improved. The algorithm proposed by Luo et al. [22] has a very good visual effect, but the poor robustness is one of its disadvantages. The algorithm proposed by Zhou et al. [23] has a considerable visual effect, but the security is flawed. The algorithm proposed by Qu et al. [24] has higher embedding efficiency and embedding rate. The proposed algorithm by Zeng et al. [25] has better security and embedding capacity. Chen and Chang proposed a steganography algorithm combining Gray code and LSB algorithm [26], this steganography algorithm has a better visual effect compared to the classical LSB algorithm without reducing the embedding capacity. This paper presents an improved algorithm based on the algorithm proposed by Chen and Chang [26]. The algorithm proposed in this paper has a significant improvement in visual effect without changing the embedding capacity.

The next parts of this paper is rough as follows. The second part will list the preparation and related knowledge. The third part will give a detailed description of the proposed algorithm as well as an analysis of the complexity of the quantum circuit. In the fourth part, the experimental results are analyzed. Finally, conclusions are drawn based on the previous sections.

2 Background

2.1 Novel Enhanced Quantum Representation for Quantum Images

The NEQR model was first proposed in 2013 [15]. Compared to the previous representation, the NEQR model represents the quantum image position well along with its grayscale value.



$$|I\rangle = \frac{1}{2}(|11111111\rangle \otimes |00\rangle + |00101111\rangle \otimes |01\rangle + |01110010\rangle \otimes |10\rangle + |11111111\rangle \otimes |11\rangle)$$

Fig. 1 NEQR representatio

This advantage substantially improves the scope of application of the NEQR model. The quantum image can be represented in the NEQR model as:

$$|I\rangle = \frac{1}{2^n} \sum_{Y=0}^{2^n-1} \sum_{X=0}^{2^n-1} |C_{YX}\rangle |YX\rangle \tag{1}$$

In this equation, $|C_{YX}\rangle = |c_{YX}^{q-1} \cdots c_{YX}^{1} c_{YX}^{0}\rangle$ denotes the grayscale value of the quantum image, the range of which can be expressed as $[0, 2^q - 1]$. $|Y\rangle$and$|X\rangle$ denote the position information of the quantum image in the vertical and horizontal directions, respectively. As shown in Fig. 1, a 2×2 quantum image can be represented in NEQR as follow.

2.2 The Least Significant Bit (LSB)

LSB steganographic algorithm [27] is a very basic algorithm nowadays. Numerous algorithms are derived on the basis of the LSB algorithm. The LSB algorithm has a very important position in the steganography field. The LSB algorithm is to replace the LSB of the carrier pixel with the information bit. The LSB of the pixel is shown in Fig. 2. For example, if the information bit to be embedded is "1" and the carrier pixel is 236, then the carrier pixel becomes 237 after the replacement. Since only the LSB of the carrier pixel is changed, it is almost imperceptible to the human eye. This advantage makes the LSB algorithm widely used.

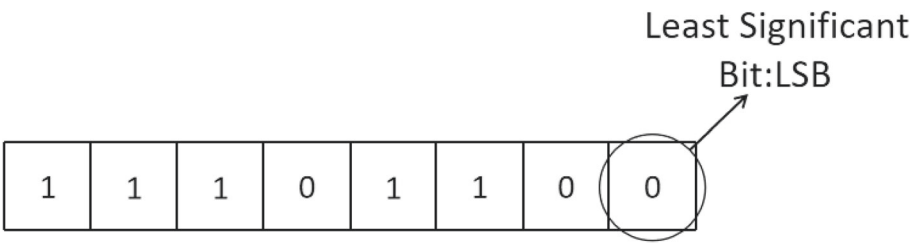


Fig. 2 Example of the LSB

2.3 Gray Code and its use in Steganography

Gray code is a kind of binary code, proposed by Frank Gray in 1953. Gray code was first used in communications. The characteristic of the Gray code is that only one bit is different between two adjacent numbers. At the same time, the minimum and maximum numbers of Gray code are also different by only one bit. Based on this feature, Gray code is also called "cyclic code" or "reflection code". The Gray code is created as follows: The first step is to change the rightmost bit. The second step is to select the first bit from the right that is 1, then change the left bit of this bit. Then repeat the previous two steps until the Gray code is generated. The generated Gray codes G_1 to G_4 are shown in Fig. 3.

In 2008, Chen and Chang [26] designed a new steganography method, which combines Gray code and LSB algorithm. In the following, this paper will use the 3 bits Gray code G_3 as an example.

In the embedding process, a function Gray(g) is defined to represent the Gray code value of the corresponding three bits g. For the value of the information bit is 1, when the last three bits of the carrier pixel correspond to an even Gray code, then the last three bits of the carrier pixel are modified to the odd Gray code with less variation in the neighboring Gray codes. When the last three bits of the carrier pixel correspond to an odd Gray code, then it remains unchanged. Similarly, for the value of the information bit is 0, when the last three bits of the carrier pixel correspond to an odd Gray code, then the last three bits of the carrier pixel are modified to the even Gray code of the neighboring Gray codes with less variation. When the last three bits of the carrier pixel correspond to an even Gray code, then it remains unchanged. For example, suppose the carrier pixel is $235_{10} = 11101011_2$. If the information bit is 1, $\text{Gray}(011_2) = 2$. Since 2 is an even number, modify 011_2 to a neighboring odd

$$G_1 = \begin{cases} 0=0 \\ 1=1 \end{cases} \quad G_2 = \begin{cases} 00=0 \\ 01=1 \\ 11=2 \\ 10=3 \end{cases} \quad G_3 = \begin{cases} 000=0 \\ 001=1 \\ 011=2 \\ 010=3 \\ 110=4 \\ 111=5 \\ 101=6 \\ 100=7 \end{cases} \quad G_4 = \begin{cases} 0000=0 \\ 0001=1 \\ 0011=2 \\ 0010=3 \\ 0110=4 \\ 0111=5 \\ 0101=6 \\ 0100=7 \\ 1100=8 \\ 1101=9 \\ 1111=10 \\ 1110=11 \\ 1010=12 \\ 1011=13 \\ 1001=14 \\ 1000=15 \end{cases}$$

Fig. 3 Gray code

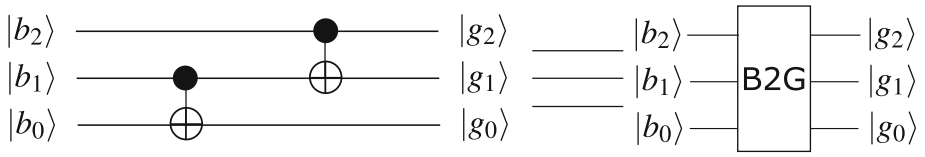


Fig. 4 Conversion of binary code to Gray code

Gray code. Since modifying 011_2 to 010_2 is a smaller color change compared to modifying it to 001_2, modify 011_2 to 010_2. The modified carrier pixel is $234_{10} = 11101010_2$. If the information bit is 0, then no modification is needed at this point.

In the extracting process, only the last three bits of the carrier pixel corresponding to the even or odd number of the Gray code need to be concerned. If the Gray code corresponding to the last three bits of the carrier pixel is an even number, then the extracted information bit is 0. If the Gray code corresponding to the last three bits of the carrier pixel is an odd number, then the extracted information bit is 1. As an example, suppose the carrier pixel is $234_{10} = 11101010_2$. Since $\text{Gray}(010_2) = 3$, which is an odd number, therefore the extracted information bit is 1. The quantum circuits of the conversion of binary code to Gray code and Gray code to binary code are shown in Figs. 4 and 5, respectively.

2.4 Arnold Scrambling

Image scrambling is the process of operating on the horizontal and vertical coordinates of an image to destroy the correlation of the image matrix. This makes the image more different from the original image. In this way, other people cannot read the image information without knowing the scrambling rules. In this paper, the scrambling method is Arnold scrambling [28], a scrambling method first used in image scrambling in 1992. The concrete steps are: First multiply a scrambling matrix with the coordinates, and then take the modulus of the result. Assuming that the coordinate information before the scrambling is (x, y) and after the scrambling is (x_A, y_A). Then the two-dimensional Arnold scrambling can be expressed as follows:

$$\begin{pmatrix} x_A \\ y_A \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} (\text{mod N}) \tag{2}$$

$$x_A = (x + y) \bmod \text{N} \tag{3}$$

$$y_A = (x + 2y) \bmod \text{N} \tag{4}$$

where $\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$ is the scrambling matrix. The inverse operation of the two-dimensional Arnold scrambling is:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}^{-1} \begin{pmatrix} x_A \\ y_A \end{pmatrix} (\text{mod N}) \tag{5}$$

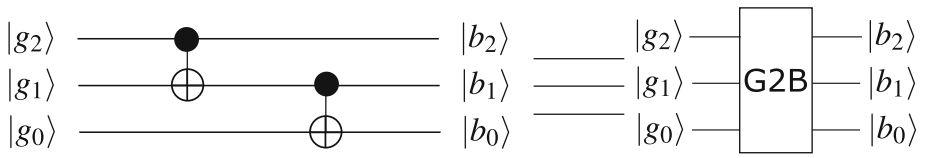


Fig. 5 Conversion of Gray code to binary code

$$x = (2x_A - y_A) \bmod \mathrm{N} \tag{6}$$

$$y = (-x_A + y_A) \bmod \mathrm{N} \tag{7}$$

where $\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}^{-1}$ is the inverse scrambling matrix. The Arnold scrambling quantum circuit is shown in [29].

2.5 Quantum Comparator

In the embedding and extraction steps, it is first necessary to ensure that the positions of the individual images are the same. Therefore, the comparison circuit [30] is introduced in this paper. The comparison circuit compares the position information of two images. When the output $|c\rangle$ is $|1\rangle$, it means that the position coordinate of the two images is equal; when the output $|c\rangle$ is $|0\rangle$, it means that the position coordinate of the two images is unequal at this time. The concrete quantum comparison circuit is shown in Fig. 6.

2.6 Judgment Circuit

The judgment circuit is used to judge whether the value of the 3 bits Gray code is in the [0, 3] interval or in the [4, 7] interval. The judgment circuit is shown in Fig. 7. When the output $|r\rangle$ is $|0\rangle$, the value of the input 3 bits Gray code is in the [4, 7] interval. When the

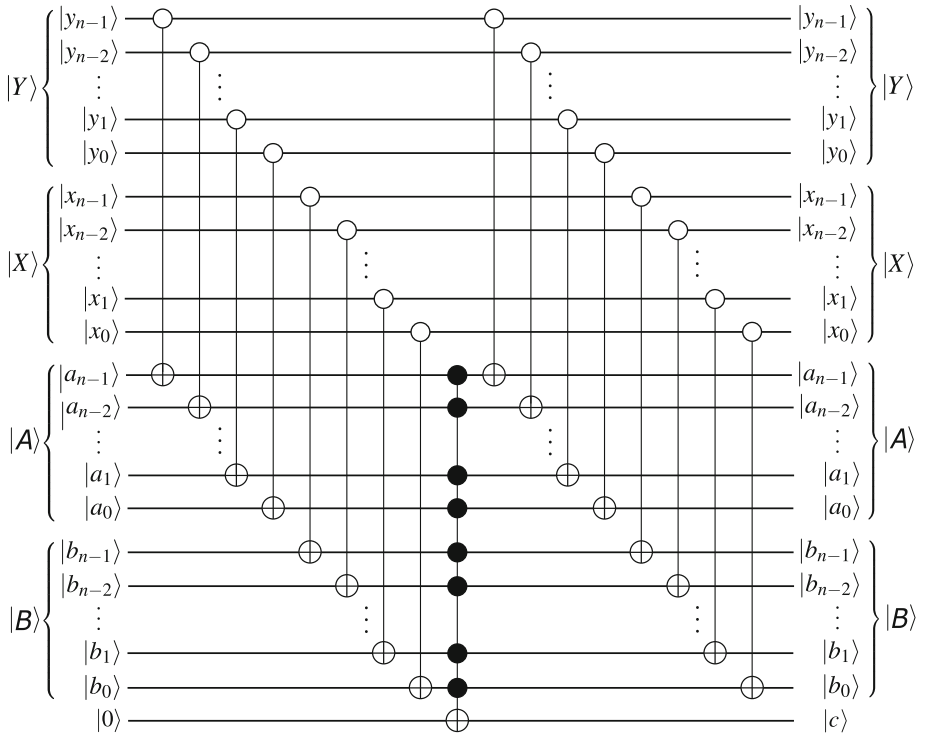


Fig. 6 Quantum comparator (QE) [30]

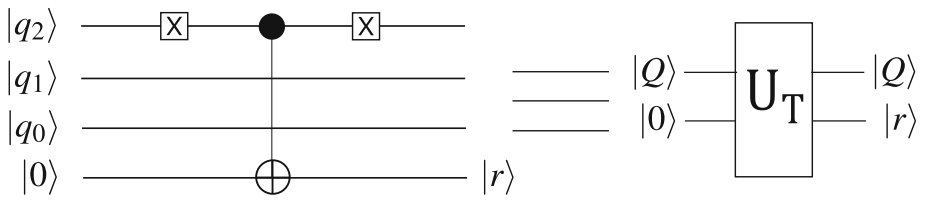


Fig. 7 Judgment circuit

output $|r\rangle$ is $|1\rangle$, it means that the value of the input three-digit Gray code is in the [0, 3] interval.

3 The Proposed Quantum Image Steganography Scheme

In this paper, the proposed quantum steganography scheme is to embed a $n \times 2n$ grayscale image into a $4n \times 4n$ grayscale image. Before embedding, a $4n \times 4n$ one-bit key image needs to be prepared. The NEQR representations of the information image, key image, and carrier image are respectively as follows:

$$|I\rangle = \frac{1}{2^n} \sum_{Y=0}^{2^{n-2}-1} \sum_{X=0}^{2^{n-1}-1} |I_{YX}\rangle |YX\rangle, \tag{8}$$

where $|I_{YX}\rangle = |i_{YX}^{q-1} \cdots i_{YX}^{1} i_{YX}^{0}\rangle$.

$$|K\rangle = \frac{1}{2^n} \sum_{Y=0}^{2^n-1} \sum_{X=0}^{2^n-1} |K_{YX}\rangle |YX\rangle, \tag{9}$$

where $K_{YX} \in \{0, 1\}$.

$$|C\rangle = \frac{1}{2^n} \sum_{Y=0}^{2^n-1} \sum_{X=0}^{2^n-1} |C_{YX}\rangle |YX\rangle, \tag{10}$$

where $|C_{YX}\rangle = |c_{YX}^{q-1} \cdots c_{YX}^{1} c_{YX}^{0}\rangle$.

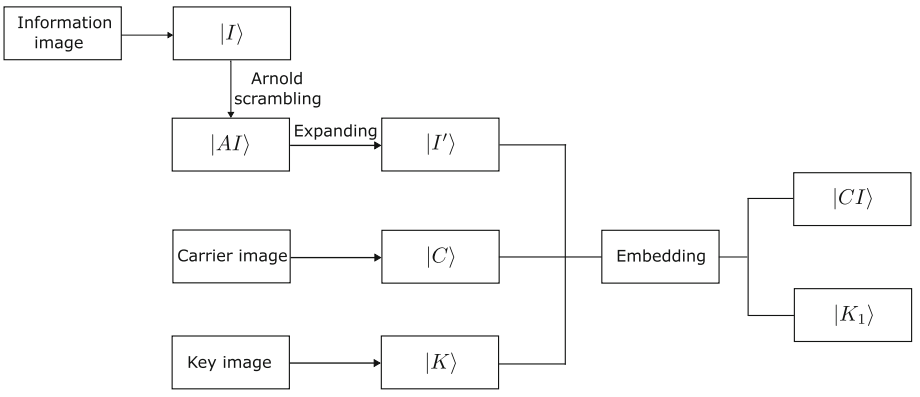


Fig. 8 Embedding process

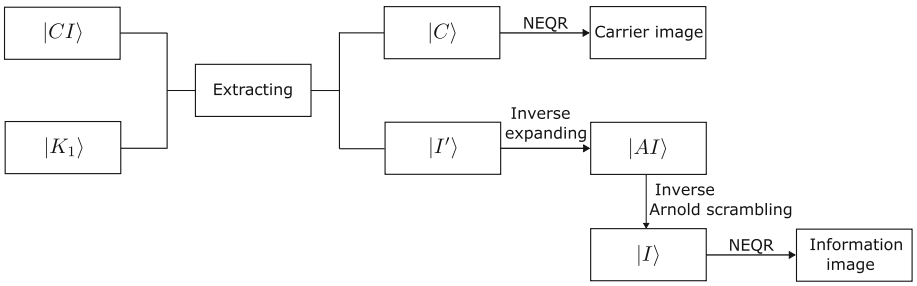


Fig. 9 Extracting process

The process of embedding and extracting are shown in Figs. 8 and 9.

3.1 Embedding Procedure

Based on the classical Gray code algorithm and the law of Gray code itself, this paper improves the classical Gray code algorithm. The concrete ideas of the improved Gray code algorithm are described in the following. In this paper, two layers of Gray code are set for the last three bits of the carrier pixels. The first layer of Gray code is $|c^2_{YX}c^1_{YX}c^0_{YX}\rangle$. The rule of the first layer of Gray code uses the rule of classical Gray codes described in Part 2, i.e., the judgment is based on the parity of Gray code. The second layer of Gray code is $|c^0_{YX}c^1_{YX}c^2_{YX}\rangle$. The rule for the second layer of Gray code is based on whether the value of the Gray code is within [0, 3] or within [4, 7], and the rule is as follows: If the information bit is 0 and the value of the second layer Gray code is in the interval [0, 3], then the LSB of the carrier pixel remains unchanged. In this case, the bit at the corresponding position of the key image is set to 1. If the information bit is 1 and the value of the second layer Gray code is in the interval [4, 7], then the LSB of the carrier pixel remains unchanged. In this case,

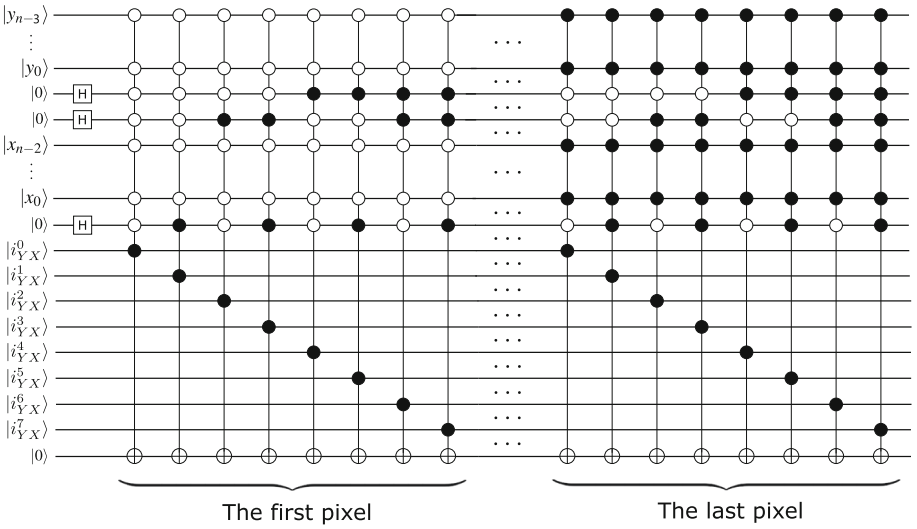


Fig. 10 Expansion circuit

the bit at the corresponding position of the key image is set to 1. The concrete embedding steps are as follows:

Step 1: The Arnold scrambling is applied to the $n \times 2n$ information image $|I\rangle$. The scrambled information image is $|AI\rangle$. Then the information image $|AI\rangle$ is expanded to a $4n \times 4n$ information image $|I'\rangle$. The quantum circuit of the expansion step is shown in Fig. 10.

Step 2: The first layer of the Gray code rule is applied to the $|c^2_{YX}c^1_{YX}c^0_{YX}\rangle$ of carrier pixel. If the LSB of the carrier pixel does not need to be changed, the operation on this pixel is ended. If the LSB of the carrier pixel needs to be changed, then go to step 3.

Step 3: It can be seen from step 2 that according to the rule of the first layer of Gray code, the LSB of the carrier pixel needs to be changed at this time. At this moment, the second layer of the Gray code rule is applied to the carrier pixel. If the LSB of the carrier pixel does not need to be changed according to the second layer of the Gray code rule, then its LSB is not changed and the corresponding position bit of the key image is set to 1. If the LSB of the carrier pixel still needs to be changed according to the second layer of the Gray code rule, then its LSB is changed according to the first layer of the Gray code rule.

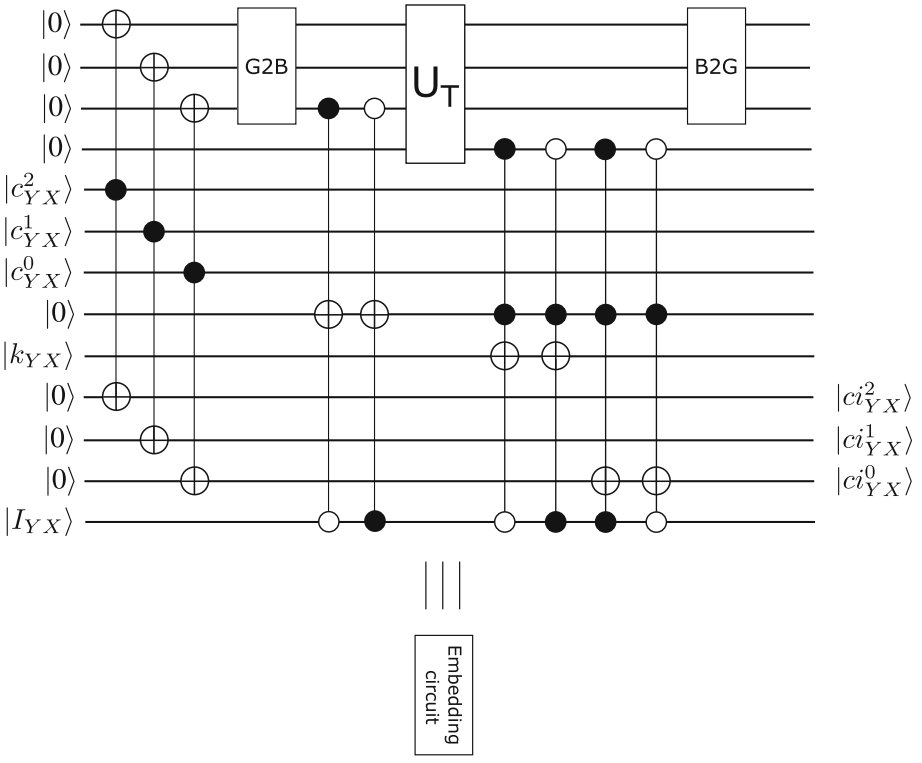


Fig. 11 Embedding module

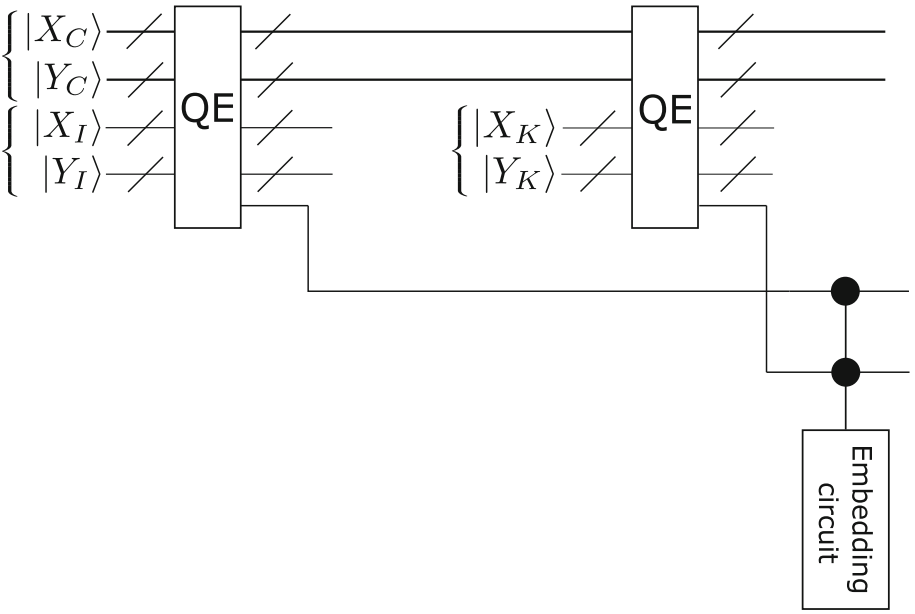


Fig. 12 Embedding circuit

The embedding algorithm is shown in (11) and (12). The embedding circuit is shown in Figs. 11 and 12.

$$|c_{YX}^2 c_{YX}^1 c_{YX}^0\rangle = |c_{YX}^2 c_{YX}^1 \overline{c_{YX}^0}\rangle \ , \quad While\ Gray\left(c_{YX}^2 c_{YX}^1 c_{YX}^0\right)\ mod\ 2 = 1 \bigcap Gray(c_{YX}^0 c_{YX}\ {}^1c_{YX}^2) \in [4,7] \bigcap I'_{YX} = 0. \tag{11}$$

$$|c_{YX}^2 c_{YX}^1 c_{YX}^0\rangle = |c_{YX}^2 c_{YX}^1 \overline{c_{YX}^0}\rangle, \quad While\ Gray\left(c_{YX}^2 c_{YX}^1 c_{YX}^0\right)\ mod\ 2 = 0 \bigcap Gray(c_{YX}^0 c_{YX}\ {}^1c_{YX}^2) \in [0,3] \bigcap I'_{YX} = 1. \tag{12}$$

For example, in Fig. 13, the 1×2 information image is first expanded into a 4×4 image. Assume that the carrier image is shown in Fig. 14. Take the first pixel $|10110100\rangle$ in the upper left corner as an example. At this time, the information bit in the corresponding position in the information image is 0. The first layer of the Gray code of $|10110100\rangle$ is $|100\rangle$. The value of the Gray code is 7, which is an odd number. The LSB needs to be changed according to the rule of the first layer of the Gray code, so the second layer of the Gray code rule is applied to it. The second layer of its Gray code is $|001\rangle$. The value of the Gray code is 1, which belongs to the range of [0, 3], so no change is needed. At this time, the corresponding position of the key image is set to 1.

Figure 15 summarizes the change of $|c_{YX}^2 c_{YX}^1 c_{YX}^0\rangle$ of the carrier pixel in different cases. It can be seen that the LSB of the carrier pixel needs to be changed only in a quarter of the cases, and the rest of the cases do not need to be changed. Therefore, the visual effect of the algorithm proposed in this paper will be better.

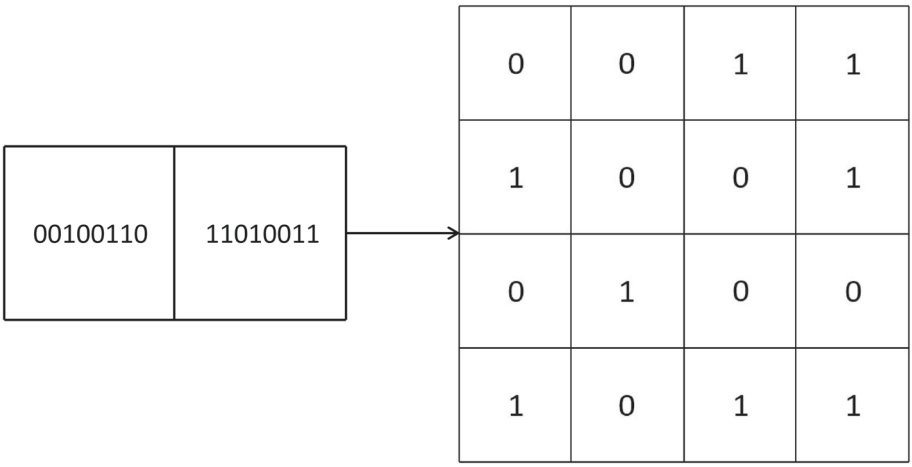


Fig. 13 Information image and its extension

3.2 Extracting Procedure

The extracting idea of the proposed Gray code algorithm in this paper is based on the key image and the last three bits of the carrier pixel. The quantum circuit of the extracting step is shown in Figs. 16 and 17. The concrete steps are as follows:

10110100	10010111	00101101	11011110
00100101	10110100	00111110	11110001
10000101	10110100	00100100	10100101
11110101	11010011	10110100	01000101

Fig. 14 Carrier image

Input: $|I'\rangle$, $|C\rangle$
Output: $|CI'\rangle$
The last three bits of $|C\rangle$ are $|c^2_{YX}c^1_{YX}c^0_{YX}\rangle$
for $Y = 0$ to $2^n - 2$ **do**
  for $X = 0$ to $2^n - 1$ **do**
    if $I'_{YX} = 1$ **then**
      if $Gray(c^2_{YX}c^1_{YX}c^0_{YX})\ mod\ 2 = 1$ **then**
        $|C\rangle$ doesn't need to change;
      else if $Gray(c^2_{YX}c^1_{YX}c^0_{YX})\ mod\ 2 = 0$ **then**
        if $Gray(c^0_{YX}c^1_{YX}c^2_{YX}) \in [4, 7]$ **then**
          $|C\rangle$ doesn't need to change;
        else if $Gray(c^0_{YX}c^1_{YX}c^2_{YX}) \in [0, 3]$ **then**
          $|c^2_{YX}c^1_{YX}c^0_{YX}\rangle = |c^2_{YX}c^1_{YX}\overline{c^0_{YX}}\rangle$
        end if
      end if
    else if $I'_{YX} = 0$ **then**
      if $Gray(c^2_{YX}c^1_{YX}c^0_{YX})\ mod\ 2 = 0$ **then**
        $|C\rangle$ doesn't need to change;
      else if $Gray(c^2_{YX}c^1_{YX}c^0_{YX})\ mod\ 2 = 1$ **then**
        if $Gray(c^0_{YX}c^1_{YX}c^2_{YX}) \in [0, 3]$ **then**
          $|C\rangle$ doesn't need to change;
        else if $Gray(c^0_{YX}c^1_{YX}c^2_{YX}) \in [4, 7]$ **then**
          $|c^2_{YX}c^1_{YX}c^0_{YX}\rangle = |c^2_{YX}c^1_{YX}\overline{c^0_{YX}}\rangle$
        end if
      end if
    end if
  end for
end for

Algorithm 1 Double-level Gray code embedding algorithm.

Step 1: Observe the value of the corresponding position of the key image and determine what extracting method to use according to its value.

Step 2: If the value of the corresponding position of the key image is 1, then the extracting is according to the value of the second layer of Gray code: If the value of the Gray code of the $|ci^0_{YX}ci^1_{YX}ci^2_{YX}\rangle$ of the carrier pixel belongs to the interval [0, 3], then the information bit is 0. If the value of the Gray code of the $|ci^0_{YX}ci^1_{YX}ci^2_{YX}\rangle$ of the carrier pixel belongs to the interval [4, 7], then the information bit is 1. If the value of the corresponding position of the key image is 0, then go to step 3.

Step 3: At this point, the extraction is performed according to the extraction rule of the classical Gray code. That is, If the value of the Gray code of the $|ci^2_{YX}ci^1_{YX}ci^0_{YX}\rangle$ of the carrier pixel is even, then the information bit is 0. If the value of the Gray code of the $|ci^2_{YX}ci^1_{YX}ci^0_{YX}\rangle$ of the carrier pixel is odd, then the information bit is 1.

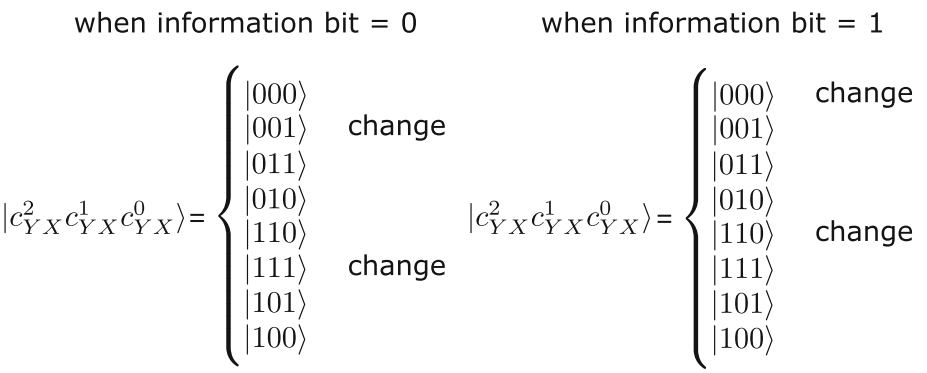


Fig. 15 Changes in carrier pixels

4 Computational Complexity

Circuit complexity is a metric to evaluate the quantum circuit of an algorithm. The circuit complexity of an algorithm depends on the number of various logic gates in the circuit diagram. The circuit complexity calculation method adopted in this paper is: The circuit complexity of a one-bit controlled-NOT gate is 1. The circuit complexity of a two-bit controlled-NOT gate is 6, and the circuit complexity of a multi-bit controlled-NOT gate ($n \geqslant 3$) is $12n - 9$. The circuit complexity of the swap gate is 3. According to the

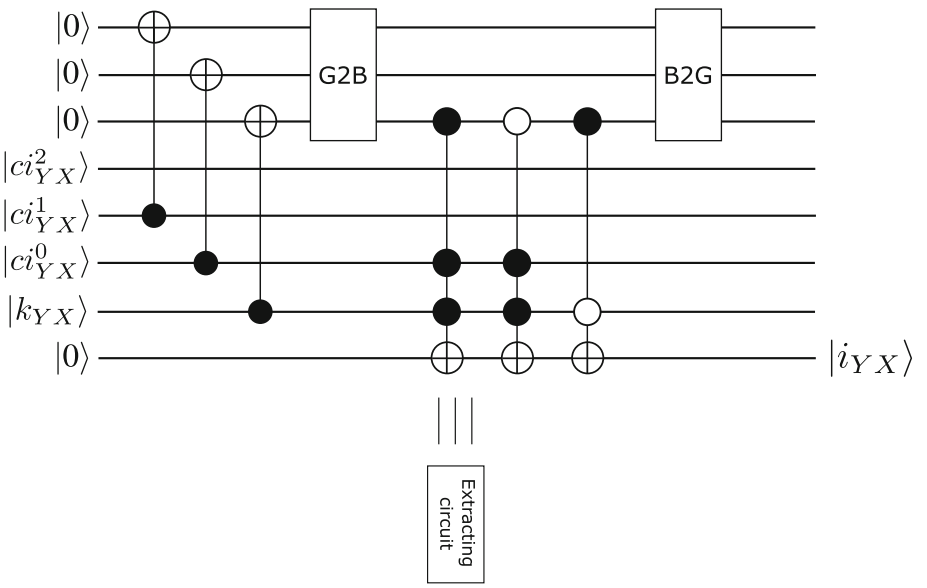


Fig. 16 Extracting module

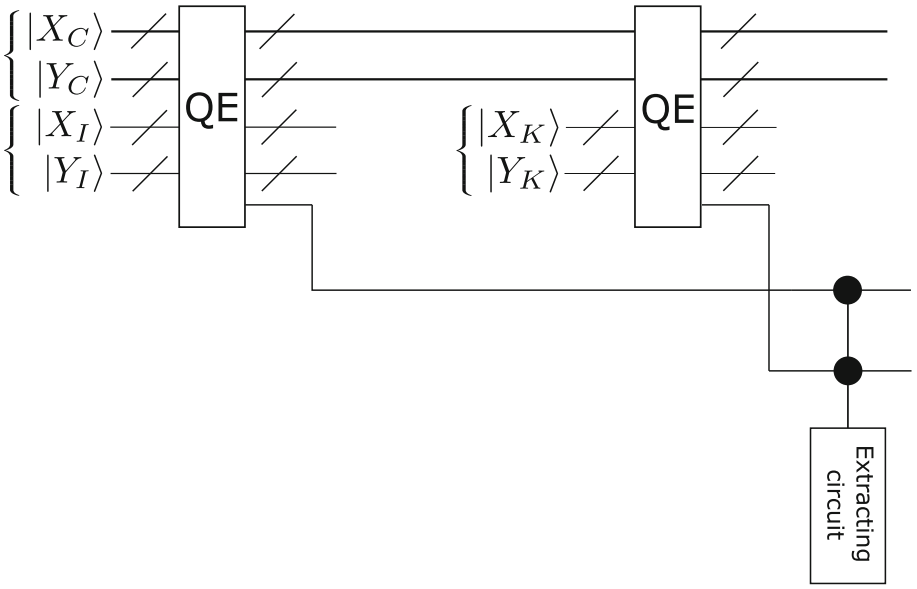


Fig. 17 Extracting circuit

embedding and extracting ideas mentioned before, the size of the information image and the carrier image are $2^{n-2} \times 2^{n-1}$ and $2^n \times 2^n$, respectively. The circuit complexity is calculated in two parts: embedding and extracting.

Embedding section: The embedding part consists of Arnold scrambling, image expansion, and embedding. The circuit complexity of Arnold scrambling is $56n$ [29]. Figures 4 and 5 conversion circuits have a circuit complexity of 2 each. The circuit complexity of the comparison circuit in Fig. 6 is $28n - 9$. The judgment circuit of Fig. 7 has a circuit complexity of 1. In Fig. 10, the circuit complexity of the expansion circuit is $(24n + 3) \cdot 2^{2n}$. The embedding module of Fig. 11 has a circuit complexity of 131. The circuit complexity of the embedding circuit in Fig. 12 is $56n + 113$. Therefore, the overall circuit complexity of the embedding part is $O(24n \cdot 2^{2n})$.

Extracting section: The extracting section consists of extracting circuit, reducing circuit, and Arnold inverse scrambling. The circuit complexity of the extracting module in Fig. 16 is 88. In Fig. 17, the circuit complexity of the extracting circuit is $56n + 70$. The circuit complexity of the reducing circuit is the same as that of the expansion circuit. The circuit complexity of Arnold inverse scrambling is $112n$ [29]. Therefore, the overall circuit complexity of the extracting section is $O(24n \cdot 2^{2n})$.

If the image scaling is not considered, the circuit complexity of the embedding circuit of the quantum image steganography algorithm proposed in this paper is $O(n)$. Similarly, the circuit complexity of the extracted circuit is $O(n)$. Therefore, the overall circuit complexity is $O(n)$. In classical image processing, the complexity is related to the size of the image that needs to be processed. Taking the image used in the algorithm of this paper as an example, the complexity in classical image processing is $O(2^{2n})$. Therefore, for the proposed algorithm in this paper, quantum image processing has significant advantages in speed compared to classical image processing.

5 Experiments and Analysis

In this section, the algorithm proposed in this paper will be experimented from two aspects: visual effect and robustness. The experimental results are compared with other algorithms. The embedding capacity of different algorthims and the auxiliary images used in different algorithms are different. In this paper, some algorithms suitable for comparison are selected in the visual effect experiment and robustness experiment, respectively. Then the advantages and disadvantages of the algorithm proposed in this paper are derived. The size of the carrier image used in the experiment is 512×512, and the information image uses 2 images of 128×128 stitched together to form a 128×256 image.

5.1 Visual Quality

The visual effect is a key element to judge the merits of a steganography algorithm. Calculating PSNR is a common way to verify the visual effect of a steganography algorithm. By comparing the carrier image after embedding the information image with the original carrier image, the similarity of the two is derived. Thus, the actual effectiveness of a steganographic algorithm is illustrated. The formula of PSNR is as follows:

$$PSNR = 10 \log_{10} \frac{MAX_I^2}{MSE} = 20 \log \frac{MAX_I}{\sqrt{MSE}} \tag{13}$$

where MSE is the mean squared error with the following formula:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - J(i, j)]^2 \tag{14}$$

The pixel histogram provides a visual representation of the number of pixels of various colors in an image. By comparing the histograms of the two images it is clear how similar they are. Figure 18 shows the comparison of the visual effect between the carrier image after embedding the information image by the algorithm proposed in this paper and the original image. The images used in the experiment are shown in Fig. 19. Figure 20 shows the visual effect of the carrier image embedded by the classical Gray code algorithm compared with the original image. The histogram shows that the algorithm proposed in this paper has a significantly smaller change on the carrier image.

Table 1 shows the PSNR values of the carrier images after embedding the information images in each group of experiments. It can be seen that the PSNR values of each group of experiments have reached around 54dB. Table 2 shows the comparison of the visual effect of the algorithm proposed in this paper with other algorithms. Among them, the algorithm of Zhou et al. [30] is better than the algorithm proposed in this paper in terms of embedding capacity but uses more key images. Meanwhile, the visual effect is slightly lower than the algorithm proposed in this paper. The algorithm of Chen and Chang [26] is the classical Gray code algorithm, which is less visually effective than the algorithm proposed in this paper. Chatterjee et al. [31] have done multiple sets of experiments with different embedding capacities. Here the experimental data with the same embedding capacity as in this paper is selected for comparison. It can be seen that the algorithm

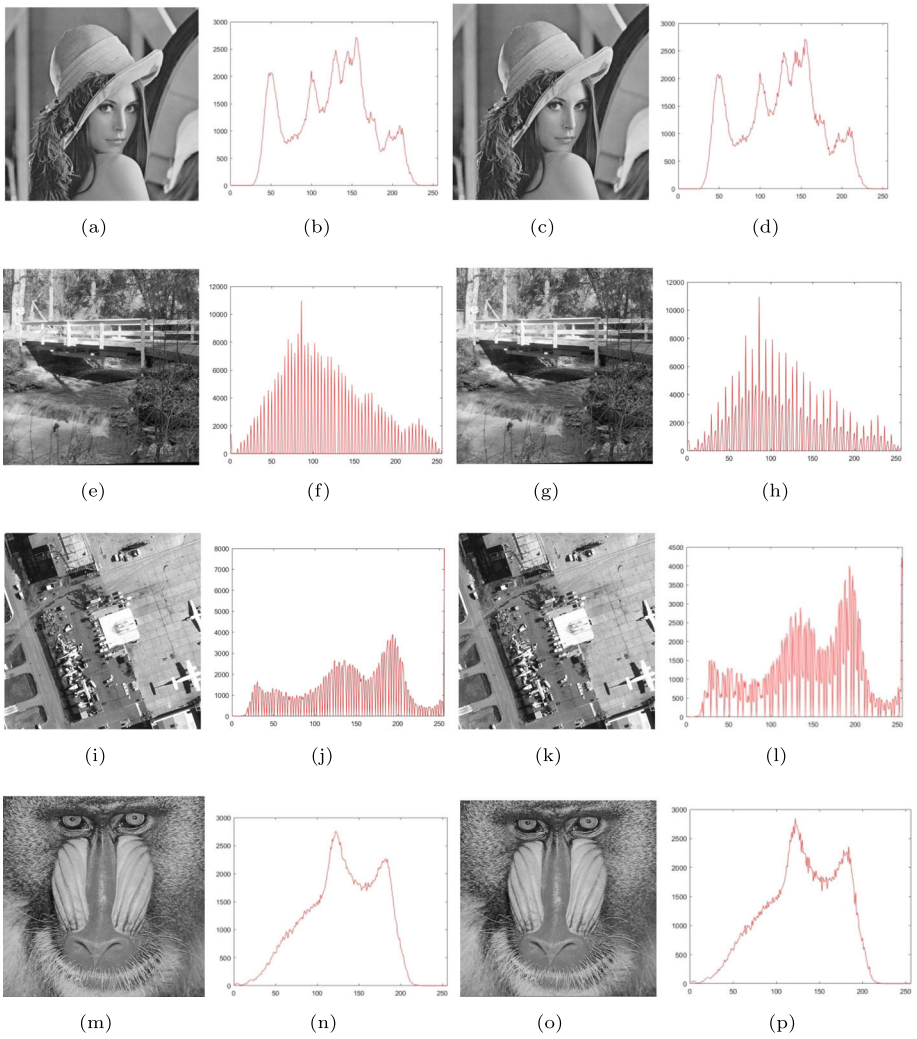


Fig. 18 Each row of images from left to right is the original carrier image, the histogram of the original carrier image, the carrier image after embedding information (the algorithm proposed in this paper) , and the histogram of the carrier image after embedding information (the algorithm proposed in this paper)

proposed in this paper is superior to its algorithm in terms of visual effect. Compared with the algorithm of Zhou et al. [21], the algorithm proposed in this paper performs better in all aspects. The algorithm of Luo et al. [22] has a higher embedding capacity than the algorithm proposed in this paper. However, the algorithm proposed in this paper has a clear advantage in visual effects.

Tables 3 and 4 show the comparison between the algorithm proposed in this paper and the classical Gray code algorithm in terms of the number of specific changed bits. It can be seen that the proportion of LSB changed by the proposed algorithm is about 25%, which is greatly improved compared with the 50% change of the classical Gray code algorithm.

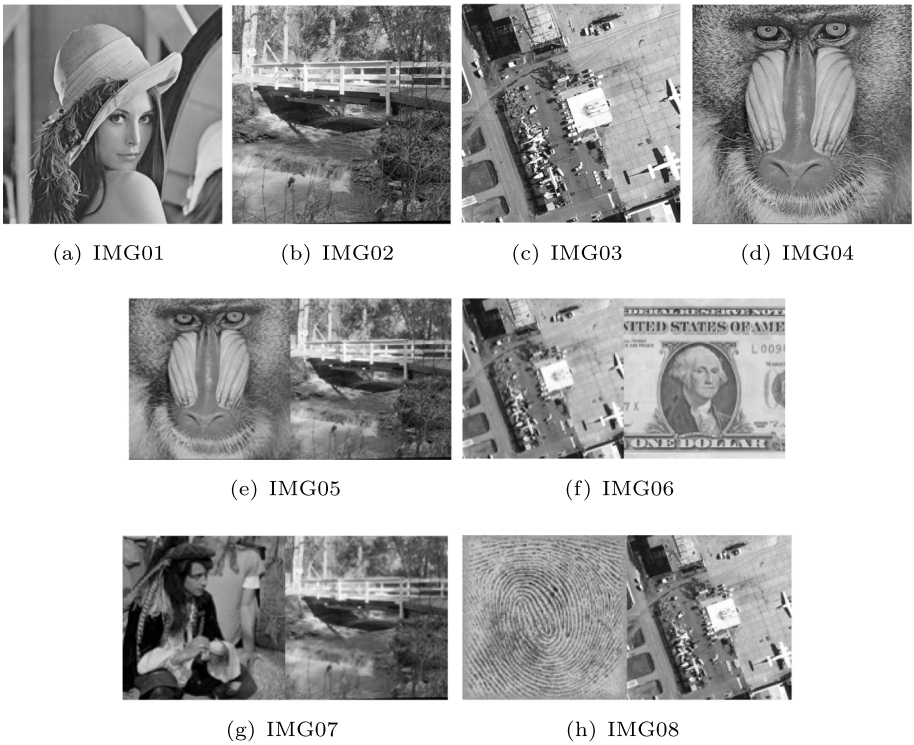


Fig. 19 Images used in the visual experiment

5.2 Robustness

Robustness is an important index to judge the stability of a steganography algorithm. By performing a noise attack on the image, the similarity between the information image extracted from the attacked carrier image and the original information image is calculated. Thus, it is judged whether a steganographic algorithm has good stability or not. Researchers usually choose salt-and-pepper noise as an attack tool in robustness experiments within the space domain. Salt-and-pepper noise is a type of noise caused by the intensity of the signal pulse. Salt-and-pepper noise is represented as discrete distributions of pure white pixels or pure black pixels. Salt represents white and pepper represents black. Since it is unlikely that the maximum/minimum value of grayscale pixels will appear in the image under normal circumstances. Therefore such pixels can be treated as noise. The noise density of the salt-and-pepper noise represents the percentage of image pixels that are attacked in the image. The experiments in this paper were conducted using salt-and-pepper noise with noise densities of 0.02, 0.05, 0.1, and 0.15, i.e., attacking 2%, 5%, 10%, and 15% of the carrier pixels. In this paper, by performing four different parameters of salt-and-pepper noise attack on the embedded image, then extracting the information image from the attacked carrier image and calculating its PSNR with the original information image. Take the fifth group of experiments as an example. The carrier images after the salt-and-pepper noise attack with noise densities of 0.02, 0.05, 0.1, and 0.15 and the information images extracted from them are shown in Fig. 21. It can be seen that the extracted image has a high similarity to the original

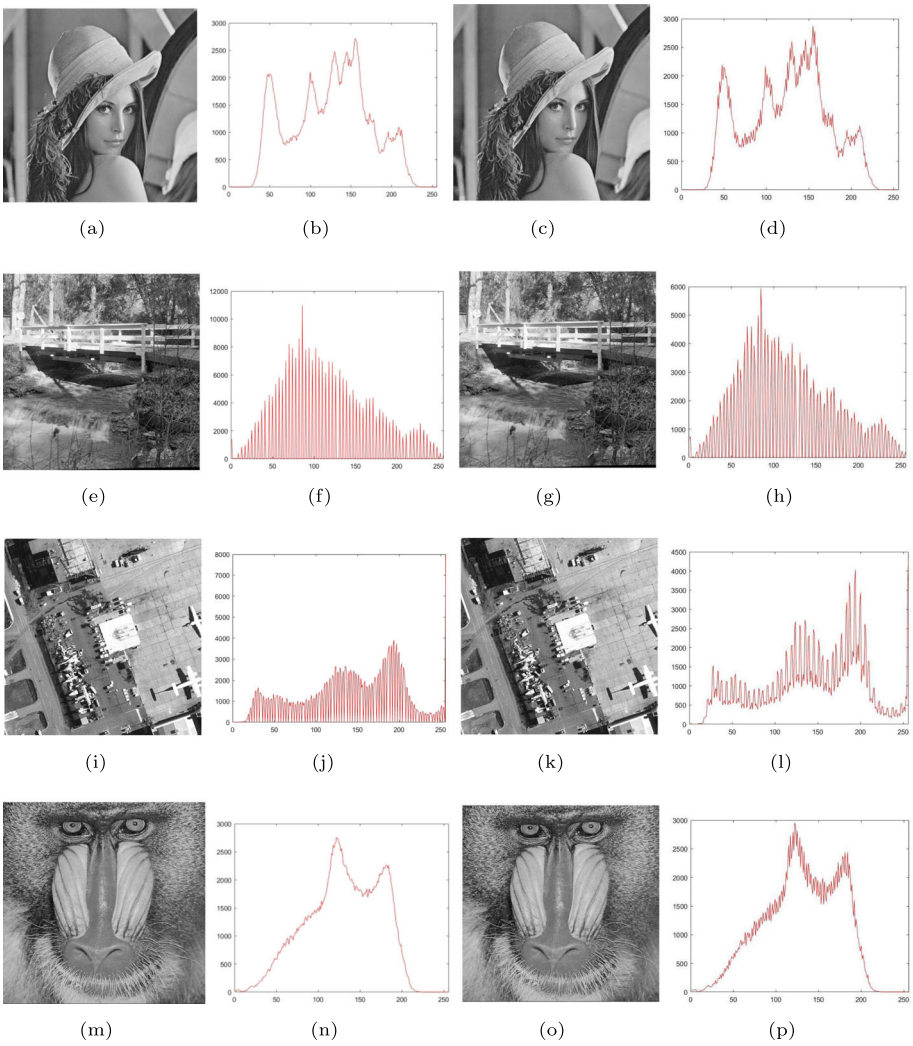


Fig. 20 Each row of images from left to right is the original carrier image, the histogram of the original carrier image, the carrier image after embedding information (the classical Gray code algorithm) , and the histogram of the carrier image after embedding information (the classical Gray code algorithm)

information image. Table 5 shows the PSNR values of the extracted information image and the original information image.

Since Chen and Chang [26] did not do robustness experiments, the algorithm proposed in this paper is compared with other algorithms. Table 6 shows the robustness of the algorithm proposed in this paper compared with other algorithms. Taking the salt and pepper noise of 0.05 as an example, the PSNR value of the algorithm proposed in this paper is 0.26 higher than that of Luo et al. [20], 6.92 higher than that of Luo et al. [32], and 10.22 higher than that of Zhou et al. [33]. Thus, it can be concluded that the algorithm proposed in this paper has good resistance in the face of attacks.

Table 1 The PSNR values of the carrier images after embedding the information images in each group of experiments

Carrier image	Information image	PSNR (dB)
IMG01	IMG05	54.1250
IMG02	IMG06	54.4780
IMG03	IMG07	53.8731
IMG04	IMG08	54.1657

Table 2 Comparison with other algorithms

Algorithm	PSNR (dB)
Proposed scheme	54.1605
Zhou et al. [30]	54.0467
Chen and Chang [26]	51.1430
Chatterjee et al. [31]	51.1070
Zhou et al. [21]	50.2267
Luo et al. [22]	48.7346

Table 3 The number of bits specifically changed using the algorithm proposed in this paper

Carrier image	Information image	Number of bits changed	Percentage of LSB
IMG01	IMG05	65802	25.10%
IMG02	IMG06	60788	23.19%
IMG03	IMG07	69873	26.65%
IMG04	IMG08	65320	24.92%

Table 4 The number of bits specifically changed using the classical Gray code algorithm

Carrier image	Information image	Number of bits changed	Percentage of LSB
IMG01	IMG05	121931	46.51%
IMG02	IMG06	126455	48.24%
IMG03	IMG07	120762	46.07%
IMG04	IMG08	126651	48.31%

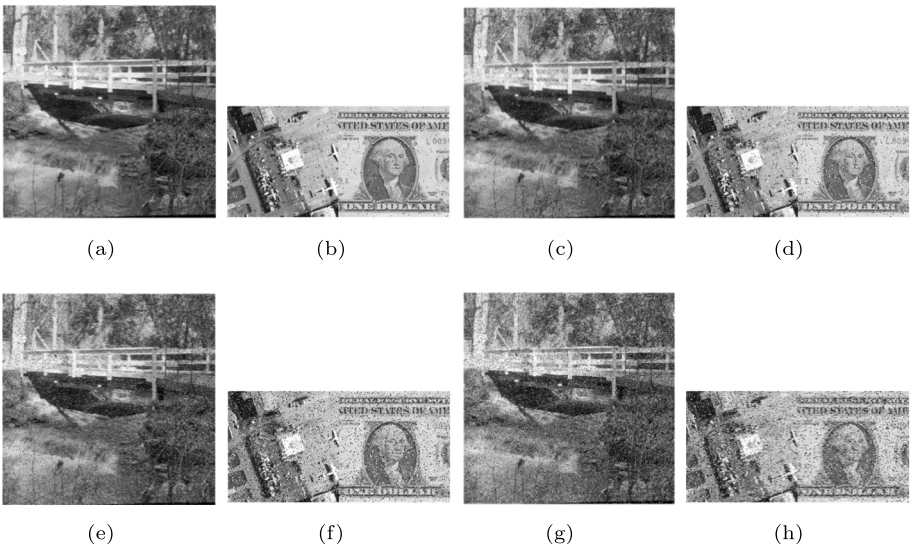


Fig. 21 Images at noise densities of 0.02, 0.05, 0.1, and 0.15 in the noise experiment, respectively

5.2.1 Security Analysis

The algorithm proposed in this paper uses the Gray code, therefore the information image cannot be extracted without knowing the Gray code rules. The extraction step of the algorithm proposed in this paper requires the key image, which has two roles in the algorithm proposed in this paper. One role is that others cannot accurately read the information embedded in this algorithm without access to the key image. The other role is that even if the carrier image after embedding information image is attacked, the key image changed during embedding can help us extract as much information as possible. In conclusion, the steganography algorithm proposed in this paper has relatively good security.

5.3 Conclusions

In this paper, a novel quantum image steganography algorithm based on double-layer Gray code is proposed based on the classical Gray code algorithm. The main advantage of the steganography algorithm proposed in this paper is the drastic reduction in the number of bits that need to be changed in the carrier image during embedding. The classical LSB

Table 5 Salt-and-pepper noise attack experiment (PSNR dB)

		Noise intensity			
Information image	Carrier image	0.02	0.05	0.1	0.15
IMG05	IMG01	40.7212	36.9638	34.0078	32.5239
IMG06	IMG02	41.2180	37.5818	34.9324	33.3682
IMG07	IMG03	40.2866	36.5711	33.6388	32.2233
IMG08	IMG04	41.0454	37.2104	34.4749	32.8607

Table 6 The robustness of the algorithm proposed in this paper (PSNR dB)

	Noise intensity		
Algorithm	0.05	0.1	0.15
Proposed scheme	37.5818	34.9324	33.3682
Luo et al. [20]	37.32	34.54	32.87
Luo et al. [32]	30.6610	29.3511	29.4198
Zhou et al. [33]	27.3591	27.3545	27.4097

algorithm needs to replace 100% bits of the LSB of the carrier pixels in the embedding step. The classical Gray code algorithm needs to change 50% bits of the LSB of the carrier pixels when embedding. The proposed algorithm in this paper only needs to change 25% bits of the LSB of the carrier pixels when embedding. Therefore, the algorithm proposed in this paper has a better visual effect. In addition, the proposed algorithm also has good performance in robustness and security.

Author Contributions All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by [Jin-Liang Yao], [Hong-Mei Yang], [Dong-Huan Jiang], [Bin Yan], [Jeng-Shyang Pan], [Meng-Xi Wang]. The first draft of the manuscript was written by [Jin-Liang Yao] and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Data Availability The datasets analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Competing interests The authors have no relevant financial or non-financial interests to disclose.

References

1. Venegas-Andraca, S.E., Bose, S.: Storing, processing, and retrieving an image using quantum mechanics. In: Quantum Information and Computation, vol. 5105, pp. 137–147. SPIE (2003)
2. Latorre, J.I.: Image compression and entanglement. arXiv:quant-ph/0510031 (2005)
3. Venegas-Andraca, S.E., Ball, J.: Processing images in entangled quantum systems. Quantum Inf. Process **9**(1), 1–11 (2010)
4. Le, P.Q., 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)
5. 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)
6. Li, H.-S., Zhu, Q., Zhou, R.-G., Song, L., Yang, X.-J.: Multi-dimensional color image storage and retrieval for a normal arbitrary quantum superposition state. Quantum Inf. Process **13**(4), 991–1011 (2014)
7. Jiang, N., Wang, L.: Quantum image scaling using nearest neighbor interpolation. Quantum Inf. Process **14**(5), 1559–1571 (2015)
8. Jiang, N., Wu, W., Wang, L., Zhao, N.: Quantum image pseudocolor coding based on the density-stratified method. Quantum Inf. Process **14**(5), 1735–1755 (2015)
9. Jiang, N., Wang, L.: Quantum image scaling using nearest neighbor interpolation. Quantum Inf. Process **14**(5), 1559–1571 (2015)

10. Zhou, R.-G., Cheng, Y., Liu, D.: Quantum image scaling based on bilinear interpolation with arbitrary scaling ratio. Quantum Inf. Process **18**(9), 1–19 (2019)
11. Zhou, R.-G., Sun, Y.-J., Fan, P.: Quantum image Gray-code and bit-plane scrambling. Quantum Inf. Process **14**(5), 1717–1734 (2015)
12. Jiang, N., Wang, L., Wu, W.-Y.: Quantum Hilbert image scrambling. Int. J. Theor. Phys. **53**(7), 2463–2484 (2014)
13. Zhou, R.-G., Hu, W., Fan, P., Luo, G.: Quantum color image watermarking based on Arnold transformation and LSB steganography. Int. J. Quantum Inf. **16**(03), 1850021 (2018)
14. Hu, W.-W., Zhou, R.-G., El-Rafei, A., Jiang, S.-X.: Quantum image watermarking algorithm based on Haar wavelet transform. IEEE Access **7**, 121303–121320 (2019)
15. Zhou, R.-G., Hu, W.W., Luo, G.F., Fan, P., Ian, H.: Optimal LSBs-based quantum watermarking with lower distortion. Int. J. Quantum Inf. **16**(07), 1850058 (2018)
16. Qu, Z., Cheng, Z., Liu, W., Wang, X.: A novel quantum image steganography algorithm based on exploiting modification direction. Multimed. Tools Appl. **78**(7), 7981–8001 (2019)
17. Li, P., Liu, X.: A novel quantum steganography scheme for color images. Int. J. Quantum Inf. **16**(02), 1850020 (2018)
18. Jiang, N., Dang, Y., Wang, J.: Quantum image matching. Quantum Inf. Process **15**(9), 3543–3572 (2016)
19. Jiang, N., Zhao, N., Wang, L.: LSB based quantum image steganography algorithm. Int. J. Theor. Phys. **55**(1), 107–123 (2016)
20. Luo, G., Zhou, R.-G., Luo, J., Hu, W., Zhou, Y., Ian, H.: Adaptive LSB quantum watermarking method using tri-way pixel value differencing. Quantum Inf. Process **18**(2), 1–20 (2019)
21. Zhou, R.-G., Luo, J., Liu, X., Zhu, C., Wei, L., Zhang, X.: A novel quantum image steganography scheme based on LSB. Int. J. Theor. Phys. **57**(6), 1848–1863 (2018)
22. Luo, G., Zhou, R.-G., Mao, Y.: Two-level information hiding for quantum images using optimal LSB. Quantum Inf. Process **18**(10), 1–19 (2019)
23. Zhou, Y., Zhou, R.-G., Liu, X., Luo, G.: A quantum image watermarking scheme based on two-bit superposition. Int. J. Theor. Phys. **58**(3), 950–968 (2019)
24. Qu, Z., Sun, H., Zheng, M.: An efficient quantum image steganography protocol based on improved EMD algorithm. Quantum Inf. Process **20**(2), 1–29 (2021)
25. Zeng, Q.-W., Wen, Z.-Y., Fu, J.-F., Zhou, N.-R.: Quantum watermark algorithm based on maximum pixel difference and tent map. Int. J. Theor. Phys. **60**(9), 3306–3333 (2021)
26. Chen, C.-C., Chang, C.-C.: LSB-Based steganography using reflected gray code. IEICE Trans. Inf. Syst. **91**(4), 1110–1116 (2008)
27. Tirkel, A.Z., Rankin, G., Van Schyndel, R., Ho, W., Mee, N., Osborne, C.F.: Electronic watermark. Digital Image Comput. Technol. Appl. (DICTA'93):666–673 (1993)
28. Dyson, F.J., Falk, H.: Period of a discrete cat mapping. Am. Math. Mon. **99**(7), 603–614 (1992)
29. Jiang, N., Wang, L.: Analysis and improvement of the quantum Arnold image scrambling. Quantum Inf. Process **13**(7), 1545–1551 (2014)
30. Zhou, R.-G., Hu, W., Fan, P.: Quantum watermarking scheme through Arnold scrambling and LSB steganography. Quantum Inf. Process **16**(9), 1–21 (2017)
31. Chatterjee, A., Ghosal, S.K., Sarkar, R.: LSB Based steganography with OCR: an intelligent amalgamation. Multimed. Tools Appl. **79**(17), 11747–11765 (2020)
32. Luo, G., Ling, M.: Novel watermarking scheme using boundary pixels least significant qubit steganography. In: 2019 3rd International Conference on Data Science and Business Analytics (ICDSBA), pp. 375–378. IEEE (2019)
33. Luo, G., Zhou, R.-G., Hu, W., Luo, J., Liu, X., Ian, H.: Enhanced least significant qubit watermarking scheme for quantum images. Quantum Inf. Process **17**(11), 1–19 (2018)

Affiliations

Jin-Liang Yao[1] · Hong-Mei Yang[1] · Dong-Huan Jiang[2] · Bin Yan[3] · Jeng-Shyang Pan[1] · Meng-Xi Wang[1]

Jin-Liang Yao
yaojinliang@sdust.edu.cn

Dong-Huan Jiang
donghuan_jiang@163.com

Bin Yan
yanbinhit@sdust.edu.cn

Jeng-Shyang Pan
jspan@cc.kuas.edu.tw

Meng-Xi Wang
wangmengxi@sdust.edu.cn

1 College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao, 266590, Shandong, People's Republic of China

2 College of Mathematics and Systems Science, Shandong University of Science and Technology, Qingdao, 266590, Shandong, People's Republic of China

3 College of Electronic and Information Engineering, Shandong University of Science and Technology, Qingdao, 266590, Shandong, People's Republic of China