Quantum Inf Process (2011) 10:63–84
DOI 10.1007/s11128-010-0177-y

A flexible representation of quantum images for polynomial preparation, image compression, and processing operations

Phuc Q. Le · Fangyan Dong · Kaoru Hirota

Received: 12 October 2009 / Accepted: 5 April 2010 / Published online: 17 April 2010

Abstract A Flexible Representation of Quantum Images (FRQI) is proposed to provide a representation for images on quantum computers in the form of a normalized state which captures information about colors and their corresponding positions in the images. A constructive polynomial preparation for the FRQI state from an initial state, an algorithm for quantum image compression (QIC), and processing operations for quantum images are combined to build the whole process for quantum image processing on FRQI. The simulation experiments on FRQI include storing, retrieving of images and a detection of a line in binary images by applying quantum Fourier transform as a processing operation. The compression ratios of QIC between groups of same color positions range from 68.75 to 90.63% on single digit images and 6.67–31.62% on the Lena image. The FRQI provides a foundation not only to express images but also to explore theoretical and practical aspects of image processing on quantum computers.

Keywords Quantum computation · Image representation · Image processing · Image compression · Quantum Fourier transform

1 Introduction

In 1982, Feynman proposed a novel computation model, named quantum computers [6], based on principles of quantum physics that seemed to be more powerful than classical ones. After that Shor's polynomial time algorithm for the integer factoring

P. Q. Le (✉) · F. Dong · K. Hirota
Department of Computational Intelligence and Systems Science, Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology, G3-49, 4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan
e-mail: phuclq@hrt.dis.titech.ac.jp; phuclevn@gmail.com

problem [15] and Grover's database search algorithm [8] were essential evidences supporting the power of quantum computers. Quantum computation has appeared in various areas of computer science such as information theory, cryptography, image processing, etc. [14] because there are inefficient tasks on classical computers that can be overcomed by exploiting the power of the quantum computation. Processing and analysis of images in particular and visual information in general on classical computers have been studied extensively. On quantum computers, the research on images has faced fundamental difficulties because the field is still in its infancy. To start with, what are quantum images or how do we represent images on quantum computers? Secondly, what should we do to prepare and process the quantum images on quantum computers?

Research in the field of quantum image processing started with proposals on quantum image representations such as Qubit Lattice [17,18], Real Ket [10] and quantum transforms related to image processing such as quantum Fourier transform [14], quantum discrete cosine transform [9,16], quantum Wavelet transform [7]. The quantum images are two dimensional arrays of qubits in Ref. [17,18] and a quantum state in Ref. [10]. The quantum versions of classical image processing transforms such as Fourier transform, discrete cosine transform, etc., are more efficient in quantum computation than classical ones [14]. Quantum algorithms have been applied to classical image processing problems because of their proven efficiency over the classical versions [2,4,5]. In addition, there are some classical image processing operations that can not be applied on quantum images, for example convolution and correlation [11], because all operations in quantum computation must be invertible. The complexity of the preparation of quantum images and the application of quantum transforms to process quantum images, however, have not been studied.

In this research, a flexible representation of quantum images (FRQI) which captures information about colors and their corresponding positions in an image into a normalized quantum state is proposed. After the proposal of FRQI, the paper studied the following computational and image processing aspects on FRQI:

- The complexity (the number of simple operations) of the preparation for FRQI,
- The method to reduce number of simple operations that are used in the FRQI preparation step or quantum image compression (QIC),
- Three types of invertible image processing operators on FRQI.

The preparation process for FRQI is indicated by using Hadamard and controlled rotation operations. As proven by the Polynomial Preparation theorem, the total number of simple operations used in the process is polynomial for the number of qubits which are used to encode all positions in an image. Considering colors indistinguishable to human vision, the QIC algorithm reduces the number of simple operations in groups of the same color positions by integrating controlled part of the controlled rotations in the groups. Processing operators on FRQI based on unitary transforms are divided into three types; dealing with only colors, colors at some positions and the combination of both colors and positions. The simulation experiments confirm the capacity of FRQI on storage and retrieval quantum images, compression ratio among the same color groups on QIC algorithm and an application of an image processing operator for the

third type, using quantum Fourier transform, for a line detection in binary images on quantum computers.

These results indicate that the FRQI can be the basis to represent and process quantum images. The preparation and processing operations on FRQI improve the whole procedure of quantum image processing, i.e. a quantum computer starts from its initial state, then it is prepared to the FRQI state, and it is finally transformed by processing operations. The QIC algorithm suggests a way to reduce the main resources used to represent quantum images and can be extended for better quantum compression methods on quantum images. The three types of processing operations point out patterns for designing and applying other operations on quantum images.

2 Flexible representation of quantum images and its polynomial preparation

Inspired by the pixel representation for images in conventional computers, a representation for images on quantum computers capturing information about colors and their corresponding positions, the flexible representation of quantum images is proposed. This proposal integrates information about an image into a quantum state having its formula in (1)

$$|I(\theta)\rangle = \frac{1}{2^n}\sum_{i=0}^{2^{2n}-1}(\cos\theta_i|0\rangle + \sin\theta_i|1\rangle)\otimes|i\rangle, \tag{1}$$

$$\theta_i \in \left[0, \frac{\pi}{2}\right],\ i = 0, 1, \ldots, 2^{2n}-1, \tag{2}$$

capturing information about colors and the corresponding positions of those colors, where $\otimes$ is the tensor product notation, $|0\rangle, |1\rangle$ are 2-D computational basis quantum states, $|i\rangle, i = 0, 1, \ldots, 2^{2n}-1$ are 2^{2n}-D computational basis quantum states and $\theta = (\theta_0, \theta_1, \ldots, \theta_{2^{2n}-1})$ is the vector of angles encoding colors. There are two parts in the FRQI representation of an image; $\cos\theta_i|0\rangle + \sin\theta_i|1\rangle$ which encodes the information about colors and $|i\rangle$ that about the corresponding positions in the image, respectively. An example of a 2×2 image is shown in Fig. 1. The FRQI state is a normalized state, i.e. $\||I(\theta)\rangle\| = 1$ as given by

$$\||I(\theta)\rangle\| = \frac{1}{2^n}\sqrt{\sum_{i=0}^{2^{2n}-1}(\cos^2\theta_i + \sin^2\theta_i)} = 1. \tag{3}$$

$$|I\rangle = \frac{1}{2}\left[(\cos\theta_0|0\rangle + \sin\theta_0|1\rangle)\otimes|00\rangle + (\cos\theta_1|0\rangle + \sin\theta_1|1\rangle)\otimes|01\rangle + (\cos\theta_2|0\rangle + \sin\theta_2|1\rangle)\otimes|10\rangle + (\cos\theta_3|0\rangle + \sin\theta_3|1\rangle)\otimes|11\rangle\right]$$

Fig. 1 A simple image and its FRQI state

θ_0 0000	θ_1 0001	θ_2 0010	θ_3 0011
θ_4 0100	θ_5 0101	θ_6 0110	θ_7 0111
θ_8 1000	θ_9 1001	θ_{10} 1010	θ_{11} 1011
θ_{12} 1100	θ_{13} 1101	θ_{14} 1110	θ_{15} 1111

θ_0 0000	θ_1 0001	θ_2 0100	θ_3 0101
θ_4 0010	θ_5 0011	θ_6 0110	θ_7 0111
θ_8 1000	θ_9 1001	θ_{10} 1100	θ_{11} 1101
θ_{12} 1010	θ_{13} 1011	θ_{14} 1110	θ_{15} 1111

Fig. 2 Two position encoding methods for *colors*

The proposed FRQI form is quite flexible because of the way the positions of colors are encoded into computational basis states. In this way, the presentation of the geometric appearance of colors will affect on the quantum representation of the image. For example, the line by line and block based addressing methods are some of the encoding mechanisms commonly used. These mechanisms are shown in Fig. 2.

In quantum computation, computers are usually initialized in well-prepared states. As a result, the preparation process that transforms quantum computers from the initialized state to the desired quantum image state is necessary. All transforms used in quantum computation are unitary transforms described by unitary matrices. A matrix is said to be unitary if its Hermitian conjugate or its adjoint is the same as its inverse. Quantum mechanics ensure the existence of such unitary transforms for the preparation process without pointing out explicitly efficient implementation in the sense of using only simple transforms such as Hadamard transform, rotations, etc. The polynomial preparation theorem (PPT) as developed by using Lemma 1 and Corollary 1 shows a constructively efficient implementation of the preparation process.

Lemma 1 *Given a vector* $\theta = (\theta_0, \theta_1, \ldots, \theta_{2^{2n}-1})(n \in \mathbb{N})$ *of angles satisfying* (2)*, there is a unitary transform* $\mathcal{P}$ *that turns quantum computers from the initialized state,* $|0\rangle^{\otimes 2n+1}$*, to the FRQI state,* $|I(\theta)\rangle$*, composed by Hadamard and controlled rotation transforms.*

Proof There are two steps to achieve the unitary transform $\mathcal{P}$ as shown in Fig. 3. Hadamard transforms are used in step 1 and then controlled-rotation transforms are used in step 2 to change from $|0\rangle^{\otimes 2n+1}$ to $|H\rangle$ and then from $|H\rangle$ to $|I(\theta)\rangle$.

Let us consider the 2-D identity matrix I and the 2-D Hadamard matrix,

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \; H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}.$$

The tensor product of $2n$ Hadamard matrices is denoted by $H^{\otimes 2n}$. Applying the transform $\mathcal{H} = I \otimes H^{\otimes 2n}$ on $|0\rangle^{\otimes 2n+1}$ produces the state $|H\rangle$,

$$\mathcal{H}\left(|0\rangle^{\otimes 2n+1}\right) = \frac{1}{2^n}|0\rangle \otimes \sum_{i=0}^{2^{2n}-1} |i\rangle = |H\rangle. \tag{4}$$

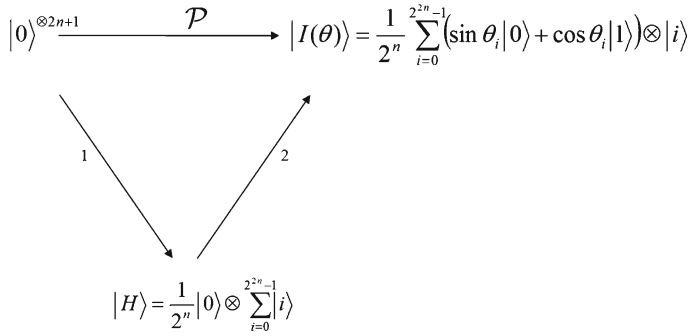


Fig. 3 Two steps to achieve $\mathcal{P}$

Let us also consider the rotation matrices (the rotations about $\hat{y}$ axis by the angle $2\theta_i$), $R_y(2\theta_i)$, and controlled rotation matrices, R_i, with $i = 0, 1, \ldots, 2^{2n} - 1$,

$$R_y(2\theta_i) = \begin{pmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{pmatrix}, \tag{5}$$

$$R_i = \left(I \otimes \sum_{j=0, j\neq i}^{2^{2n}-1} |j\rangle\langle j| \right) + R_y(2\theta_i) \otimes |i\rangle\langle i|. \tag{6}$$

The controlled rotation R_i is a unitary matrix since $R_i R_i^\dagger = I^{\otimes 2n+1}$. Applying R_k and $R_l R_k$ on $|H\rangle$ gives us

$$\begin{aligned} R_k(|H\rangle) &= R_k\left(\frac{1}{2^n}|0\rangle \otimes \sum_{i=0}^{2^{2n}-1} |1\rangle \right) \\ &= \frac{1}{2^n}\left[I|0\rangle \otimes \left(\sum_{i=0, i\neq k}^{2^{2n}-1} |i\rangle \right) \left(\sum_{i=0}^{2^{2n}-1} |i\rangle \right) \right. \\ &\qquad \left. + R_y(\theta_k)|0\rangle \otimes |k\rangle\langle k| \left(\sum_{i=0}^{2^{2n}-1} |i\rangle \right) \right] \\ &= \frac{1}{2^n}\left[|0\rangle \otimes \left(\sum_{i=0, i\neq k}^{2^{2n}-1} |i\rangle\langle i| \right) \right. \\ &\qquad \left. + (\cos\theta_k|0\rangle + \sin\theta_k|1\rangle) \otimes |k\rangle \right], \end{aligned} \tag{7}$$

$$
\begin{aligned}
R_l R_k |H\rangle &= R_l(R_k|H\rangle) \\
&= \frac{1}{2^n}\left[|0\rangle \otimes \left(\sum_{i=0, i\neq k,l}^{2^{2n}-1} |i\rangle\langle i|\right) + (\cos\theta_k|0\rangle + \sin\theta_k|1\rangle) \otimes |k\rangle \right. \\
&\quad \left. + (\cos\theta_l|0\rangle + \sin\theta_l|1\rangle) \otimes |l\rangle\right]. \qquad (8)
\end{aligned}
$$

From (8), it is clear that

$$
\mathcal{R}|H\rangle = \left(\prod_{i=0}^{2^{2n}-1} R_i\right)|H\rangle = |I(\theta)\rangle. \qquad (9)
$$

Therefore, the unitary transform $\mathcal{P} = \mathcal{R}\mathcal{H}$ is the transform turning quantum computers from the initialized state, $|0\rangle^{\otimes 2n+1}$, to the FRQI state, $|I(\theta)\rangle$. □

In the quantum circuit model, a complex transform is broken down into simple gates, i.e., single qubit and controlled two qubit gates, such as NOT, Hadamard, and CNOT gates which are shown in Figs. 4 and 5.

Corollary 1 *The unitary transform $\mathcal{P}$ described in the Lemma 1, for a given vector $\theta = (\theta_0, \theta_1, \ldots, \theta_{2^{2n}-1})$, $(n \in \mathbb{N})$ of angles, can be implemented by Hadamard, CNOT and $C^{2n}\left(R_y\left(\frac{2\theta_i}{2^{2n}-1}\right)\right)$ gates, where $R_y\left(\frac{2\theta_i}{2^{2n}-1}\right)$ are the rotations about $\hat{y}$ axis by the angle $\frac{2\theta_i}{2^{2n}-1}$, $i = 0, 1, \ldots, 2^{2n}-1$.*

Proof From the proof of Lemma 1, the transform $\mathcal{P}$ is composed of $\mathcal{R}\mathcal{H}$. The transforms $\mathcal{H}$ and $\mathcal{R}$ can be directly implemented by $2n$ Hadamard gates and 2^{2n} controlled rotations R_i or generalized-$C^{2n}\left(R_y(2\theta_i)\right)$ operations, respectively. In addition, the controlled rotations R_i can be implemented by $C^{2n}\left(R_y(2\theta_i)\right)$ and NOT operations

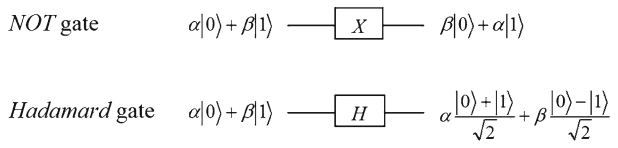


Fig. 4 NOT gate and Hadamard gate

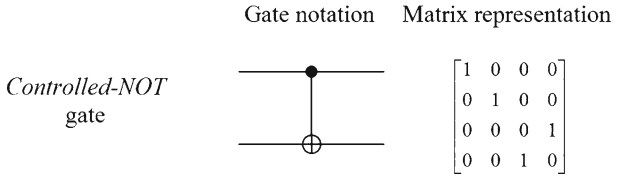


Fig. 5 CNOT gate

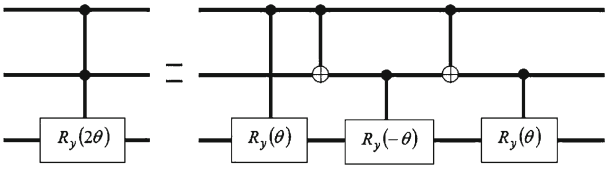


Fig. 6 $C^2(R_y(2\theta))$ gates can be built from $C(R_y(\theta))$, $C(R_y(-\theta))$ and $CNOT$ gates

[14]. Furthermore, the result in [1] implies that $C^{2n}\left(R_y(2\theta_i)\right)$ operations can be broken down into $2^{2n}-1$ simple operations, $R_y\left(\frac{2\theta_i}{2^{2n}-1}\right)$, $R_y\left(-\frac{2\theta_i}{2^{2n}-1}\right)$, and $2^{2n}-2$ CNOT operations. The example in the case of $n=1$ is shown in Fig. 6.

The total number of simple operations used to prepare FRQI state is

$$2n + 2^{2n} \times \left(2^{2n-1} - 1 + 2^{2n-1} - 2\right) = 2^{4n} - 3.2^{2n} + 2n. \tag{10}$$

This number is quadratic to the total 2^{2n} angle values, $\theta_i, i = 0, 1, \ldots, 2^{2n}-1$. This indicates the effiency of the preparation process. □

Theorem 1 (Polynomial preparation theorem) *Given a vector* $\theta = (\theta_0, \theta_1, \ldots, \theta_{2^{2n}-1})$, $(n \in \mathbb{N})$ *of angles, there is a unitary transform* $\mathcal{P}$ *that turns quantum computers from the initialized state,* $|0\rangle^{\otimes 2n+1}$ *to the FRQI state,*

$$|I(\theta)\rangle = \frac{1}{2^n}\sum_{i=0}^{2^{2n}-1}(\cos\theta_i|0\rangle + \sin\theta_i|1\rangle) \otimes |i\rangle,$$

composed of polynomial number of simple gates.

Proof Coming from Lemma 1 and Corollary 1. □

3 Quantum image compression based on minimization of Boolean expressions

Classical image compression techniques reduce the amount of computational resources, used to restore or reconstruct images. Similarly, quantum image compression is the procedure that reduces the quantum resources used to prepare or reconstruct quantum images. The main resource in quantum computation is the number of simple quantum gates or simple operations used in the computation. Therefore, the process, which decreases the number of simple quantum gates in the preparation and reconstruction of quantum images, is called Quantum Image Compression (QIC). The preparation and reconstruction of quantum images are the same, however, in the sense of their computation.

There are several reasons why image compression must be considered in FRQI. To start with, studies in classical image processing point out that there is redundancy in the image that can be reduced for compression in quantum image as well. Secondly, as shown in Sect. 2, preparing a quantum image needs a large number of simple gates.

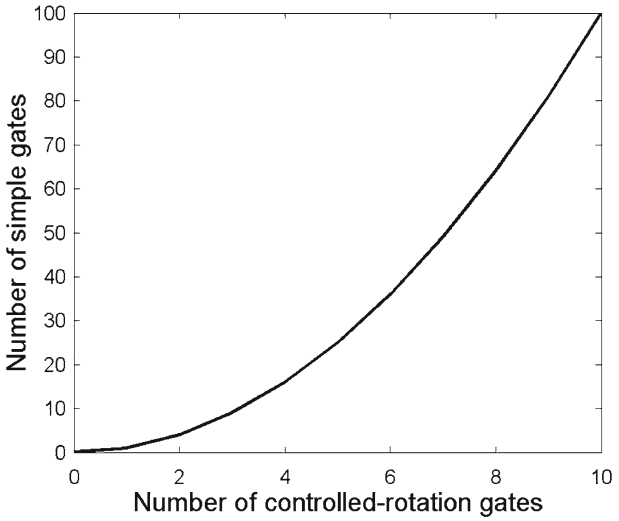


Fig. 7 The relation between number of rotation gates and total number of simple gates

For example a 2^{16} position image needs 2^{32} simple gates for preparation. Thirdly, in physical experiments, the number of simple gates should be decreased for robust implementation. For all of these reasons, the reduction of simple gates is necessary for both theoretical and practical aspects of the FRQI.

The amount of simple quantum gates used for preparing the FRQI depends mainly on the number of controlled rotation gates, as shown in Fig. 7. Essentially, most of the basic gates required for the preparation process are to simulate the $C^{2n}(\cdot)$ gates. Therefore, the reduction of controlled rotation gates results in a decrease in the total number of gates. One of the ways to reduce the number of gates is to integrate some of them under certain conditions. This section describes a method to integrate controlled rotation gates having the same rotation angle.

As observed in classical image processing, many colors are indistinguishable to the human eyes. Exploiting this fact of human vision, the classical image representations use a limited number of levels for expressing gray scales or colors in digital images with various sizes without significant impact on the quality of the images. Regarding this observation, the input angles encoding colors can take their values from a discrete set of numbers. Consequently, in the preparation process, controlled rotation operators with the same angle but different conditions affect the positions having the same colors. Therefore, all rotations can be divided into groups such that each group includes only operators having the same rotation angle.

The difference between controlled rotation gates in one of the divided groups is only the conditional part on each gate. As presented in Sect. 2, the conditional part of a controlled rotation gate depends on the binary string which encodes the corresponding position in a image. Therefore, the rotation angle and binary strings encoding conditional parts of rotation gates in a group characterize the group. From this point of view, each group has a generalized controlled rotation gate in which the rotation angle is

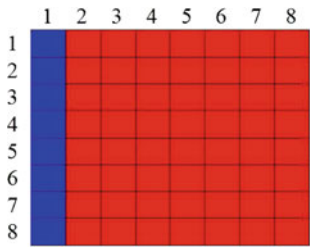


Fig. 8 8×8 *two-color* image

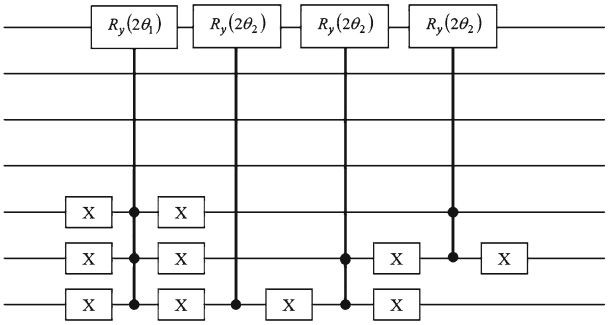


Fig. 9 The minimized circuit for 8×8 *two-color* image

the group's rotation angle and the controlled condition is the integration of all binary strings in the group.

In order to make the above arguments explicit, let us consider a 8×8 image shown in Fig. 8 as an example. This image contains only two colors, blue and red with 8 and 56 positions respectively, which requires 64 $C^6(\cdot)$ gates in its general preparation discussed in Sect. 2. Dividing all the 64 controlled rotation gates into 2 groups helps to reduce the number of gates from 64 to 4 as shown in Fig. 9 resulting in a reduction of the number of controlled-rotation gates by 93.75%. In addition, the controlled-rotation gates in the minimized circuit are much simpler than $C^6(\cdot)$ gates, implying that the number of basic gates used in each controlled-rotation gate is reduced as well. Consequently, the red-group uses one $C^1(\cdot)$ and two $C^2(\cdot)$ gates with the controlled conditions satisfy only the red positions.

There is a way to transform a binary string to a Boolean minterm by considering each position in the binary string as a Boolean variable. If x is the Boolean variable at a position in the string and the value of that position is 1 then the lateral x is used in the minterm otherwise the lateral $\bar{x}$ is used. For example, the binary string 000 and 101 are equivalent to $\bar{x}_2\bar{x}_1\bar{x}_0$ and $x_2\bar{x}_1x_0$, respectively. With this method, there is a one-to-one correspondence between the set of all binary strings with length n and the set of all Boolean minterms generating from n Boolean variables.

After dividing the controlled rotation gates of an image into same color groups, the next step is the compression or minimization of the gates in each group. For this purpose, the binary strings of each group play a key role. Using the method described in the above paragraph, each of these binary strings corresponds to a Boolean term. Therefore, the integration of all of binary strings in a group is equivalent to the conjunction all

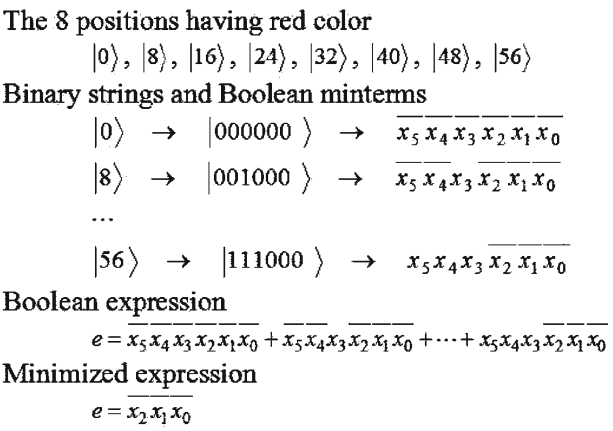


Fig. 10 8-position group, the corresponding Boolean expression and its minimized expression

of their corresponding Boolean minterms. This conjunction forms a Boolean expression. For example, let us consider an 8-position group in Fig. 10 which comes from the blue-group in Fig. 8. The Boolean expression captures all information about the binary strings in the group. This means that the expression contains all information about the conditions for controlling the gates in that group. The expression can be rewritten in minimized form which contains only one term, as shown in Fig. 10. This observation suggests that only one controlled-rotation gate can be used instead of 8 gates.

The quantum image compression (QIC) algorithm is proposed to reduce controlled rotation gates in the same color groups based on the minimization of their Boolean expression as shown in Fig. 11. The procedure starts with the information about positions in a same color group and ends with the minimized form of the Boolean expression.

The minimized Boolean expressions are used to construct a quantum circuit with a lesser number of simple gates than the original circuit. The number of product terms in a minimized Boolean expression indicates the number of conditioned rotation gates that can be used for the corresponding group of same color positions. The analysis of the compression ratio in Sect. 5.2 is based on this observation. The laterals in a product term in each minimized expression point out the condition part of the conditioned rotation gates. The Boolean variables with complement laterals use extra pairs of NOT gates for each complement.

In the QIC algorithm, the minimization of Boolean expressions plays a fundamental role because the number of Boolean variables in the expression is not trivial. The problem of minimizing Boolean expressions has been studied extensively, starting with Karnaugh maps, Espresso algorithm, etc [3]. The Espresso algorithm is widely used in the implementation of practical design software programs like Friday Logic, Minilog, etc. Using the Espresso algorithm, programs can minimize the Boolean expressions on 100 inputs and 100 outputs within reasonable running time. There are heuristic synthesis methods [12,13] that reduce the number of simple gates in quantum circuits in

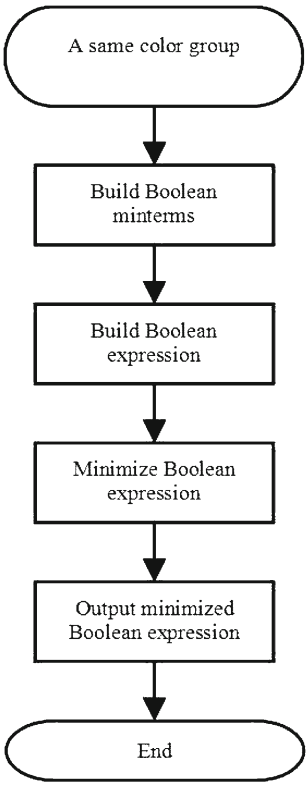


Fig. 11 Flow chart of the quantum image compression algorithm

general. Therefore, applying these methods after the QIC algorithm could give better results.

4 Image processing operators on quantum images based on unitary transforms

Representations of images provide the background for image processing algorithms. The algorithms use an image as input to produce another image as output by performing simple operations. Furthermore the output image is analyzed to obtain useful information. This procedure in classical computers can be applied to quantum computers by using unitary transforms as image processing operations.

In classical image processing, basic operators provide fundamental manipulations in various algorithms for processing images. These operators include changes of colors at some positions, shifting the color of whole image, performing Fourier transform, etc. These basic operations are important in constructing and understanding the processing algorithms. In quantum images, however, the primary manipulations are not obvious since they should be invertible. Meanwhile, some classical operations are not invertible such as convolution operators [11] that means they are physically impossible in quantum computation. With FRQI, the basic operations can be classified and constructed by using unitary transforms.

With the FRQI proposal, images are expressed in their FRQI states and quantum image processing operations are performed using unitary transforms on those states. These transforms are divided into 3 categories; G_1, G_2 and G_3, applied to FRQI states dealing with only colors, colors at some specific positions and the combination of colors and positions, respectively. The first two type of operators are simple in the sense that the appearance of the output and input images is highly related. The last type is more complex because it involves the combination of both color and position in the output images which make the interpretation of measurements on the output images difficult.

Operators in the first group use only information about the color, such as color shifting and the second group contains those based on colors at some position in the images, for instance the changes in color at specific positions. The last group targets information about both color and position as in Fourier transform. Each category has its own type of unitary transform. The unitary transforms are in the following forms

$$G_1 = U_1 \otimes I^{\otimes n}, \tag{11}$$
$$G_2 = U_2 \otimes C + I \otimes \bar{C}, \tag{12}$$
$$G_3 = I \otimes U_3, \tag{13}$$

where U_1, U_2 are single qubit transforms, U_3 is n-qubit transforms, C and $\bar{C}$ are matrices regarding eligible and ineligible positions, $I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ is the identity operator and n is the number of qubits encoding positions. From the point of quantum circuit modeling, G_1 uses a single qubit gate U_1, G_2 uses an additional control from the position on the gate U_2 and G_3 just use the n-qubit transform U_3. These circuits are shown in Figs. 12, 13 and 14, respectively.

The color shifting operator, S, is defined as an operator in the group G_1, $S = U \otimes I^{\otimes n}$, by using rotation matrices $U = R_y(2\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$, where θ is the shifting parameter. The following calculation produces the result $|I_S\rangle$ of the application of S on $|I\rangle$,

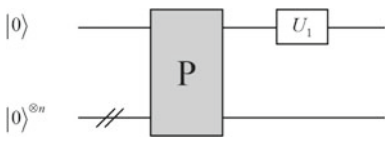


Fig. 12 Quantum circuit of G_1 operations dealing with only the *color* part by single qubit gates U_1

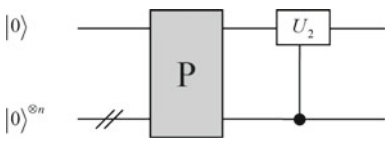


Fig. 13 Quantum circuit of G_2 operations dealing with the *colors* at some positions by single qubit gates U_2

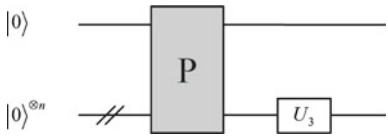


Fig. 14 Quantum circuit of G_3 operations combining both *color* and position information by n-qubit gates U_3

$$
\begin{aligned}
|I_S\rangle &= S(|I\rangle) \\
&= (U \otimes I^{\otimes n}) \left(\frac{1}{2^n} \sum_{i=0}^{2^{2n}-1} (\cos\theta_i |0\rangle + \sin\theta_i |1\rangle) \otimes |i\rangle \right) \\
&= \frac{1}{2^n} \sum_{i=0}^{2^{2n}-1} (\cos(\theta_i + \theta)|0\rangle + \sin(\theta_i + \theta)|1\rangle) \otimes |i\rangle. \qquad (14)
\end{aligned}
$$

The quantum image $|I_S\rangle$ has all of its colors coming from the original image $|I\rangle$ by shifting the θ angle.

The change in color at some points in an image depends on the specific positions in the image. Information about the positions is used as conditions encoded in the matrix C of the controlled-gate G_2 to construct the processing operators. For instance, let us consider a 2×2 image and the change in color at positions $|0\rangle, |2\rangle$. The matrix $C = |0\rangle\langle 0| + |2\rangle\langle 2|$ and $\bar{C} = |1\rangle\langle 1| + |3\rangle\langle 3|$ are used to construct the controlled-gate G_2,

$$
G_2 = U \otimes (|0\rangle\langle 0| + |2\rangle\langle 2|) + I \otimes (|1\rangle\langle 1| + |3\rangle\langle 3|). \qquad (15)
$$

The action of this particular G_2 on a general 2×2 image in FRQI form, $|I\rangle = \frac{1}{2}\sum_{i=0}^{3}(\cos\theta_i|0\rangle + \sin\theta_i|1\rangle) \otimes |i\rangle$, is given by

$$
\begin{aligned}
G_2|I\rangle = \frac{1}{2} [&(\cos\theta_0 \times U|0\rangle + \sin\theta_0 \times U|1\rangle) \otimes |0\rangle \\
&+ (\cos\theta_2 \times U|0\rangle + \sin\theta_2 \times U|1\rangle) \otimes |2\rangle \\
&+ (\cos\theta_1|0\rangle + \sin\theta_1|1\rangle) \otimes |1\rangle \\
&+ (\cos\theta_3|0\rangle + \sin\theta_3|1\rangle) \otimes |3\rangle] . \qquad (16)
\end{aligned}
$$

The calculation in (16) shows that the action of operator U for changing color has affects only on the specific positions $|0\rangle, |2\rangle$.

One of the examples on the G_3 operators, which combine both colors and positions in output images, is the operator based on quantum Fourier transform. The application of QFT on FRQI can be considered as the application of Fourier transform on the cosine part and sin part of the FRQI state coefficients as in the following calculations of c_k and s_k.

$$|T\rangle = \frac{1}{2^n}\sum_{i=0}^{2^{2n}-1}(\cos\theta_i|0\rangle + \sin\theta_i|1\rangle) \otimes QFT(|i\rangle)$$
$$= \frac{1}{2^n}\left[\sum_{k=0}^{2^{2n}-1} c_k|0k\rangle + \sum_{k=0}^{2^{2n}-1} s_k|1k\rangle\right], \quad (17)$$

where

$$c_k = \frac{1}{2^n}\sum_{k=0}^{2^{2n}-1} e^{2\pi jik/2^{2n}}\cos\theta_i, \quad (18)$$

$$s_k = \frac{1}{2^n}\sum_{k=0}^{2^{2n}-1} e^{2\pi jik/2^{2n}}\sin\theta_i, \quad (19)$$

$$k = 0, 1, \ldots, 2^{2n} - 1.$$

The complexity of each type of operation is specified based on the number of simple gates in the corresponding quantum circuit. The number of simple gates used for an operation in the G_1 category is one gate as in Fig. 12 that means the complexity of the G_1 operation is $O(1)$. The number of controlled rotations used for an operator in the G_2 category depends linearly on the number of positions involving the operator, $O(N)$, where N is the number of positions in the whole image. The complexity of G_3 operations depends on the complexity of the n-qubit operations U_3 as shown in Fig. 14. If we use the U_3 operations with $O(log^2(N))$ like quantum Fourier transform, quantum Wavelet transform, etc. then the complexity of the G_3 operation is $O(log^2(N))$.

5 Experiments on quantum images

A classical computer was used to simulate the experiments on quantum images. The simulations are based on linear algebra with complex vectors as quantum states and unitary matrices as unitary transforms using Matlab. The final step in any quantum

Fig. 15 The image used in experiments of storage and retrieval quantum images

computation is that of measurement which converts the quantum information into the classical form as probability distributions. Extracting and analyzing the distributions gives information for retrieving quantum images and revealing structures in these images. In Sect. 3, the QIC algorithm reduces the number of conditioned rotation gates in quantum image storage process. The experiment on the analysis of how many gates are reduced (or compression ratio) is done using the Lena image. The minimization part in the QIC is done by Logic Friday software which is widely used in practice. The application of QFT to combine information on colors and positions on FRQI as presented in Sect. 4 is used to detect lines in binary images. Experiments are performed on desktop computer with Intel Core 2 Duo 1.86 GHz CPU and 2 GB RAM.

5.1 Storage and retrieval of quantum images

The essential requirements for representing a classical or quantum image are the simplicity and efficiency in the storage and retrieval of the image. The storage of a quantum image is achieved by the preparation process which is ensured by the proposed PPT in Sect. 2. The measurement of the quantum image state produces a probability distribution that is used for the retrieval of the image. Input information for preparation in this experiment is the gray levels coming from the 64×64 gray image in Fig. 15.

From the image, the angles encoding the gray levels and corresponding positions are extracted. The controlled rotation gates used in the quantum circuit are built based on this data. The quantum image state has 8,192 complex numbers as its coefficients, Fig. 16 shows the 64 coefficients of the 8×8 lower-right block from the input image.

A measurement of a quantum state based on the set of basis vectors produces only one result which is one of the basis vectors. In quantum computation, measurements are performed on identical states instead of one state. With only one quantum state, it is impossible to get information from that state. Therefore, a measurement process needs many identical quantum states. For instance, in order to retrieve information about the quantum state

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{20}$$

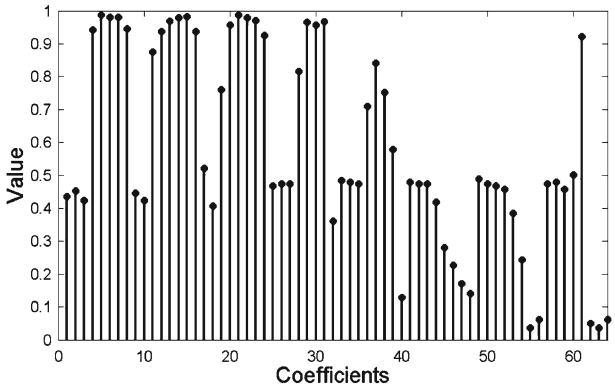


Fig. 16 64 coefficients of the 8×8 *lower-right* block from the input image

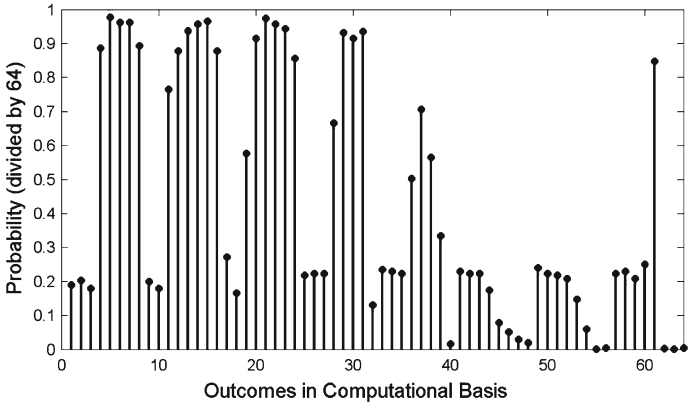


Fig. 17 Probability distribution of the 8×8 block

many identical states, $|\psi\rangle$, are prepared. Each measurement on $|\psi\rangle$ gives either 0 or 1 as result. Many measurements, however, reveal either the result 0, with probability $|\alpha|^2$, or the result 1, with probability $|\beta|^2$. This implies that a measurement process on a quantum state gives information about the quantum state in form of a probability distribution.

With general quantum states, the probability distributions are not enough to understand clearly the states because their coefficients are complex numbers. The FRQI, however, contains only real valued coefficients that make retrieval of all information about the state possible. The Fig. 17 shows the probability distribution of the 8×8 block mentioned in Fig. 16. In addition, the quantum circuits indicated by the PPT provide a way to prepare many identical FRQI states used in the measurement process.

5.2 Analysis of quantum image compression ratios

As presented in Sect. 3, the QIC method provides a way to reduce resources used in preparation and reconstruction of quantum images. In our experiments, compression ratios among groups are estimated based on the analysis of minimizations of Boolean expressions derived from the 8×4 single digit images (0–9) and the 256×256 gray image of Lena as in the Figs. 18 and 19 respectively. The minimization step in the QIC algorithm is done by Logic Friday free software. The compression ratio is as in Eq. 21.

$$\frac{Rotations - reducedRotations}{Rotations} \times 100\%, \tag{21}$$

where *Rotations* is the number of rotations indicated by the PPT and *reducedRotations* is the number of rotations indicated by the QIC algorithm.

The single digit images are binary 8×4 images of digits from 0 to 9. The quantum system for the images in the FRQI form contains 6 qubits, 5 qubits for encoding

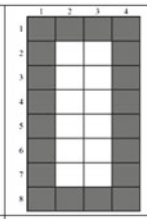
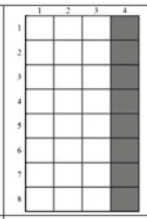
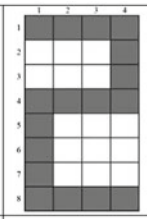
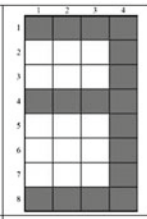
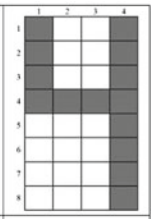
Image					
Reduced Rotations	10	3	10	9	9
Ratio (%)	68.75	90.63	68.75	71.86	71.86

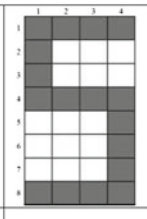
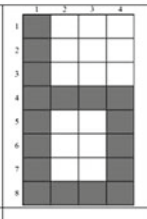
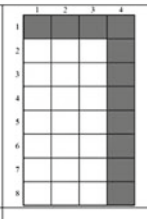
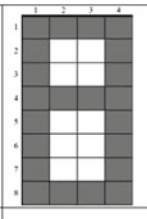
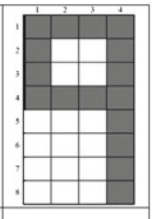
Image					
Reduced Rotations	10	9	8	10	10
Ratio (%)	68.75	71.86	75.00	68.75	68.75

Fig. 18 Single digit images and their corresponding compression ratios

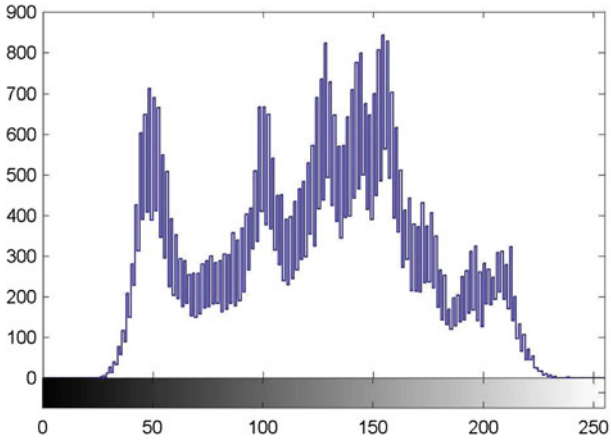


Fig. 19 Lena image and its histogram graph

position and 1 qubit for colors. As shown in PPT, the upper bound of the number of controlled rotations to prepare the FRQI states of the images is 32 rotations. There are only 2 groups of positions having the same color because the images are binary images. The single digit images, the number of reduced rotations when QIC algorithm is applied on each group of positions and the compression ratios for each single digit image are shown in Fig. 18.

With experiments on the 256×256 gray scale Lena image, the quantum circuit includes 17 qubits of which 16 are used to address positions in the image and the remaining qubit is used to storing colors. The quantum circuit indicated by the PPT contains 2^{16} conditioned rotation gates. The purpose of this experiment is to analyze the compression ratio and not to deal with the preparation of the quantum state. Therefore, the preparation involving very large number of conditioned rotation gates does not matter.

A gray image can be partitioned into groups of positions having the same gray level. For the Lena image in Fig. 19, there are 207 groups containing at least one position having the same color as shown in its histogram graph in Fig. 19. The number of positions in groups ranges from one to 822 positions. The total number of conditioned rotation gates for preparation is reduced by applying the QIC for each group as shown in the upper graph in Fig. 20. The compression ratios of groups are different since the numbers and of positions and relations between the positions in groups are different.

The compression ratios range from 6.67 to 31.62% between the groups as shown in Table 1; Fig. 20. The reasons for the variety of compression ratios between groups are

- Number of positions in the groups are different,
- The relation between positions in each group is different.

5.3 Simple detection of a line of quantum image based on quantum Fourier transform

Based on the discussion on QFT in Sect. 4, the simulation of QFT is the application of discrete Fourier transform on a classical computer. In this experiment, each 8×8 binary image contains a line as a simple structure. These lines can be defined as periodic functions. The FRQI for the binary images used in the experiment includes 7 qubits; 6 qubits for all the positions and 1 qubit for the color. The computational basis measurements on the transformed quantum states produce the probability distribution. The detection of lines in images from the probability distributions is done by using observations on the shapes of the distributions generated from cosine part of the FRQI states. The experiment studies four cases of lines in binary images shown in left side of Fig. 21.

Since all coefficients in the FRQI states are real numbers, there is a symmetric property among amplitudes of coefficients. The difference between the probability distributions in the first and fourth cases is the distance between the maximas within each distribution as shown in the right side of Fig. 21.

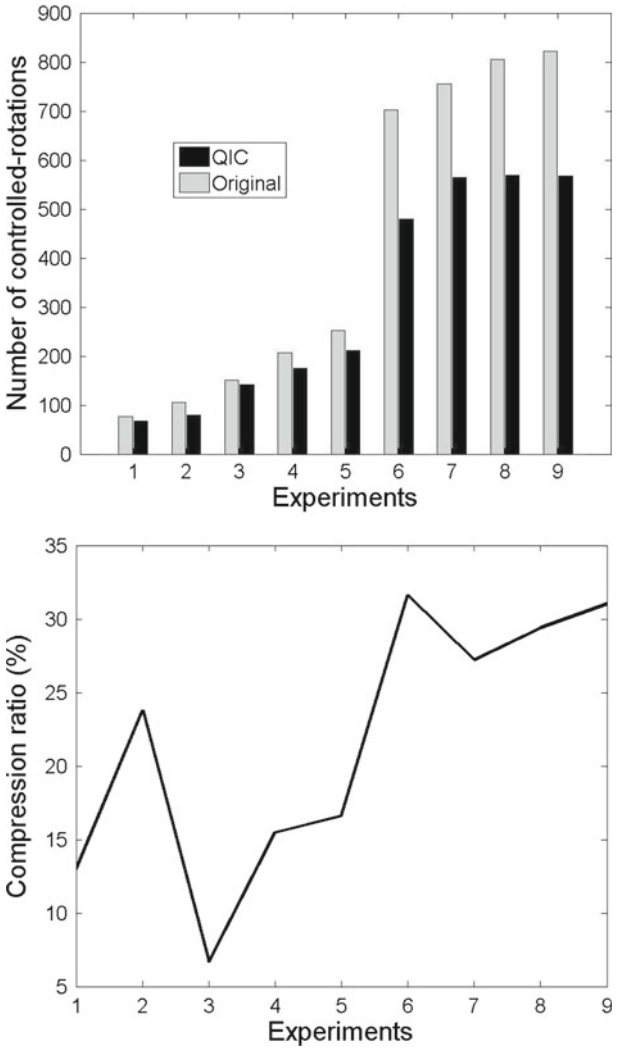


Fig. 20 The graphs of rotations and compression ratios for each group of positions having the same *gray level*

6 Conclusions

A flexible representation of quantum image (FRQI) is proposed in order to provide a basis for the polynomial preparation process and quantum image processing operations based on unitary operators. The FRQI captures image colors and their corresponding positions in a quantum state. The proposed PPT achieves a unitary preparation process using a polynomial number of simple operators transforming quantum computers from the initial state to FRQI state. It also points out the design of the quantum circuit using Hadamard gates and controlled rotation gates for the transform. Positions in an image

Table 1 Compression ratios of same color groups in the Lena image

Positions level	Gray level	Rotations	Reduced rotations	Compression ratio (%)
50	34	77	67	12.99
100	218	105	80	23.81
150	67	151	141	6.67
200	38	207	175	15.46
250	66	253	211	16.60
700	158	702	480	31.62
750	142	775	564	27.23
800	152	806	569	29.40

can be divided into groups of positions having the same color. Using the proposed QIC algorithm on the information of the groups, the number of simple gates used in FRQI preparation is reduced. The QIC is based on the minimization of Boolean equations which derives from the binary strings encoding positions in same color groups. Quantum image processing operators based on unitary transforms are addressed on FRQI. These operators are divided into 3 categories based on the 3 types of unitary transforms applied on FRQI dealing with only colors, colors at some specific positions and the combination of colors and positions. Experiments on FRQI including storing and retrieving quantum images, compression ratios of the QIC algorithm and an application of QFT as an image processing operation were done. Using the result of PPT and measurements of identical quantum states, the quantum image can be stored and retrieved. The compression ratios of the QIC algorithm on single digit binary images range from 68.75 to 90.63% and on groups having the same gray level in the Lena image range from 6.67 to 31.62%. The application of QFT in FRQI as in the detection of a line in a binary image was also discussed.

The above results imply that the FRQI can play a fundamental role in representing and processing images on quantum computers. The polynomial preparation and QIC express the efficiency of FRQI in both theory and practice. The division of three types of image processing operators on FRQI provides a guide to designing new unitary operators.

As for future work, the results in this paper will be extended to the following directions. Firstly, as the discussion in Sect. 4, the image processing operators are divided into 3 types. The investigation on each of three types of image processing operations explores new operators on quantum images. For example, the quantum Wavelet transform, the quantum discrete cosine transform, etc. are able to replace quantum Fourier transform in the type-3 operation. Secondly, the systematic analysis of the compression ratios of QIC algorithm, presented in Sect. 3, on a larger database of images will provide more insights into the efficiency of the algorithm. Thirdly, information-theoretic aspects on FRQI considering the existence of errors such as error-correcting codes are required for a robust representation for practical applications. The above mentioned directions are all on a single image. There are interesting questions on quantum operations having impacts on multiple images such as image matching, image searching

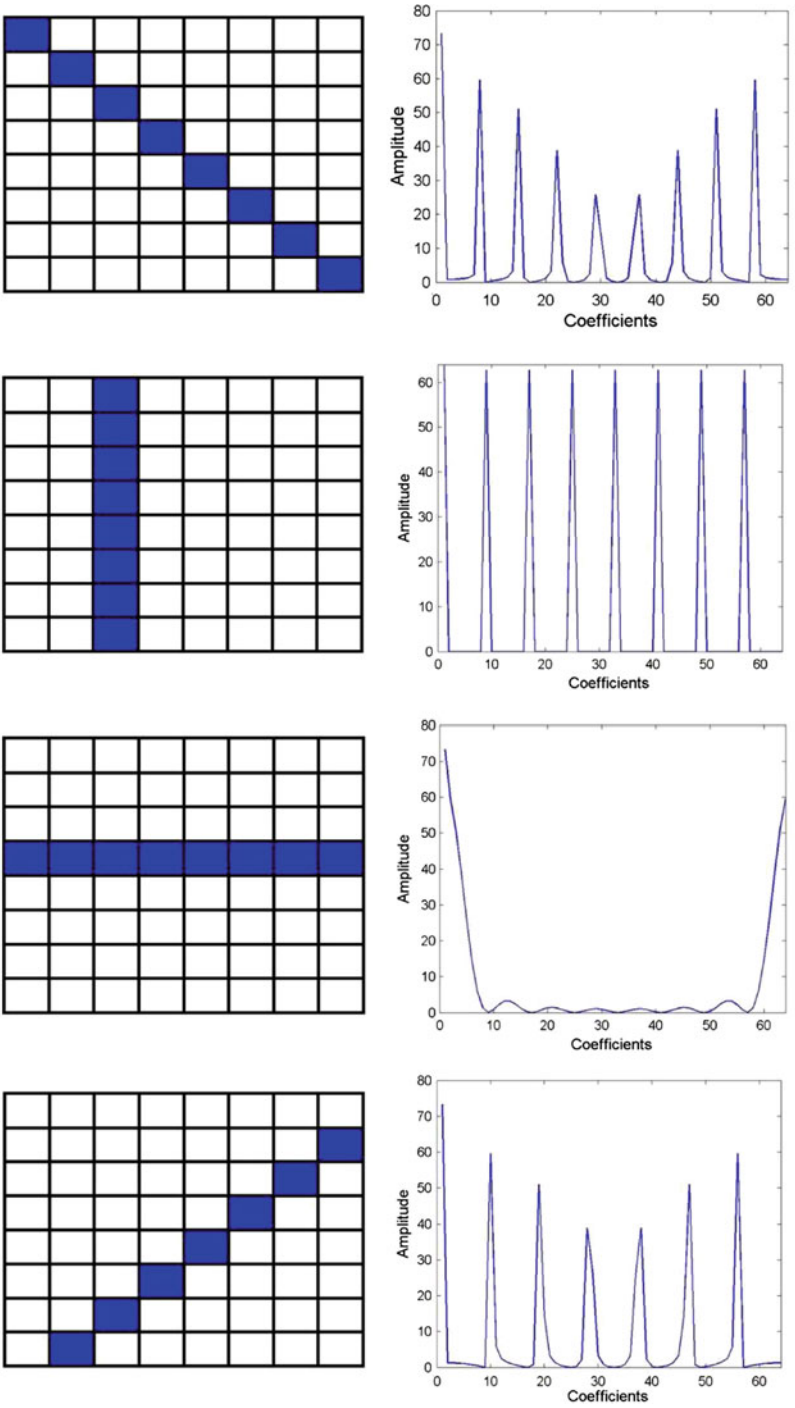


Fig. 21 8×8 binary images and their corresponding probability distributions

on a set of images in FRQI states. These directions will open new results on quantum image processing in general.

References

1. Barenco, A., Bennett, C.H., Cleve, R., DiVincenzo, D.P., Margolus, N., Shor, P., Sleator, T., Smolin, J.A., Weinfurter, H.: Elementary gates for quantum computation. Phys. Rev. A **52**, 3457 (1995)
2. Beach, G., Lomont, C., Cohen, C.: Quantum image processing (quip). Proc. Appl. Imagery Pattern Recognit. Workshop, 39–44 (2003)
3. Brayton, R.K., Sangiovanni-Vincentelli, A., McMullen, C., Hachtel, G.: Logic Minimization Algorithms for VLSI Synthesis. Kluwer Academic Publishers, Dordrecht (1984)
4. Caraiman, S., Manta, V.I.: New applications of quantum algorithms to computer graphics: the quantum random sample consensus algorithm. Proc. 6th ACM Conf. Comput. Frontier, Ischia, Italy. ACM, New York, 81–88 (2009)
5. Curtis, D., Meyer, D.A.: Towards quantum template matching. Proc. SPIE **5161**, 134–141 (2004)
6. Feynman, R.P.: Simulating physics with computers. Int. J. Theor. Phys. **21**(6/7), 467–488 (1982)
7. Fijany, A., Williams, C.P.: Quantum wavelet transform: fast algorithm and complete circuits. arXiv:quant-ph/9809004 (1998)
8. Grover, L.: A fast quantum mechanical algorithm for database search. Proc. 28th Ann. ACM Symp. Theory Comput. (STOC 1996), ACM, New York, 212–219 (1996)
9. Klappenecker, A., Rötteler, M.: Discrete cosine transforms on quantum computers. Proc. IEEER8-EURASIP Symp. on Image and Signal Processing and Analysis (ISPA01), Pula, Croatia, 464–468 (2001)
10. Latorre, J.I.: Image compression and entanglement. arXiv:quant-ph/0510031 (2005)
11. Lomont, C.: Quantum convolution and quantum correlation algorithms are physically impossible. arXiv:quant-ph/0309070 (2003)
12. Lomont, C.: Quantum circuit identities. arXiv:quant-ph/0307111 (2003)
13. Maslov, D., Dueck, G.W., Miller, D.M., Camille, N.: Quantum circuit simplification and level compaction. IEEE Trans. Comput.-Aided Design Integr. Circuits Syst. **27**(3), 436–444 (2008)
14. Nielsen, M., Chuang, I.: Quantum Computation and Quantum Information. Cambridge University Press, New York (2000)
15. Shor, P.W.: Algorithms for quantum computation: discrete logarithms and factoring. Proc. 35th Ann. Symp. Found. Comput. Sci. IEEE Computer Soc. Press, Los Almitos, CA. 124–134 (1994)
16. Tseng, C.C., Hwang, T.M.: Quantum circuit design of 8×8 discrete cosine transforms using its fast computation flow graph. ISCAS 2005. vol. I. 828–831 (2005)
17. Venegas-Andraca, S.E., Ball, J.L.: Storing Images in engtangled quantum systems. arXiv:quant-ph/0402085 (2003)
18. Venegas-Andraca, S.E., Bose, S.: Storing, processing and retrieving an image using quantum mechanics. Proc. SPIE Conf. Quantum Inf. Comput. vol. 5105, 137–147 (2003). doi:10.1117/12.485960