IEEE Transactions on Quantum Engineering

Received July 26, 2021; accepted December 1, 2021; date of publication December 13, 2021; date of current version January 18, 2022.

Digital Object Identifier 10.1109/TQE.2021.3134648

Quantum Radon Transforms and Their Applications

GUANGSHENG MA[1], HONGBO LI[1,2], AND JIMAN ZHAO[3]

[1] Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China
[2] University of Chinese Academy of Sciences, Beijing 100049, China
[3] School of Mathematical Sciences, Beijing Normal University, Beijing 100875, China

Corresponding author: Guangsheng Ma (e-mail: gs_ma@qq.com)

This work was supported in part by the China National Key Research and Development Projects 2020YFA0712300 and 2018YFA0704705, in part by the National Natural Science Foundation of China under Grant 11671388, Grant 11471040, and Grant 11761131002, and in part by the Chinese Postdoctoral Science Foundation under Grant 2020M680716.

ABSTRACT This article extends the Radon transform, a classical image-processing tool for fast tomography and denoising, to the quantum computing platform. A new kind of periodic discrete Radon transform (PDRT), called the quantum periodic discrete Radon transform (QPRT), is proposed. The quantum implementation of QPRT based on the amplitude encoding method is exponentially faster than the classical PDRT. We design an efficient quantum image denoising algorithm using QPRT. The simulation results show that QPRT preserves good denoising capability as in the classical PDRT. Also, a quantum algorithm for IDRT is proposed, which can be used for fast line detection. Both the quantum extension of IDRT and the line detection algorithm can provide polynomial speedups over the classical counterparts in certain cases.

INDEX TERMS Quantum computing, radon transform.

I. INTRODUCTION

Radon transform, proposed by Radon in 1917 [1], is an important image processing tool with widespread applications in computed tomography, geophysics, remote sensing, etc. [2]. It changes a function f defined on the plane to a function $R_\theta f(\rho)$ defined on the space of lines in the plane, whose value on the line with intercept ρ and slope θ equals the integral of function f along the line

$$R_\theta f(\rho) := \iint_{\mathbb{R}^2} f(x, y)\delta(\rho - x\sin\theta + y\cos\theta)dxdy \quad (1)$$

where $\theta \in (0, \pi]$, $\rho \in \mathbb{R}$, and δ is the Dirac function. By definition, Radon transform possesses the capability of detecting singularities along straight lines and performs better at denoising images with linear singularities than other image processing tools [2], [3].

To implement Radon transform, discretization is necessary. However, the different discretization methods will result in different discrete Radon transforms (DRTs) that have different applications. In discretization, there are two methods to approximate the line integral in (1): the interpolation method and the periodic discrete grid method.

The interpolation-based discrete Radon transform (IDRT) method evaluates the integral along a straight line by making interpolation among the adjacent points on the discrete image grid of the line. The earlier DRTs are based on this method [1]. In 1987, Beylkin [4] discovered an exact inversion formula and proved that if the discrete version is based on Radon's original formula, then the reconstruction can only be approximate. The IDRT can be used in line detection, X-ray-computed tomography [5], etc. Performing the IDRT on an $N \times N$ image often requires at least $\Omega(N^3)$ arithmetic operations [6].

The periodic discrete Radon transform (PDRT) method calculates the integrals along a set of warped lines, and does not have direct connection with the continuous Radon transform [6]–[8]. Still the method possesses some very nice properties, such as the exact reconstruction property, Fourier slice property, etc. Matus and Flusser [9] first investigated PDRT on $\mathbb{Z}_p^2$, where p is the prime. Then, Hsung *et al.* [10] extended PDRT to $\mathbb{Z}_{p^n}^2$. The PDRT has been used in image denoising [3], tomographic reconstruction [11], image watermarking and encryption [12], etc. To compute the PDRT of an $N \times N$ image, $\Omega(N^2 \log N)$ arithmetic operations are required [13].

With the emergence of quantum computing, for many important computational problems, it is found that quantum algorithms can provide dramatic speedup [14], for example, exponentially fast quantum algorithms, such as quantum

Fourier algorithm, Shor's factoring algorithm [15], polynomially fast quantum algorithms, such as Grover's search algorithm [16], and so on [17].

To speed up image processing in the forthcoming quantum computing age, quantum image processing (QIMP), whose topics range from quantum image representations to image processing, has drawn a lot of attention in the last decade [18]–[20]. Early work on QIMP concentrated on the quantum representation of images, e.g., [21]. Roughly speaking, there are two typical representation methods: (1) amplitude representation method, such as the Real Ket representation [22], which utilizes the qubits' amplitude and a computational basis state to encode the grayscale and the location of a pixel, respectively; (2) basis representation method, such as the novel enhanced quantum representation (NEQR) [23], which utilizes a register's computational basis state to encode both the grayscale and the location of pixels.

Another focus in QIMP is to develop image processing tools in the quantum computation framework [24], [25]; for example, Yao [26] proposed an efficient quantum image edge detection algorithm for Real Ket images. Image segmentation, watermarking, scrambling, and some other image-processing problems have also been investigated in QIMP [27]–[29].

However, there is still no extension of Radon transform to the field of QIMP. In this article, we make such extension.

The main difficulty in making the extension comes from the fact that there is no preference of using unitary transforms in designing classical algorithms; conversely, quantum algorithms mostly use unitary operators. Particularly, to realize the classical "nonunitary" PDRT with unitary transformations, we design a unitary quantum reversible multiplication, and then utilize Fourier slice property of PDRT while replacing the conventional multiplication with the new quantum reversible multiplication. Finally, we obtain a transform that is similar but distinct from any existing PDRT, which we call the quantum periodic discrete Radon transform (QPRT).

A. CONTRIBUTIONS

We propose two fast quantum algorithms for DRTs. The first one is an "exponentially fast" quantum algorithm for performing QPRT on Real Ket images, which has the following properties.

1) The QPRT on an $N \times N$ Real Ket image can be implemented in time $O(\log^3 N)$, which runs exponentially faster than the classical PDRT on classical images, which has runtime $\Omega(N^2 \log N)$.
2) The QPRT preserves many good properties as in the classical PDRT, such as Fourier slice property (cf., Proposition 6) and good denoising capability (cf., Figs. 6 and 7).

The second one is a quantum algorithm for IDRT on Real Ket images. Due to the particular techniques used for achieving interpolation, the transformation result is naturally stored in the computational basis state (i.e., encoded in NEQR form).

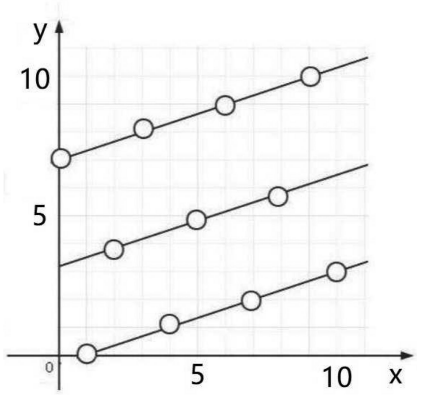

FIGURE 1. Example of a discrete line $L^{11}_{1,3} = \{(x, y) | x - 3y = 1 \bmod 11\}$ on lattice $\mathbb{Z}^2_{11}$. The hollow circles indicate the lattice points over the discrete line $L^{11}_{1,3}$.

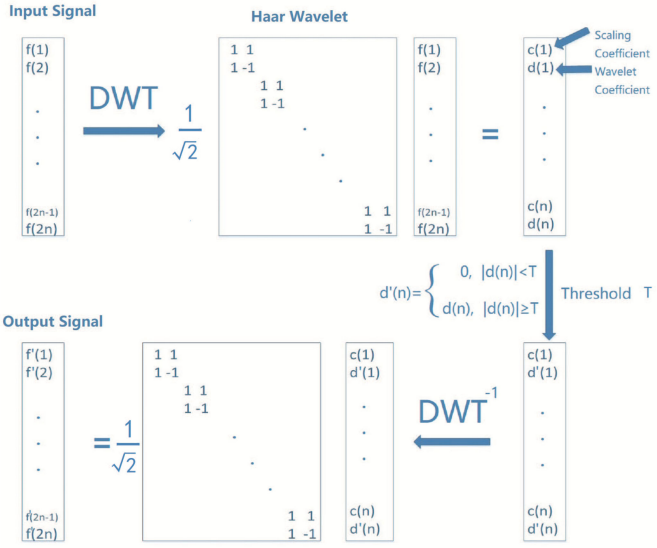

FIGURE 2. Denoising using DWT. There are three steps in total: 1) Apply the DWT to the input noisy signal f, i.e., multiply f by the matrix form of some wavelet (the one used here is Haar wavelet). The resulting coefficients can be divided into two parts: the scaling coefficients c, composed of the odd rows, and the wavelet coefficients d, composed of even rows. 2) Apply hard-thresholding to wavelet coefficients d with threshold T to obtain new wavelet coefficients d'. 3) Apply inverse DWT to thresholded coefficients, where in this case the inverse of the Haar wavelet matrix is itself.

Also, this algorithm requires multiple calls for image preparation oracles, and has a time dependence on the desired precision ϵ. Specifically,

3) Performing IDRT on an $N \times N$ Real Ket image has a time complexity $O(\frac{T_{\text{in}}}{\epsilon} \text{polylog} N)$, where T_{in} is the image preparation time, ϵ is the precision.

Our quantum algorithm for IDRT can achieve polynomial speedups in certain cases. Also, we present a quantum line detection algorithm using IDRT that provides a quadratic speedup for the detection process (cf., Section VI-B). Fig. 9

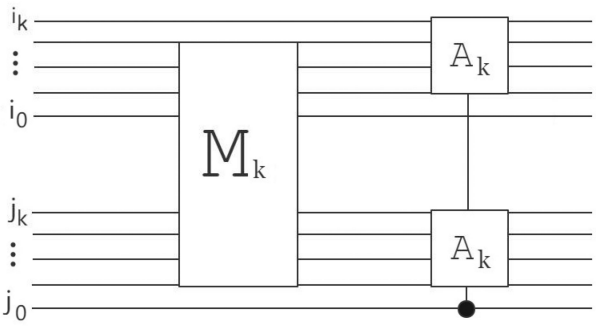

FIGURE 3. Quantum circuit realizing M_{k+1} by M_k and A_k.

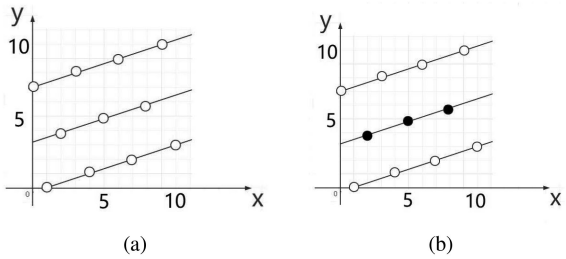

FIGURE 4. Toy example to show the different summation methods adopted to compute the PDRT (left) of f and QPRT (right) of f, where f is a function defined on $\mathbb{Z}_{11}^2$. (a) Points used to compute $r_{-3}(1)$, as defined in (3), i.e., summation along the discrete line $x - 3y = 1 \mod 11$. (b) Points used to compute $\mathrm{QR}_f(1, -3 \mod 2n)$.

shows the line detection capability of such IDRT that can obtain the quantum speedup.

The representation we use for quantum images is the Real Ket (amplitude encoding[1]) method. Since it is not easy to perform interpolation operation in amplitudes of qubits, we realize the IDRT with the output encoded in the computation basis. Interestingly, the output stored in the computation basis can be more easily read out by quantum measurements, making it easier to design linear detection algorithms.

Considering that the Real Ket method and NEQR method are exchangeable (cf., Section II-C), the input and output formats of our quantum Radon algorithms can actually become more flexible; however, it is worth noting that the efficiency of exchange depends on some parameters (e.g., condition numbers) and may be inefficient in certain cases. While we are still many years away from the practical application of quantum computers for fast image processing, the development of theoretical tools and techniques can serve as a basis for developing concrete tools in due course.

The rest of this article is organized as follows. In Section II, we introduce some background on Radon transform and classical/QIMP. In Section III, we present a reversible quantum multiplication. In Section IV, we introduce QPRT and explore some basic properties of it. In Section V, we extend the interpolation-based DRT to the quantum case. In Section VI, we present two applications of our proposed quantum transforms. Finally, Section VII concludes this article.

[1]The famous quantum Fourier transform [17] and HHL algorithm [30] are both based on the amplitude-encoding method.

II. PRELIMINARIES

A. PERIODIC DISCRETE RADON TRANSFORM

Throughout this article, we use p to denote a prime number, use n to denote a positive integer, and use N to denote a power of 2. $[n]$ is the subset of integers $\{0, 1, \ldots, n-1\}$. We use I to denote the imaginary unit. The L^2-norm of vector $\vec{a} = (a_0, \ldots, a_{n-1})$ is $||\vec{a}||_2 = \sqrt{\sum_{i\in[n]} |a_i|^2}$.

We begin with a specific kind of discrete "line," an example of which is given in Fig. 1.

Discrete line $L_{l,k}^n$: The discrete line on lattice $\mathbb{Z}_n^2$ with intercept l and slope k is

$$L_{l,k}^n = \begin{cases} \{(x, y) \,|x + ky = l \mod l,\ x, y \in [n]\}, \text{if } k \in [n] \\ \{(x, l) \,|x \in [n]\}, \quad \text{if} \quad k = n. \end{cases} \tag{2}$$

The PDRT—sometimes called finite Radon transform—is defined as summations of function values at points over these discrete lines.

Definition 1 (Periodic Discrete Radon Transform [9]): The PDRT of a function f defined on lattice $\mathbb{Z}_n^2$ is

$$r_k(l) = \frac{1}{\sqrt{n}} \sum_{(i,j)\in L_{l,k}^n} f(i, j), \quad l \in [n],\ k \in [n+1]. \tag{3}$$

The following is an important property of PDRT.

Proposition 1 (Fourier Slice Property of PDRT [3]): Let $\mathcal{F}_1$ and $\mathcal{F}_2$ be the 1-D and 2-D discrete Fourier transforms, respectively. For a function f defined on $\mathbb{Z}_n^2$, let $r_k(l)$ be the PDRT of f so that it is defined on $\mathbb{Z}_n$. Then, for any $\omega, k \in [n]$

$$\mathcal{F}_1\{r_k\}(\omega) = \mathcal{F}_2\{f\}(\omega, k\omega \bmod n) \tag{4}$$

for $k = n$

$$\mathcal{F}_1\{r_n\}(\omega) = \mathcal{F}_2\{f\}(0, \omega). \tag{5}$$

Proof: For any $k \in [n]$, the discrete Fourier transform of function $r_k(l)$ in variable l is

$$\begin{aligned} \mathcal{F}_1\{r_k\}(\omega) &= \frac{1}{n} \sum_{l,y\in[n]} f(l - ky, y) e^{-2\pi I \frac{l\omega}{n}} \\ &= \frac{1}{n} \sum_{l,y\in[n]} f(l - ky, y) e^{-2\pi I \frac{(l-ky)\omega + yk\omega}{n}} \\ &= \mathcal{F}_2\{f\}(\omega, k\omega \bmod n). \end{aligned} \tag{6}$$

For $k = n$

$$\mathcal{F}_1\{r_n\}(\omega) = \frac{1}{n} \sum_{x,l\in[n]} f(x, l) e^{-2\pi I \frac{l\omega}{n}} = \mathcal{F}_2\{f\}(0, \omega). \tag{7}$$

■

On the one hand, this Fourier slice property provides a fast implementation of PDRT. For an $n \times n$ image f, each value of its PDRT can be computed in time $O(n)$, so the

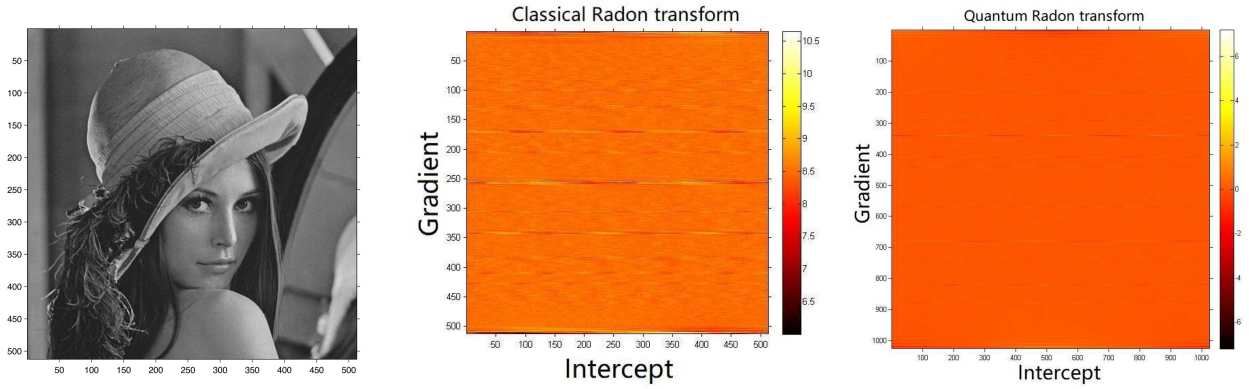

FIGURE 5. **(Left) "Lena" image. (Middle) Classical Radon transform of "Lena." (Right) QPRT of "Lena."**

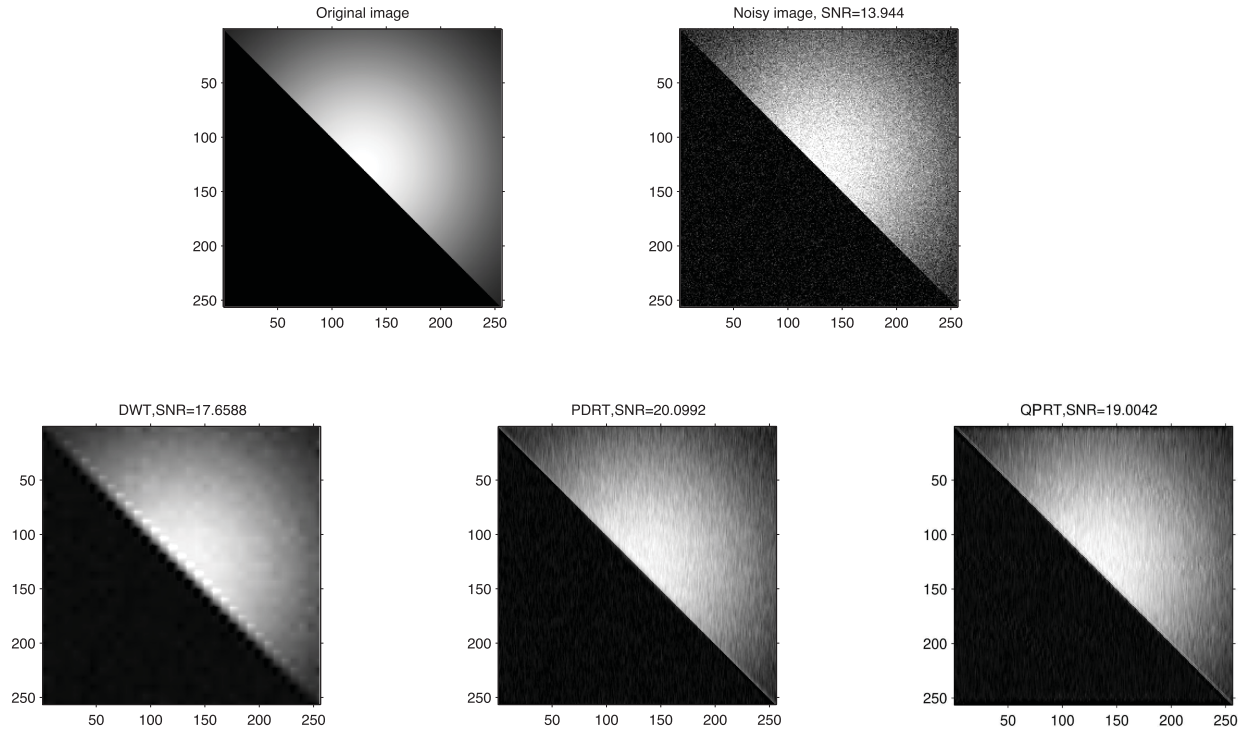

FIGURE 6. **Comparison of denoising half-plane truncated Gaussian function $f(x_1, x_2) = 1_{\{x_1 > x_2\}} e^{(x_1-128)^2-(x_2-128)^2}$. By (12), a higher signal-to-noise ratio (SNR) indicates a better performance of denoising. So, QPRT, with a comparable performance to PDRT, is better than DWT at handling the image, which is piecewise smooth with singularities along a straight line (here is the diagonal line).**

whole PDRT can be obtained in time $O(n^3)$ if directly computed by definition. On the other hand, the Fourier slice property of PDRT allows one to compute the PDRT in time $O(n^2 \log n)$: since the 1-D and 2-D (inverse) Fourier transforms can be implemented in time $O(n \log n)$ and $O(n^2 \log n)$ [31], and the PDRT of f can be obtained by performing 1-D inverse Fourier transform on $\mathcal{F}_2\{f\}$ according to (4), the computational complexity of the PDRT is, thus, reduced to $O(n^2 \log n)$.

Besides designing fast algorithms, another important practical issue is to recover the original image from its Radon transform. In [9], Matúš and Flusser proposed the following reconstruction formula for PDRT on $\mathbb{Z}_p^2$.

Proposition 2 (Reconstruction Formula for PDRT on $\mathbb{Z}_p^2$): Let $r_k(l)$ be the PDRT of a function f defined on $\mathbb{Z}_p^2$. Then, for any $i, j \in [p]$

$$f(i, j) = \frac{1}{\sqrt{p}} \sum_{\substack{\{(l,k)|(i,j)\in L_{l,k}^p, \\ l\in[p],\ k\in[p+1]\}}} r_k(l) - \frac{1}{p} \sum_{x,y\in[p]} f(x, y). \quad (8)$$

Proof: The following geometric properties of discrete lines are easy to verify.

1) Every discrete line $L_{l,k}^p$ contains p lattice points, and two parallel discrete lines have no point of intersection.

IEEE Transactions on Quantum Engineering

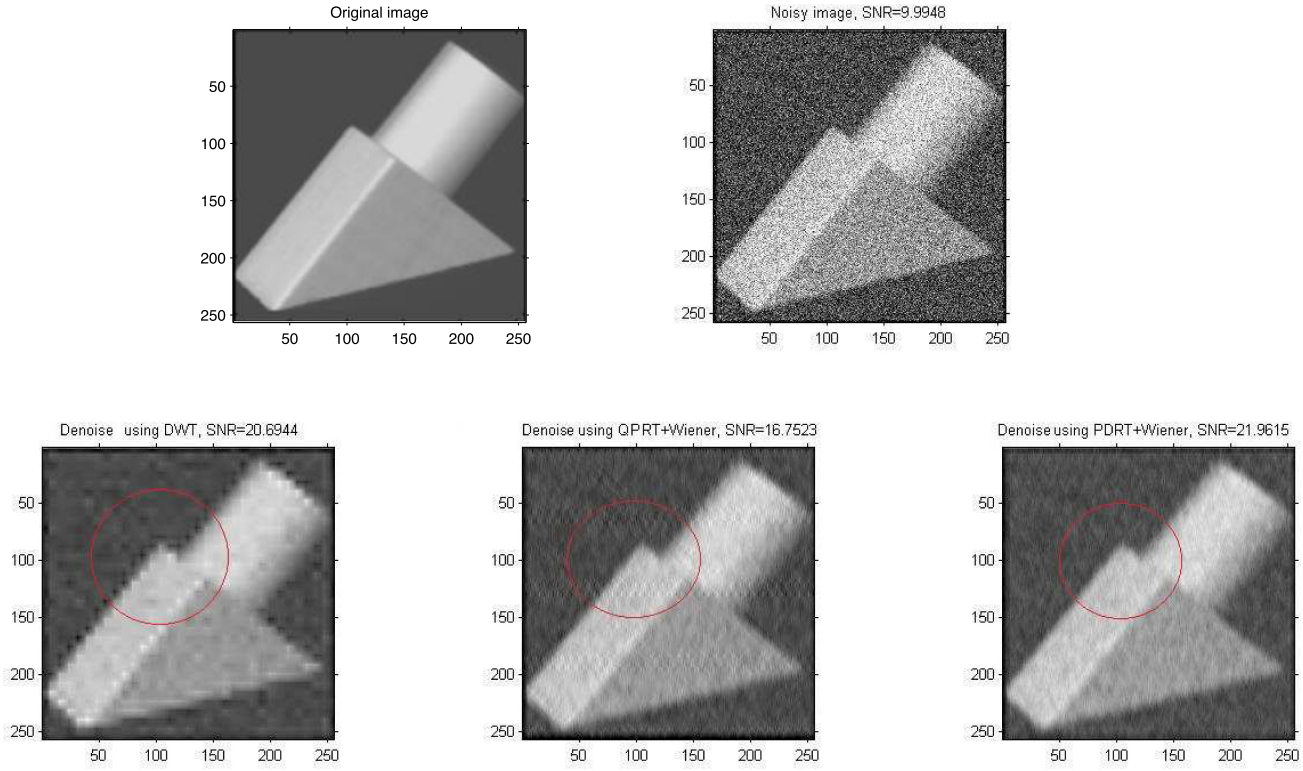

FIGURE 7. In the area marked by the red circles, it can be seen that the QPRT, similar to PDRT, enables to restore the straight edges of geometric solids more clearly than DWT, even in a lower total denoising performance.

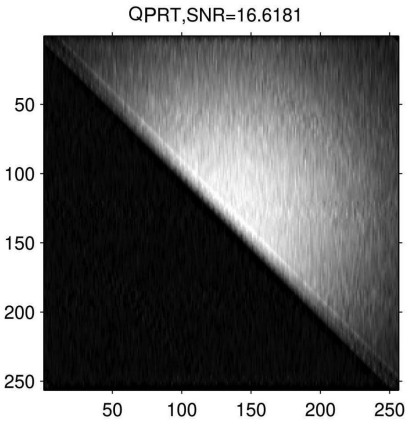

FIGURE 8. Denoise using QPRT + Haar wavelet with setting threshold value ∞ (i.e., make all wavelet coefficients zeros). The original and noisy images are shown in Fig. 6. By denoising, the SNR increases 2.674. The ratio of (L^2-norm of) wavelet coefficients to filter coefficients is 0.0422, which means that the probability of success of the quantum denoising algorithm on Section VI-A is higher than 95% for this example.

2) For any fixed slope k, the p parallel lines $L_{l,k}^p$ (where $l \in [p]$) provide a complete cover of the lattice $\mathbb{Z}_p^2$.
3) Two discrete lines $L_{l,k}^p$ of different slopes will interact in exactly one point.

For all the $p+1$ lines through a fixed point (i, j), by term 1, every two of them have only one point in common, which is just (i, j). Since there are $p(p+1)$ points on the lines, and there are $p+1$ copies of point (i, j) on such lines, there are all together p^2 different points on these lines, which are exactly the total number of points in lattice $\mathbb{Z}_p^2$

$$\sum_{x,y\in[p]} f(x,y) + pf(i,j) = \sqrt{p} \sum_{\substack{\{(l,k)|(i,j)\in L_{l,k}^p,\\ l\in[p],\ k\in[p+1]\}}} r_k(l). \quad (9)$$

■

In general, the size of an image is not the square of a prime number. There is a series of work to extend PDRT to images of more general sizes [9], [10], [32]. For one example, Kingston [33] extends PDRT to images of size $p^n \times p^n$. The reconstruction formula for such images are much more complicated, e.g., [33], eq. (11).

B. CLASSICAL IMAGE DENOISING

Image denoising is to remove noise from a noisy image, so as to restore the true image [34], [35]. Suppose we are given a real-valued noisy signal

$$h_i = f_i + e_i, \quad i \in [n] \quad (10)$$

where $\vec{f} = (f_0, f_1, \ldots, f_{n-1})$ is the original signal, and e_i is the noise sampled independently from the normal distribution $V(0, \sigma^2)$, where 0 is the mean and σ^2 is the variance. After performing some denoising method on $\vec{h} =$

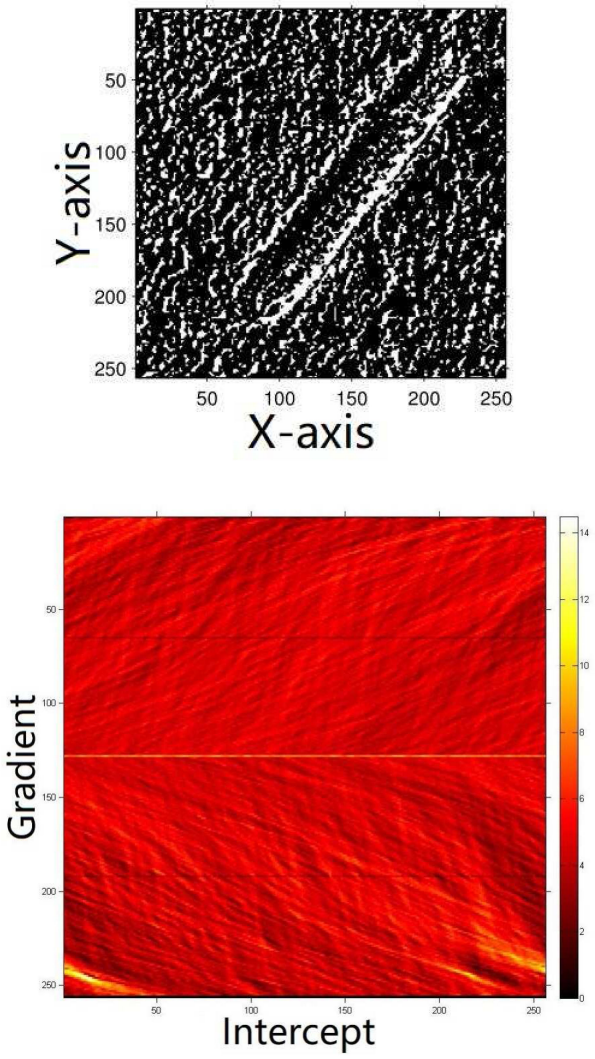

FIGURE 9. **Line detection using SIDRT. The upper is the detected image, where a distinct straight line connecting (228,53) and (97,217) is of the gradient approximately -1.251 and y-intercept $(\frac{53}{1.251} + 228 \mod 256 \approx) 14.366$. The bottom shows the SIDRT of the upper image. The SIDRT reaches maximum at point (13,246), which implies there may be a straight line of the y-intercept 13 and gradient $\frac{1}{\tan(\frac{256-246}{256} - \frac{1}{4})\pi} \approx -1.284$ in the upper image. This detection result is in good agreement with the observation.**

$(h_0, h_1, \ldots, h_{n-1})$ to prepare the proceed signal $\vec{h'}$, if the following noise level decreases (i.e., $\text{Risk}(\vec{h}, \vec{f}) > \text{Risk}(\vec{h'}, \vec{f})$):

$$\text{Risk}(\vec{h}, \vec{f}) = \frac{1}{n} E(||\vec{h} - \vec{f}\,||_2^2) \tag{11}$$

where $E(\cdot)$ is the expectation, then we say such denoising method is effective.

Fig. 2 shows a general procedure for denoising signal using discrete wavelet transform (DWT). We explain why DWT denoising method is effective in the following. Let $\vec{f} = (f_i)$ be a pure signal, where $f_i \equiv 1, i \in [n]$. Let $\vec{h} = (h_i)$ be the noisy signal where $h_i = 1 + e_i$, $e_i \sim V(0, \sigma^2)$. After denoising $\vec{h}$ using Haar wavelet and threshold $T = \infty$ (i.e., change all wavelet coefficients to 0), by Fig. 2, each element of denoised signal is of the form $h'_i = 1 + e'_i$, where

$$e'_i \sim \frac{1}{2} V(0, \sigma^2) + \frac{1}{2} V(0, \sigma^2) = V(0, \sigma^2/2).$$

Now that $\text{Risk}(\vec{h'}, \vec{f}) = \frac{\sigma^2}{2} < \frac{\sigma^2}{n} = \text{Risk}(\vec{h}, \vec{f})$, the noise level decreases, and thus the Haar denoising method works.

However, for general denoising algorithms, it is often hard to make such a statistical analysis as the above-mentioned. Experimentally, the following so-called signal-to-noise ratio (SNR) is often taken to measure denoising performance:

$$\text{SNR}(\vec{h}, \vec{f}) = 10 \log_{10} \left(\frac{||\vec{h}||_2^2}{||\vec{h} - \vec{f}||_2^2} \right) \tag{12}$$

where f and h are some specific test signals. In comparison with the probability value in (11), SNR is more easily accessible by numerical experiments. A denoising method is said to be effective if $\text{SNR}(\vec{h'}, \vec{f}) > \text{SNR}(\vec{h}, \vec{f})$.

Now, we consider image denoising, i.e., 2-D signal denoising. The simplest 2-D DWT denoising method is to apply 1-D denoising method to each row in the signal matrix, then use 1-D denoising method to each column. However, image possesses various geometric features. By the work of Do and Vetterli [3], the following PDRT denoising method is more effective than the general 2-D DWT in denoising images with obvious singularities along straight lines, as shown in Fig. 6.

1) Apply the PDRT to the noisy image to obtain its PDRT $r_k(l)$.
2) For each slope k, perform once 1-D DWT denoising method on the PDRT $r_k(l)$ along the direction of intercept l.
3) Perform inverse PDRT according to (9).

Do and Vetterli [3] explained why PDRT is better: by PDRT, the typical linear singularities of pure image are represented by a few wavelet coefficients (in Step 2), while randomly located noisy singularities are unlikely to produce significant coefficients. This is unlike using the 2-D DWT where both noisy pixels and image singularities can produce significant wavelet coefficients. Therefore, one can remove the noise with less damage to the original image by properly thresholding wavelet coefficients.

A routine computation shows that the time complexity of denoising an $n \times n$ image using 2-D DWT is $O(n^2)$. The time complexity of PDRT denoising method is $O(n^3)$, because performing inverse PDRT has a time complexity $O(n^3)$ by (9).

C. QUANTUM IMAGE REPRESENTATION AND PREPARATION

Definition 2 (Real Ket Representation [22]): Let f be an $N \times N$ image, where $N = 2^n$, and $f(i, j) \geq 0$ is the image

intensity at point (i, j), then its Real Ket representation is

$$\frac{1}{\sqrt{\sum_{i,j\in[N]}|f(i,j)|^2}}\sum_{i,j\in[N]} f(i,j)|i\rangle|j\rangle. \tag{13}$$

Definition 3 (Novel Enhanced Quantum Representation [23]): The NEQR representation of image f is

$$\frac{1}{N}\sum_{i,j\in[N]}|f(i,j)\rangle|i\rangle|j\rangle. \tag{14}$$

In this article, we use the Real Ket representation of images in the quantum computation framework. Unless otherwise specified, the term "quantum image" refers to an image encoded with the Real Ket method.

To show the connection between the previous two representation methods, we need an efficient operation called *conditional rotation* [30], [36].

Proposition 3 (Conditional Rotation): Let a be the k-bit finite precision representation of a positive number that is smaller than 1. Then, the following mapping can be performed in time $O(k^{2.5})$:

$$|a\rangle|0\rangle \to |a\rangle\left(a|0\rangle + \sqrt{1-a^2}|1\rangle\right). \tag{15}$$

Proof: Let $a := \sum_{i\in[k]} 2^{-i-1}a_i$, where $a_i \in \{0, 1\}$. Then, $|a_i\rangle$ is the $(i+1)$st qubit of $|a\rangle = |a_{k-1}\rangle\cdots|a_1\rangle|a_0\rangle$ counted from right. For $s > 0$, define 1-qubit quantum gate

$$R_s = \begin{bmatrix}\cos s, & -\sin s\\ \sin s, & \cos s\end{bmatrix}. \tag{16}$$

Then it holds that

$$\prod_{i\in[k]} R_{2^{-(i+1)}}^{\alpha_i} = \prod_{i\in[k]} R_{2^{-(i+1)}\alpha_i} = R_\alpha. \tag{17}$$

By performing the following k successive 1-bit conditional rotations: for each $i \in [k]$, the corresponding rotation is $R_{2^{-(i+1)}}^{\alpha_i}$, where $\alpha_j \in \{0, 1\}$ is the control bit, on the last qubit of $|a\rangle|0\rangle$, one gets

$$|a\rangle|0\rangle \to |a\rangle(\cos(a)|0\rangle + \sin(a)|1\rangle). \tag{18}$$

By [37, Lemma 48], given $|a\rangle$ where $a \in (0, 1)$, using Taylor series approximation allows to prepare the state $|\arccos(a)\rangle$ in time $O(k^{2.5})$, where $\arccos(a) \in [0, \frac{\pi}{2})$. So with the help of ancilla qubits and (18), the following sequence of mappings can be implemented in time $O(k^{2.5})$:

$$\begin{aligned}|a\rangle|0\rangle|0\rangle &\longrightarrow |a\rangle|\arccos(a)\rangle|0\rangle\\ &\longrightarrow |a\rangle|\arccos(a)\rangle(a|0\rangle + \sqrt{1-a^2}|1\rangle)\\ &\longrightarrow |a\rangle|0\rangle\left(a|0\rangle + \sqrt{1-a^2}|1\rangle\right).\end{aligned}$$

■

By the following proposition, a Real Ket image can be prepared from its NEQR version, and this preparation procedure is efficient if the condition number $\kappa := \frac{\min_{i,j\in[N]}|f(i,j)|}{\max_{i,j\in[N]}|f(i,j)|}$ is large enough.

Proposition 4: Let $\vec{a}$ be a real vector realized by a unitary operator $U : |i\rangle|0\rangle \to |i\rangle|a_i\rangle$, $i \in [N]$, where a_i is the m-bit finite precision representation of the vector entries, and U can be performed in time $O(T_U)$. Given $a_{\max} = \max_j |a_j|$, the state $|\vec{a}\rangle$ can be prepared in time $O(\frac{T_U + poly(m,\log N))}{\kappa^2})$ by the following mapping:

$$|0\rangle \to \cos\theta|\vec{a}\rangle|0\rangle + \sin\theta|\vec{a}^{\perp}\rangle|1\rangle \tag{19}$$

where

1) $|\vec{a}\rangle := \sum_{i\in[N]} \frac{a_i}{||\vec{a}||_2}|i\rangle$;
2) $\vec{a}^{\perp}$ denotes the vector of entries $a_i^{\perp} := \sqrt{1 - |\frac{a_i}{a_{\max}}|^2}$ for $i \in [N]$;
3) the coefficient $\cos\theta = \sqrt{\frac{||\vec{a}||_2^2}{Na_{\max}^2}} \geq \frac{\min_{i\in[N]}|a_i|}{\max_{i\in[N]}|a_i|} := \kappa$, and κ is called the condition number of vector $|a\rangle$.

Proof: With the number $a_{\max}$ and unitary operator U at hand, we can implement the following transform [30]:

$$|i\rangle|a_i\rangle \to |i\rangle\left|\frac{a_i}{a_{\max}}\right\rangle. \tag{20}$$

We first perform the this transform on the input state $\sum_{i\in[N]}\frac{1}{\sqrt{N}}|i\rangle|a_i\rangle$ and then use the rotation conditioned on $|\frac{a_i}{a_{\max}}\rangle$ (cf., Proposition 3)

$$|i\rangle\left|\frac{a_i}{a_{\max}}\right\rangle|0\rangle \to |i\rangle\left|\frac{a_i}{a_{\max}}\right\rangle\left(\frac{a_i}{a_{\max}}|0\rangle + \sqrt{1-\left(\frac{a_i}{a_{\max}}\right)^2}|1\rangle\right) \tag{21}$$

and finally undo U to clear the second register. After the previous operations, we measure the last qubit.

The running time of the previous procedure is $O(T_U + \text{poly}(m, \log N))$. The expected result $|0\rangle$ by measurement has probability $\frac{\sum_{i\in[N]} a_i^2}{Na_{\max}^2} \geq \frac{a_{\min}^2}{a_{\max}^2} = \kappa^2$, which indicates that we have successfully prepared the state $|\vec{a}\rangle := \sum_{i\in[N]}\frac{a_i}{||\vec{a}||_2}|i\rangle$. ■

Remark: It is also possible to convert a Real Ket image to its NEQR version by applying phase estimations [17]. The close connections between different image representations give researchers more confidence in developing QIMP tools based on a particular representation, as they are likely to become universal once quantum techniques are sufficiently developed.

Quantum image preparation, also known as quantum (initial) state preparation, has been extensively studied over years, e.g., [36], [38], and [39]. Current techniques allow to efficiently prepare the Real Ket (or NEQR) state of an $N \times N$ image in time $O(\text{polylog}N)$ if the prepared image has some special structures[2]. The time required to prepare an arbitrary quantum image is at most no more than the classical preparation time, up to a logarithmic factor. Specifically, a quantum image of form (13) or (14) can be prepared by applying at

[2]Such as the corresponding quantum data structure [36] is given or the quantum state to be prepared has a well condition number [40].

most N^2 conditional operations, each of which has a time complexity polylog(N).

The following proposition will be used in Section V. It states that the inner products of quantum states can be estimated in parallel.

Proposition 5 (Parallel Swap Test [41]): Given $2N$ quantum states $|\vec{u}_0\rangle, |\vec{v}_0\rangle \ldots |\vec{u}_{N-1}\rangle, |\vec{v}_{N-1}\rangle$, and two state preparation unitaries: $|k\rangle|0\rangle \to |k\rangle|\vec{u}_k\rangle$ and $|k\rangle|0\rangle \to |k\rangle|\vec{v}_k\rangle$ (where $k \in [N]$) that can be implemented in time $\mathrm{O}(T_{\mathrm{in}})$. There is a quantum algorithm with runtime $\mathrm{O}(\frac{T_{\mathrm{in}}}{\epsilon})$ to achieve $|i\rangle|0\rangle \to |i\rangle|s_i\rangle$ for all $i \in [N]$, where $|s_i - \langle\vec{u}_i|\vec{v}_i\rangle| \le \epsilon$.

More details can be found in [41 Th. 1], or the Appendix in this article.

III. QUANTUM REVERSIBLE MULTIPLICATION

This section presents a reversible modular multiplication in the quantum computation framework. Fix $N = 2^n$. For fixed $0 \le i < 2^{n-1}$ and $C = 2i$, on the one hand, the multiplication in $\mathbb{Z}_N$ by C is irreversible, because the mapping $a \to aC \bmod N$ maps both $a = 0$ and $a = 2^{n-1}$ to 0. On the other hand, for $D = 2i + 1$, the multiplication in $\mathbb{Z}_N$ by D is reversible, because if $aD = bD \bmod N$ for some $a, b \in [N]$ and $a \ne b$, then $(a - b)D \mid 2^n$. Since D is odd, it must be that $a - b$ is a multiple of 2^n; in particular, $|a - b| \ge 2^n$, which is impossible for $a, b \in [N]$.

We design a unitary realization of the multiplication in $\mathbb{Z}_N$ by any odd number $D \in [N]$ in the following. For $1 \le k \le n$,

1) let M_k be a to-be-realized unitary operator performing

$$|a_k\rangle|b_k\rangle \to |a_k\rangle|a_k b_k \bmod 2^k\rangle \forall a_k, b_k \in [2^k] \text{ and odd } a_k \tag{22}$$

2) let A_k be the following controlled addition in $\mathbb{Z}_{2^k}$:

$$|a_k\rangle|b_k\rangle|c\rangle \to \begin{cases} |a_k\rangle|a_k + b_k \bmod 2^k\rangle|c\rangle, & \text{if } c = 1 \\ |a_k\rangle|b_k\rangle|c\rangle, & \text{if } c = 0 \end{cases} \quad \forall a_k, b_k \in [2^k], c \in [2].$$

The time complexity for performing each k-qubit addition is $\mathrm{O}(k^2)$ [42], so is the time complexity of performing unitary operator A_k (see [17, Sec. 4.2]).

For any $1 \le k \le n$, for any $a_k \in [2^k]$, let the binary representation of integer a_k be

$$B_{i_{k-1}\ldots i_0} := \sum_{l\in[k]} 2^l i_l \quad \text{where } i_l \in [2].$$

When $k = 1$, multiplication operator M_1 is the identity

$$|1\rangle|b_1\rangle \to |1\rangle|b_1\rangle \quad \forall\, b_1 \in [2].$$

In the following, we realize multiplication operator M_{k+1} by A_l for $1 \le l \le k$ recursively.

For any $a_{k+1}, b_{k+1} \in [2^{k+1}]$, where a_{k+1} is odd, let their binary representations be $B_{i_k\ldots i_1 1}$ and $B_{j_k\ldots j_1 j_0}$, respectively. Then their modular multiplication is

$$\begin{aligned}&(2^k i_k + B_{i_{k-1}\ldots i_1 1})(2B_{j_k\ldots j_1} + j_0) \bmod 2^{k+1}\\ &= \left(2B_{i_{k-1}\ldots i_1 1}B_{j_k\ldots j_1} + B_{i_k\ldots i_1 1}\frac{1 + (-1)^{j_0-1}}{2}\right) \bmod 2^{k+1}\\ &= \left\{2(B_{i_{k-1}\ldots i_1 1}B_{j_k\ldots j_1} \bmod 2^k) + 2\delta^1_{j_0}B_{i_k\ldots i_1} + j_0\right\} \bmod 2^{k+1}\end{aligned} \tag{23}$$

where $\delta^1_{j_0}$ is the Kronecker symbol.

So the modular multiplication M_{k+1} of two integers $B_{i_k\ldots i_1 1}$ and $B_{j_k\ldots j_1 j_0} \in [2^{k+1}]$, each containing $k + 1$ binary digits, can be decomposed into two operators: the modular multiplication M_k of integers $B_{i_{k-1}\ldots i_1 1}, B_{j_k\ldots j_1} \in [2^k]$, which occurs in the first k binary digits, followed by the controlled modular addition A_k of the previous modular multiplication result and $B_{i_k\ldots i_1}$, which also occurs in the first k binary digits, while the control digit $j_0 \in [2]$ remains in the last binary digit. The quantum circuit realizing this decomposition is shown in Fig. 3.

In (23), the modular multiplication M_k can be further decomposed into M_{k-1} and A_{k-1}, and by doing so recursively, M_{k+1} is finally decomposed into a series of controlled modular additions: $A_1, A_2, \cdots, A_k$. Since performing A_j has time complexity $\mathrm{C}j^2$ for fixed constant $C > 0$ and varying $1 \le j \le k$, by $1^2 + 2^2 + \cdots + k^2 = O(k^3)$, we get that the time complexity of realizing M_{k+1} (hence M_k) by (14) is $\mathrm{O}(k^3)$.

IV. QUANTUM PERIODIC DISCRETE RADON TRANSFORM

The classical PDRT in (3) can be viewed as a mapping $\mathbb{R}^{n^2} \to \mathbb{R}^{n(n+1)}$ as follows:

$$\begin{aligned}&(f(0,0), f(1,0), \ldots, f(n-1, n-1))\\ &\xrightarrow{\text{PDRT}} (r_0(0), r_0(1), \ldots, r_n(n-1))\end{aligned} \tag{24}$$

whose transformation matrix is not unitary.[3] It is hard to directly design a quantum algorithm for PDRT by (3). So we consider utilizing the related Fourier slice property in (4).

By replacing the traditional multiplication in (4) with the quantum reversible multiplication, we successfully design an efficient quantum algorithm that can realize a novel transform similar to PDRT. We name it the QPRT.

Definition 4 (Quantum Periodic Discrete Radon Transform): Let f be a function defined on $\mathbb{Z}_n^2$, let $\lfloor\cdot\rfloor$ be the floor function, and let $\tilde{f}$ defined on $\mathbb{Z}_{2n}^2$ be related to f as

$$\tilde{f}(x', y') := \frac{1}{2}(-1)^{\lfloor\frac{x'}{n}\rfloor + \lfloor\frac{y'}{n}\rfloor} \times f(x' \bmod n, y' \bmod n), x', y' \in [2n]. \tag{25}$$

Recall that $L^{2n}_{l,k}$ represent discrete lines on the lattice $\mathbb{Z}_{2n}^2$

$$L^{2n}_{l,k} := \{(x', y') | x' + ky' = l \ (\mathrm{mod}\ 2n),\ x',\ y' \in [2n]\}.$$

[3]The lines $L^n_{0,0}$ and $L^n_{0,1}$ have intersection, so the rows of transformation matrix corresponding to $r_0(0)$, $r_1(0)$ are not orthogonal.

Then, the QPRT of f is a function defined on $\mathbb{Z}_{2n}^2$ as

$$\mathrm{QR}_f(l,k) = \frac{1}{\sqrt{2n}} \sum_{(x',y')\in L_{l,k}^{2n}} \tilde{f}(x',y'), \quad l,k \in [2n]. \tag{26}$$

The definition of QPRT is derived from the following Algorithm 1 rather than a deliberate construction, and it may seem a little complicated. Now, let us give a closer observation of $\tilde{f}$ in (25). For any $x, y \in [n]$ 1

$$\begin{aligned} f(x,y) &= \tilde{f}(x,y) = -\tilde{f}(x+n,y) = -\tilde{f}(x,y+n) \\ &= \tilde{f}(x+n,y+n). \end{aligned} \tag{27}$$

Graphically speaking, the QPRT of an image f can be viewed as the classical PDRT that performs on the symmetrized double-sized original image, Fig. 5. Fig. 4 shows the particular summation method adopted by the QPRT. Intuitively, the alternating sums seems to tend to suppress the changes in the Radon domain along the direction of intercept, resulting in a lower sensitivity of QPRT to linear singularities. Even worse, only a half of slopes (lines) could be detected by QPRT since when slope k is even, by (27) it holds that

$$\mathrm{QR}_f(l,k) = 0. \tag{28}$$

However, by experiments (see Figs. 6 and 7), it has been found that QPRT preserves the good denoising capability as in the classical PDRT, and it will be further discussed in Section VI-A later. On the one hand, we hold the view that the QPRT has a practical value comparable to that of the classical PDRT. On the other hand, as an advantage, QPRT is invertible, which is derived from the following invertible QPRT algorithm. The most important advantage lies in that the QPRT can be performed exponentially faster than the classical PDRT by the following QPRT algorithm (see Algorithm 1).

A. ANALYSIS OF ALGORITHM 1

The Step 1 of Algorithm 1 is adding two extra qubits initially in the state $|1\rangle|1\rangle$. In Step 2, the conditional phase shifts can be implemented in time logarithmic in N, by a method similar to the implementation of conditional rotation (cf., Proposition 3). After applying Step 3, the resulting state can be viewed as the quantum Fourier transform of some $(2N+2)$-qubit state $\sum_{x',y'\in[2N]} g'(x',y')|x'\rangle|y'\rangle$, i.e.,

$$\begin{cases} \mathcal{F}_2\{g'\}(2i+1,2j+1) = \mathcal{F}_2\{g\}(i,j), & i,j \in [N] \\ \mathcal{F}_2\{g'\}(i,j) = 0, & i,j \in [2N] \text{ are not both odd.} \end{cases} \tag{34}$$

The explicit expression of function g' can be deduced by inverse Fourier transform as follows:

$$\begin{aligned} g'(l',t') &= \frac{1}{2N}\frac{1}{N} \sum_{x,y,i,j\in[N]} g(x,y) e^{-2\pi I \frac{ix+jy}{N}} e^{2\pi I \frac{(2i+1)l'+(2j+1)t'}{2N}} \\ &= \frac{1}{2} g\left(l' \bmod N, t' \bmod N\right) e^{2\pi I \frac{l'+t'}{2N}} \\ &= \tilde{f}(l',t') l', t' \in [2N] \end{aligned}$$

Algorithm 1: Quantum Periodic Discrete Radon Transform.

Input: An $N \times N$ quantum image $\sum_{x,y\in[N]} f(x,y)|x\rangle|y\rangle$.

Output: $\sum_{l,j''\in[2N]} \mathrm{QR}_f(l,j'')|l\rangle|j''\rangle$.

1: Prepare the state $\sum_{x,y\in[N]} f(x,y)|x\rangle|1\rangle|y\rangle|1\rangle$.

2: Apply phase shifts conditioned on $|x\rangle$, $|y\rangle$ to prepare

$$\sum_{x,y\in[N]} g(x,y)|x\rangle|1\rangle|y\rangle|1\rangle \tag{29}$$

where $g(x,y) = f(x,y)e^{-2\pi I \frac{x+y}{2N}}$.

3: Perform quantum Fourier transforms on $|x\rangle$ and $|y\rangle$, respectively, to get

$$\sum_{i,j\in[N]} \left(\frac{1}{N} \sum_{x,y\in[N]} g(x,y) e^{-2\pi I \frac{ix+jy}{N}} \right) |i\rangle|1\rangle|j\rangle|1\rangle. \tag{30}$$

4: Perform inverse quantum reversible multiplication on registers $|i'\rangle = |i,1\rangle$ and $|j'\rangle = |j,1\rangle$ to prepare

$$\sum_{i',j''\in[2N]} \left(\frac{1}{2N} \sum_{x',y'\in[2N]} \tilde{f}(x',y') e^{-2\pi I \frac{i'x'+(i'\odot j'')y'}{2N}} \right) |i'\rangle|j''\rangle \tag{31}$$

where $i' \odot j'' = j'$, $\odot$ is the quantum reversible multiplication such that $i' \odot j'' = i'j''$ for odd $i' \in [2N]$.

5: Apply inverse quantum Fourier transform to $|i'\rangle$ to prepare

$$\begin{aligned} &\frac{1}{\sqrt{2N}} \sum_{l,j''\in[2N]} \left(\frac{1}{2N} \sum_{i',t\in[2N]} \sum_{(x',y')\in L_{t,j''}^{2N}} \tilde{f}(x',y') \right. \\ &\qquad \left. \times e^{-2\pi I \frac{i't}{2N}} e^{2\pi I \frac{i'l}{2N}} \right) |l\rangle|j''\rangle \qquad (32) \\ &= \sum_{l,j''\in[2N]} \mathrm{QR}_f(l,j'')|l\rangle|j''\rangle. \qquad (33) \end{aligned}$$

so that the resulting state in Step 3 can be rewritten as

$$\sum_{i',j'\in[2N]} \left(\frac{1}{2N} \sum_{x',y'\in[2N]} \tilde{f}(x',y') e^{-2\pi I \frac{i'x'+j'y'}{2N}} \right) |i'\rangle|j'\rangle. \tag{35}$$

After applying the inverse quantum reversible multiplication to $|i'\rangle|j'\rangle$ of (35) in Step 4, by (22), the resulting state is as in (31). Now, from the fact that for any $j'' \in [2N]$

$$\begin{cases} i' \odot j'' = i'j'', & \text{if } i' \text{ is odd} \\ \mathcal{F}_2\{\tilde{f}\}(i',j'') = 0, & \text{if } i' \text{ is even} \end{cases}$$

one gets

$$\mathcal{F}_2\{\tilde{f}\}(i', i' \odot j'') = \mathcal{F}_2\{\tilde{f}\}(i', i'j''), \qquad i, j \in [2N].$$

Then the resulting state in Step 4 can be rewritten as

$$\sum_{i', j'' \in [2N]} \mathcal{F}_2\{\tilde{f}\}(i', i'j'')|i'\rangle|j''\rangle.$$

The final result in Step 5 is by the following proposition.

Proposition 6 (Fourier Slice Property of QPRT): In the notations of Algorithm 1, for any given slope $j'' \in [2N]$, the 1-D Fourier transform of $\mathrm{QR}_f(l, j'')$ with respect to l is

$$\mathcal{F}_1\{\mathrm{QR}_f(\cdot, j'')\}(i') = \mathcal{F}_2\{\tilde{f}\}(i', i'j''), \quad i' \in [2N]. \tag{36}$$

Proof: Let $\mathcal{F}_1^{-1}$ be the inverse transform of $\mathcal{F}_1$. Then

$$\begin{aligned}
&\mathcal{F}_1^{-1}\{\mathcal{F}_2\{\tilde{f}\}(\cdot, \cdot j'')\}(l) \\
&= \frac{1}{\sqrt{2N}} \sum_{i' \in [2N]} \left(\frac{1}{2N} \sum_{x', y' \in [2N]} \tilde{f}(x', y') e^{-2\pi I \frac{i'x' + i'j''y'}{2N}} \right) e^{2\pi I \frac{i'l}{2N}} \\
&= \frac{1}{\sqrt{2N}} \left(\frac{1}{2N} \sum_{i', t \in [2N]} \sum_{(x', y') \in L^{2N}_{t, j''}} \tilde{f}(x', y') e^{-2\pi I \frac{i't}{2N}} e^{2\pi I \frac{i'l}{2N}} \right) \\
&= \frac{1}{\sqrt{2N}} \sum_{i', t \in [2N]} \left(\frac{1}{2N} \sum_{(x', y') \in L^{2N}_{t, j''}} \tilde{f}(x', y') \right) e^{-2\pi I \frac{i't}{2N}} e^{2\pi I \frac{i'l}{2N}} \\
&= \frac{1}{\sqrt{2N}} \sum_{(x', y') \in L^{2N}_{l, j''}} \tilde{f}(x', y') = \mathrm{QR}_f(l, j''). \qquad (37)
\end{aligned}$$

■

The overall runtime of Algorithm 1 is the sum of $O(\log N)$ time required to perform conditional phase shifts [43], $O(\log^2 N)$ time required to perform quantum Fourier transform and its inverse, and $O(\log^3 N)$ time required to perform the inverse multiplication. Observe that all quantum gates used in Algorithm 1 are unitary and invertible. So, an efficient inverse QPRT algorithm follows immediately by reversing all circuits in Algorithm 1. In conclusion, we have the following theorem.

Theorem 1: Given an $N \times N$ quantum image state

$$\sum_{x, y \in [N]} f(x, y)|x\rangle|y\rangle$$

then its (inverse) QPRT can be performed in time $\mathrm{O}(\log^3 N)$.

V. QUANTUM-MECHANICAL IMPLEMENTATION OF DRT WITH THE INTERPOLATION METHOD

The discretization of Radon transform needs to approximate the line integral. A simple solution is to sample and sum along the integral line, where the interpolation method can be employed to estimate the undefined values at noninteger lattice sample points. Several kinds of sample and interpolation methods are available for implementing IDRT [6], [44], [45]. We define a simplest kind of IDRT (SIDRT), and give its quantum algorithm, with the aim of showing a general quantum approach to achieve this interpolation-based kind of DRT in the following.

Definition 5: Let the set of slopes of the basically horizontal lines and basically vertical lines be

$$\begin{aligned}
S^{\parallel} &:= \left\{ \tan\theta_j | \theta_j = \frac{\pi j}{N}, -\frac{N}{4} \le j < \frac{N}{4} \text{ and } j \in \mathbb{Z} \right\} \\
S^{\perp} &:= \left\{ \frac{1}{\tan\theta_j} | \theta_j = \frac{\pi j}{N}, -\frac{N}{4} < j \le \frac{N}{4} \text{ and } j \in \mathbb{Z} \right\} \qquad (38)
\end{aligned}$$

respectively, where agree that $\frac{1}{0} = \infty$ and $\frac{1}{\infty} = 0$. Let Δ_{ki} be the fractional part of ki. Then, the SIDRT of a function f on $\mathbb{Z}_N^2$ is

$$\begin{aligned}
P_k(l) = \frac{1}{\sqrt{N}} \sum_{i \in [N]} \Big[&\sqrt{1 - |\Delta_{ki}|^2} f(i, l + \lfloor ki \rfloor \bmod N) \\
&+ \Delta_{ki} f(i, l + \lfloor ki \rfloor + 1 \bmod N) \Big], k \in S^{\parallel}, l \in [N] \qquad (39)
\end{aligned}$$

$$\begin{aligned}
P_k(l) = \frac{1}{\sqrt{N}} \sum_{i \in [N]} \Big[&\sqrt{1 - |\Delta_{\frac{i}{k}}|^2} f\left(l + \lfloor \frac{i}{k} \rfloor \bmod N, i \right) \\
&+ \Delta_{\frac{i}{k}} f\left(l + \lfloor \frac{i}{k} \rfloor + 1 \bmod N, i \right) \Big], k \in S^{\perp}, l \in [N]. \qquad (40)
\end{aligned}$$

By definition, $P_k(l)$ can be viewed as an approximate discrete line integral along the line with intercept l and slope k.

The SIDRT is proposed for easily achieved in the quantum case. Although it adopts a relatively simple interpolation method, the SIDRT is so useful that enables to detect lines in complicated image, as shown in Fig. 9; more details can be found in Section VI-B.

Now, we turn to the quantum implementation of the SIDRT. We first consider approximating the integrals along basically horizontal lines, i.e., $P_k(l)$ as defined in (39), where $l \in [N]$ and $k \in S^{\parallel}$.

Let $k_\theta := \tan(\frac{\pi\theta}{N} - \frac{\pi}{4})$. For any $i \in [N]$, $\theta \in [\frac{N}{2}]$, we can prepare a quantum state that contains the location information of lattice points used to compute $P_{k_\theta}(l)$ by the following sequence of mappings:

$$\begin{aligned}
|i\rangle|\theta\rangle|0\rangle|0\rangle &\longrightarrow |i\rangle|\theta\rangle|ik_\theta\rangle|0\rangle \\
&\longrightarrow |i\rangle|\theta\rangle\big|\lfloor ik_\theta \rfloor\big\rangle|\Delta_{ik_\theta}\rangle \left(\sqrt{1 - |\Delta_{ik_\theta}|^2}|0\rangle + \Delta_{ik_\theta}|1\rangle \right) \\
&\longrightarrow |i\rangle|\theta\rangle\big|\lfloor ik_\theta \rfloor\big\rangle|0\rangle \left(\sqrt{1 - |\Delta_{ik_\theta}|^2}|0\rangle + \Delta_{ik_\theta}|1\rangle \right). \qquad (41)
\end{aligned}$$

In (41), the first step is by trigonometric function [37] and arithmetics [17] in computation basis. The second step is to use the decimal part of ik_θ to perform control rotations (cf., Proposition 3) on the last qubit. The third step is uncomputing $|\Delta_{ik_\theta}\rangle$.

Combining (41) with the addition in the computational basis gives the following lemma.

Lemma 2: The following "location state" can be prepared in time $O(polylogN)$:

$$|\theta\rangle|0\rangle \xrightarrow{V} |\theta\rangle \sum_{i\in[N]} \frac{1}{\sqrt{N}}\Big(\sqrt{1-|\Delta_{ik_\theta}|^2}|i\rangle\big|\lfloor ik_\theta\rfloor\big\rangle|0\rangle + \Delta_{ik_\theta}|i\rangle\big|\lfloor ik_\theta\rfloor+1\big\rangle|1\rangle\Big), \quad \theta\in\left[\frac{N}{2}\right]. \tag{42}$$

Remark: The state on the right-hand side of (42) is called "location state," since its first two quibts record the locations of points used to compute $P_k(0)$ for $k\in S^{\|}$, according to (39).

Now, with the unitary capable of preparing a quantum image f as follows:

$$|0\rangle \longrightarrow \sum_{i,j\in[N]} f(i,j)|i\rangle|j\rangle \tag{43}$$

one can approximate the SIDRT of f by the following theorem.

Theorem 3: Given the unitary that can prepare a quantum image $\sum_{i,j\in[N]} f(i,j)|i\rangle|j\rangle$ in time $O(T_{in})$, one can approximate $P_k(l)$, the SIDRT of f, to precision $O(\epsilon)$ in time $O(\frac{T_{in}}{\epsilon}polylogN)$. Namely, there is a quantum algorithm runs in time $O(\frac{T_{in}}{\epsilon}polylogN)$ to achieve the mapping

$$|\theta\rangle|l\rangle|0\rangle \to |\theta\rangle|l\rangle|\tilde{P}_{k_\theta}(l)\rangle \qquad \forall l\in[N],\theta\in[N] \tag{44}$$

where $|\tilde{P}_{k_\theta}(l)-P_{k_\theta}(l)|\le\epsilon$, and $k_\theta := \tan(\frac{\pi\theta}{N}-\frac{\pi}{4})$ such that $k_\theta\in S^{\|}\cup S^{\perp}$, as defined in (38).

Proof: We first consider the realization of (44) for $\theta\in[\frac{N}{2}]$, i.e., the case where $k_\theta\in S^{\|}$. Observe that the following "image state" can be prepared in time $O(T_{in}polylogN)$ by the mapping

$$|\theta\rangle|l\rangle|0\rangle \longrightarrow |\theta\rangle|l\rangle \sum_{i,j\in[N]} f(i,j)|i\rangle|j\rangle\left(\frac{1}{\sqrt{2}}|0\rangle+\frac{1}{\sqrt{2}}|1\rangle\right) \tag{45}$$

where $\theta\in[\frac{N}{2}]$ and $l\in[N]$. Lemma 2 allows us to prepare the "location state" by the mapping

$$|\theta\rangle|l\rangle|0\rangle \longrightarrow |\theta\rangle|l\rangle \sum_{i\in[N]} \frac{1}{\sqrt{N}}\left(\sqrt{1-|\Delta_{ik_\theta}|^2}|i\rangle\big|l+\lfloor ik_\theta\rfloor\big\rangle|0\rangle + \Delta_{ik_\theta}|i\rangle\big|l+\lfloor ik_\theta\rfloor+1\big\rangle|1\rangle\right) \tag{46}$$

in time $O(polylogN)$, where $\theta\in[\frac{N}{2}]$ and $l\in[N]$.

Now, by parallel swap test (Proposition 5), one can approximate the $\frac{N^2}{2}$ inner products of the "image state" in (45) and the "location state" in (46) in parallel θ, l. Since the result of each inner product is just $\frac{1}{\sqrt{2}}P_{k_\theta}(l)$, one can perform

$$|\theta\rangle|l\rangle|0\rangle \to |\theta\rangle|l\rangle|p_{k_\theta}(l)\rangle \quad \forall\theta\in\left[\frac{N}{2}\right],\ l\in[N] \tag{47}$$

in time $O(\sqrt{2}\frac{T_{in}}{\epsilon}polylogN)$, where $|p_{k_\theta}(l)-\frac{1}{\sqrt{2}}P_{k_\theta}(l)|\le\frac{\epsilon}{\sqrt{2}}$. The theorem holds by setting $\tilde{P}_{k_\theta}(l)=\sqrt{2}p_{k_\theta}(l)$. An efficient implementation of multiplication by $\sqrt{2}$, a known constant, in the computation basis can be found in [46]. The realization of (44) for $\theta\in[N]\setminus[\frac{N}{2}]$ is similar. ■

Given a Real Ket quantum image f and its preparation unitary as in (43), by Theorem 3, after performing the transform of (44) on input state $\sum_{\theta,l\in[N]}\frac{1}{N}|\theta\rangle|l\rangle|0\rangle$, one can prepare the SIDRT of f in NEQR-encoded form. This NEQR-encoded output already can be used for practical application, such as line detection shown in Section VI-B later. Moreover, if necessary, one can continue to transform the NEQR-encoded output into its Real Ket version by Proposition 4, thus keeping the input and output encoded in the same way.

A. EFFICIENCY ANALYSIS OF THEOREM 3

We discuss what is a reasonable choice of the precision ϵ in Theorem 3.

Random image. The term "random image" refers to an image whose each pixel value is sampled from the uniform distribution $\mathbb{U}[0,1]$ independently.

Proposition 7: Let f be an $N\times N$ random image, i.e., $f(i,j)\sim\mathbb{U}[0,1]$ for $i,j\in[N]$. Then, the expectation of the minimal value of the SIDRT of normalized f is no less than $\frac{\sqrt{3}}{2N^{0.5}}$.

Remark: The reason why we consider the normalized rather than the original image here is that the quantum image is a normalized state.

Proof: We begin with a probability inequality. Let a and b be two discrete random variables whose density functions are $P(a=a_i)=p_i$ and $P(b=b_i)=q_i$, where $i\in[N]$, respectively. Then, by Cauchy-Schwarz inequality

$$E\left(\frac{a}{b}\right)=\sum_{i,j\in[N]}p_iq_j\frac{a_i}{b_j}=\frac{\left(\sum_{i,j}p_iq_jb_j\right)\left(\sum_{i,j}p_iq_j\frac{a_i}{b_j}\right)}{E(b)} \ge \frac{\left(\sum_{i,j}q_jp_i\sqrt{a_i}\right)^2}{E(b)}=\frac{\left(\sum_i p_i\sqrt{a_i}\right)^2}{E(b)}=\frac{E^2\left(\sqrt{a}\right)}{E(b)}. \tag{48}$$

Let $\mathbb{U}[0,1]$ be the uniform distribution on [0,1], and let random variables $x_k, y_k\sim\mathbb{U}[0,1]$ for $k\in[N^2]$. Notice that $f(i,j)\sim\mathbb{U}[0,1]$. Let $P_{k_0}(l_0)$ be the SIDRT of f as in (39), then for any $k_0, l_0\in[N]$, the expectation of the SIDRT of normalized f at point (k_0,l_0) is

$$E\left(\frac{P_{k_0}(l_0)}{\sqrt{\sum_{i,j\in[N]}f^2(i,j)}}\right) = E\left(\frac{\frac{1}{\sqrt{N}}\sum_{k\in[N]}\Delta_k x_k+\sqrt{1-|\Delta_k|^2}x_k}{\sqrt{\sum_{k\in[N]}x_k^2+\sum_{k\in[N^2-N]}y_k^2}}\right)$$

$$\geq E\left(\frac{\frac{1}{\sqrt{N}}\sum_{k\in[N]}x_k}{\sqrt{\sum_{k\in[N]}1+\sum_{k\in[N^2-N]}y_k^2}}\right)$$
$$\overset{(48)}{\geq}\frac{E^2\left(\sqrt{\frac{1}{\sqrt{N}}\sum_{k\in[N]}x_k}\right)}{E\left(\sqrt{N+\sum_{k\in[N^2-N]}y_k^2}\right)}$$
$$\geq\frac{E^2\left(\sqrt{\frac{1}{\sqrt{N}}\sum_{k\in[N]}x_k}\right)}{\sqrt{E\left(N+\sum_{k\in[N^2-N]}y_k^2\right)}}\approx\frac{\sqrt{3}}{2N^{0.5}}\qquad(N\to\infty)\tag{49}$$

where the last inequality follows from that $E^2(\nu)\leq E(\nu^2)$ for arbitrary random variable ν, and the last approximate equality is by the central limit theorem [47], which states that for any independent and identically distributed (IID) random variables $\nu_0,\nu_1,\ldots,\nu_{n-1}$ with mean μ and variance σ^2, it holds that $\mathrm{E}(\sqrt{|\sum_{j\in[n]}\nu_j|})\approx\sqrt{n\mu-\frac{\sigma^2}{\mu}}$ $(n\to\infty)$, and so $E(\sqrt{\sum_{k\in[N]}x_k})\approx\sqrt{\frac{N}{2}-\frac{1}{6}}$.

The previous proposition implies that a reasonable choice of precision in Theorem 3 is $\epsilon=\Theta(\frac{1}{N^{0.5}})$, e.g., $\frac{1}{100N^{0.5}}$, for producing a good approximation to SIDRT. In this case, when compared with the classical SIDRT whose running time is $\Omega(N^3)$, quantum SIDRT achieves a polynomial speedup by Theorem 3, because the quantum image preparation time $T_{\text{in}}\leq O(N^2)$ in the worst case, thus $\frac{T_{\text{in}}}{\epsilon}\leq O(N^{2.5})<N^3$.

VI. QUANTUM APPLICATION

We present two potential applications—quantum image denoising using QPRT and quantum line detection using SIDRT.

A. DENOISING USING QPRT

We replicate the denoising experiments in [3], which are specially designed for testing PDRT. The experiment results shown in Figs. 6 and 7 suggest that our QPRT is of the good denoising capability as in the classical PDRT. We give an efficient quantum image denoising algorithm using QPRT in the following.

Step 1: Apply the QPRT algorithm on the input $N\times N$ quantum image f to perform

$$\sum_{x,y\in[N]}f(x,y)|x\rangle|y\rangle\longrightarrow\sum_{l,k\in[2N]}\mathrm{QR}_f(l,k)|l\rangle|k\rangle.\tag{50}$$

Step 2: Perform Hadamard gate (namely, 2×2 Haar transform, as shown in Fig. 2) on the least significant qubit (LSB) of $|l\rangle$ in (50) to produce a state

$$\frac{1}{\sqrt{2}}\sum_{l'\in[N],k\in[2N]}(\mathrm{QR}_f(2l',k)+\mathrm{QR}_f(2l'+1,k))|l'0\rangle|k\rangle+\big(\mathrm{QR}_f(2l',k)-\mathrm{QR}_f(2l'+1,k)\big)|l'1\rangle|k\rangle.\tag{51}$$

Step 3: Measure the LSB of the first register. If the outcome is $|0\rangle$, then apply the inverse Hadamard gate to the measured qubit, and then execute the inverse QPRT algorithm. The resulting state is (up to a normalization factor) given as follows:

$$\mathrm{QR}^{-1}\left(\sum_{\substack{l'\in[N],s\in[2]\\k\in[2N]}}\frac{1}{2}(\mathrm{QR}_f(2l',k)+\mathrm{QR}_f(2l'+1,k))|l's\rangle|k\rangle\right).$$

The above-mentioned is a quantum analogue of PDRT denoising method introduced in Section II-B. Notice that the Hadamard gate in Step 2 plays the role of Haar wavelet, and measuring LSB to obtain $|0\rangle$ in Step 3 has the effect of making all wavelet coefficients zero. So, this quantum QPRT denoising method indeed simulates denoising using QPRT + Haar wavelet and threshold ∞.

To evaluate the efficiency of this denoising method, we consider the success probability (namely, the probability of measuring $|0\rangle$ in Step 3). Since $\mathrm{QR}_f(l',k)=0$ holds for any even $k\in[2N]$, the success probability is, thus, reduced to

$$\frac{1}{2}\sum_{l',k\in[N]}|\mathrm{QR}_f(2l',2k+1)+\mathrm{QR}_f(2l'+1,2k+1)|^2.\tag{52}$$

By the following theoretical analysis (cf., Proposition 8) and numerical simulation (cf., Fig. 8), this probability of success is quite high; particularly, it has no tendency to be tiny as the image size N increases.

Proposition 8: Let f be an $N\times N$ random image, i.e., $f(i,j)\sim\mathbb{U}[0,1]$ for $i,j\in[N]$. Given the noisy image $h(i,j)=f(i,j)+\epsilon e_{ij}$, $i,j\in[N]$, where e_{ij} is the noise sampled from the normal distribution $V(0,\sigma^2)$ independently, and $\epsilon\in\mathbb{R}$ is noise level, then the QPRT denoising method can be implemented with an average probability of success $p>1/2$.

Proof: By Definition 4, for any fixed $l,k\in[2N]$

$$\mathrm{QR}_h(l,k)-\mathrm{QR}_f(l,k)=\frac{1}{2\sqrt{2N}}\sum_{(i,j)\in L_{l,k}^{2N}}\epsilon e_{ij}.\tag{53}$$

Notice that for two independent random variables $x\sim V(u_1,\sigma_1^2)$ and $y\sim V(u_2,\sigma_2^2)$, it holds that $x\pm y\sim V(u_1\pm u_2,\sigma_1^2+\sigma_2^2)$. So, the distribution of the right-hand side of (53) is $\epsilon V(0,\sigma^2/4)$. Let $E_{f,e}(\mathrm{QR}_h)$ denote the expectation of QR_h with respect to the random variables f and e. Then, for any $l,k\in[2N]$

$$E_{f,e}(|\mathrm{QR}_h(l,k)-\mathrm{QR}_h(l+1,k)|^2)=E_f(|\mathrm{QR}_f(l,k)-\mathrm{QR}_f(l+1,k)|^2)+E_e\left(\left|\frac{\epsilon}{2\sqrt{2N}}\sum_{s\in[2N]}e_s-\sum_{s\in[2N]}e_s'\right|^2\right)\tag{54}$$
$$=E_f(|\mathrm{QR}_f(l,k)-\mathrm{QR}_f(l+1,k)|^2)+\epsilon^2\sigma^2/2\tag{55}$$

where e_s and $e'_s \sim V(0, \sigma^2)$ are both the short hands of IID noises e_{ij} in (53). Equation (54) is by combining (53) and the fact that the mean of random variables e_s, e'_s is 0. Similarly

$$E_{f,e}(|\mathrm{QR}_h(l,k) + \mathrm{QR}_h(l+1,k)|^2) = E_f(|\mathrm{QR}_f(l,k) + \mathrm{QR}_f(l+1,k)|^2) + \epsilon^2\sigma^2/2. \quad (56)$$

Now, we denote

$$\Delta_+ = \textstyle\sum_{l,k\in[N]} |\mathrm{QR}_h(2l, 2k+1) + \mathrm{QR}_h(2l+1, 2k+1)|^2$$
$$\Delta_- = \textstyle\sum_{l,k\in[N]} |\mathrm{QR}_h(2l, 2k+1) - \mathrm{QR}_h(2l+1, 2k+1)|^2$$
$$\Delta'_+ = \textstyle\sum_{l,k\in[N]} |\mathrm{QR}_f(2l, 2k+1) + \mathrm{QR}_f(2l+1, 2k+1)|^2$$
$$\Delta'_- = \textstyle\sum_{l,k\in[N]} |\mathrm{QR}_f(2l, 2k+1) - \mathrm{QR}_f(2l+1, 2k+1)|^2.$$

From (52), our aim is to lower bound the following kind of average probability of success:

$$P_{\text{success}} = \frac{E_{f,e}(\Delta_+)}{E_{f,e}(\Delta_+ + \Delta_-)} = (55), (56) \frac{E_f(\Delta'_+) + N^2\epsilon^2\sigma^2/2}{E_f(2\sum_{i,j\in[N]} |f(i,j)|^2) + N^2\epsilon^2\sigma^2} \quad (57)$$

where the denominator in (57) is by the following relations:

$$\sum_{l,k\in[N]} \Delta'_+ + \Delta'_- = \sum_{l,k\in[2N]} 2|\mathrm{QR}_f(l,k)|^2 = 2\sum_{i,j\in[N]} |f(i,j)|^2. \quad (58)$$

Notice that if k is even, then $\mathrm{QR}_f(l,k) \equiv 0$. We first consider estimating $E_f(|\mathrm{QR}_f(l,k) + \mathrm{QR}_f(l+1,k)|^2)$ and $E_f(|\mathrm{QR}_f(l,k) - \mathrm{QR}_f(l+1,k)|^2)$ for any odd $k \in [2N]$. Denote intervals

$$P_1 := [0,N)\times[0,N) \bigcup [N,2N)\times[N,2N)$$
$$P_2 := [0,N)\times[N,2N) \bigcup [N,2N)\times[0,N) \quad (59)$$

and denote the number of points in $L^{2N}_{l,k} \bigcap P_1$ and $L^{2N}_{l,k} \bigcap P_2$ by

$$C_1(L^{2N}_{l,k}) = \mathrm{card}(\{(x,y) \in L^{2N}_{l,k} | (x,y) \in P_1\})$$
$$C_2(L^{2N}_{l,k}) = \mathrm{card}(\{(x,y) \in L^{2N}_{l,k} | (x,y) \in P_2\}) \quad (60)$$

respectively. By geometry, it can be verified that

$$C_1(L^{2N}_{l,k}) + C_2(L^{2N}_{l,k}) = 2N \quad \forall l,k \in [2N] \quad (61)$$
$$|C_1(L^{2N}_{l,k}) - C_1(L^{2N}_{l+1,k})| = 2 \quad \forall \text{ odd } k \in [2N]. \quad (62)$$

By (26), $\mathrm{QR}_f(l,k)$ is a sum of values $\tilde{f}(i,j)$ over line $L^{2N}_{l,k}$ on $\mathbb{Z}^2_{2N}$, where each $\tilde{f}(i,j)$ takes values from uniform distribution on [0,1] for $(i,j) \in P_1$, and each $\tilde{f}(i,j)$ takes values from uniform distribution on $[-1,0]$ for $(i,j) \in P_2$. Combining (61) and (62) gives that for any odd k

$$E_f\left(|\mathrm{QR}_f(l,k) - \mathrm{QR}_f(l+1,k)|^2\right) = \frac{1}{8N}E\left(\left|\sum_{C_1(l)+C_2(l+1)} \nu - \sum_{C_2(l)+C_1(l+1)} \nu\right|^2\right) = \frac{1}{8N}E\left(\left|\sum_{2N+2} \nu - \sum_{2N-2} \nu\right|^2\right) \quad (63)$$

where $C_1(l)$ is the short hand of $C_1(L^{2N}_{l,k})$, and $\sum_k \nu$ denotes the summation of k numbers of independent random variables $\nu \sim \mathbb{U}[0,1]$. Notice that for IID random variables $\nu_1, \nu_2 \sim \mathbb{U}[0,1]$, we have $E(|\nu_1|^2) = |E(\nu_1)|^2 + D(\nu_1)$, where D is the variance, $D(\nu_1 \pm \nu_2) = 2D(\nu_1)$ and $E(\nu_1 \pm \nu_2) = E(\nu_1) \pm E(\nu_1)$. So

$$E\left(\left|\sum_{2N+2} \nu - \sum_{2N-2} \nu\right|^2\right) = D\left(\sum_{2N+2} \nu - \sum_{2N-2} \nu\right) + E\left(\sum_{2N+2} \nu - \sum_{2N-2} \nu\right) = \frac{N}{3} + 2. \quad (64)$$

Combining (64) and (63) gives

$$E_f\left(\sum_{l\in[N],k\in[N]} |\mathrm{QR}_f(2l,2k+1) - \mathrm{QR}_f(2l+1,2k+1)|^2\right) = N^2 \times \frac{1}{8N} \times \left(\frac{N}{3} + 2\right) = \frac{N^2}{24} + \frac{N}{4}. \quad (65)$$

Then by (57), (58), (64), and $E(\sum_{i,j\in[N]} |f(i,j)|^2) = \frac{N^2}{3}$, the lower bound is achieved as follows:

$$P_{\text{success}} = \frac{E(2|f|^2) - (\frac{N^2}{24} + \frac{N}{4}) + N^2\frac{\epsilon^2\sigma^2}{2}}{E(2|f|^2) + N^2\epsilon^2\sigma^2} = 1 - \frac{\frac{1}{24} + \frac{1}{4N} + \frac{\epsilon^2\sigma^2}{2}}{\frac{2}{3} + \epsilon^2\sigma^2} > 1/2. \quad (66)$$

■

Remark: For a real-world image, due to the smoothness of images, the ratio of wavelet coefficients to filter coefficients is likely to be very small, so that the probability of success of quantum denoising method is likely to be high, such as 95%, as shown in Fig. 8.

By treating the probability of success as a constant, we conclude that the QPRT-based quantum image denoising method has a time complexity $O(\log^3 n)$, which is exponentially faster than the classical PDRT denoising method.

Also, there are more quantum techniques that allow to simulate denoising using other Daubechies wavelets [24] and thresholds [36] in the quantum computation framework.

B. LINE DETECTION USING SIDRT

The SIDRT enables one to detect possible line in images: when the SIDRT $P_k(l)$ of image f reaches maximum for some pair l and k, there is likely to be a straight line with gradient k and intercept l in the image. For example, suppose that

f depicts a line composed of points with grayscale 1, while the grayscales of all other points in the picture are 0. Then, the SIDRT reaches maximum at the corresponding line, with the coordinates of the maximum in the Radon domain giving the slope and intercept of the line, respectively. Fig. 9 shows a powerful line detection capability possessed by SIDRT.

In classical case, performing SIDRT has a running time $\Omega(N^3)$, and finding the maximum of SIDRT has a time complexity $O(N^2)$. So, the total running time of classical SIDRT-based line detection algorithm is $\Omega(N^3)$; in contrast, there is a quantum algorithm that can run in $O(N^{1.5})$ time to execute line detections (in the best case) by the following proposition.

Proposition 9: Let f be an $N \times N$ quantum image that can be prepared in time $O(T_{\text{in}})$ by the unitary $U : |0\rangle \to |\vec{f}\rangle$. There exist a quantum algorithm that can detect line in f by using SIDRT in time $O(\frac{N^{1.5}T_{\text{in}}}{\epsilon} polylog\ N)$, where ϵ is the desired precision. This algorithm outputs a pair (θ_0, l_0) such that $\tilde{P}_{k_{\theta_0}}(l_0) \geq (1 - \frac{2}{\sqrt{3}}\epsilon) \max_{\theta,l} P_{k_\theta}(l)$.

Proof: Applying Theorem 3 to the input state $\sum_{\theta,l\in[N]} \frac{1}{N}|\theta\rangle|l\rangle|0\rangle$ with setting precision $\frac{\epsilon}{N^{0.5}}$, one can prepare the following state in time $O(\frac{N^{0.5}T_{\text{in}}\text{polylog N}}{\epsilon})$:

$$\sum_{\theta,l\in[N]} \frac{1}{N}|\theta\rangle|l\rangle|\tilde{P}_{k_\theta}(l)\rangle \tag{67}$$

where $|\tilde{P}_{k_\theta}(l) - P_{k_\theta}(l)| \leq \frac{\epsilon}{N^{0.5}}$. Then, by the quantum algorithm for finding the maximum among N terms with query complexity $O(\sqrt{N})$ [48], one can find (l_0, θ_0) such that $\tilde{P}_{k_{\theta_0}}(l_0) = \max_{\theta,l\in[N]} \tilde{P}_{k_\theta}(l)$ in time $O(\frac{N^{1.5}T_{\text{in}}\text{polylog N}}{\epsilon})$.

By Proposition 7, in the average case, it holds that $\max_{\theta,l} P_{k_\theta}(l) \geq \frac{\sqrt{3}}{2N^{0.5}}$. Now that

$$|\tilde{P}_{k_\theta}(l) - P_{k_\theta}(l)| \leq \frac{\epsilon}{N^{0.5}} \leq \frac{2\epsilon \max P_{k_\theta}(l)}{\sqrt{3}} \tag{68}$$

it holds that

$$\tilde{P}_{k_{\theta_0}}(l_0) = \max \tilde{P}_{k_\theta}(l) \geq \left(1 - \frac{2\epsilon}{\sqrt{3}}\right) \max P_{k_\theta}(l). \tag{69}$$

VII. CONCLUSION

This article presents a novel DRT for efficient quantum implementation. By theoretical analysis and numerical experiments, it is shown that our new proposed QPRT has similar functionality to the classical PDRT, and is more suitable for quantum implementation than PDRT. In addition to its quantum advantage, the QPRT has a classical application value that it can be used as a "reversible PDRT."

Also, a polynomially fast quantum implementation of another interpolation-based kind of DRT is given. There are two problems that deserve further investigation, which are as follows.

1) Is it possible to provide exponential speedup for performing interpolation-based DRTs or line detection?
2) Considering that the computational basis state of the maximum amplitude is most likely to be observed by quantum measurement, can this property be used to design a better line detection algorithm?

APPENDIX
SKETCH PROOF OF PROPOSITION 5

Proposition 5 (see [41, Th. 1]) is a parallel version of the following lemma, which allows one to estimate the inner product of two quantum states, where a key quantum subroutine used is called phase estimation (cf., [17, Sec. 5.2]).

Lemma 4 (Swap Test [49]): Let $|\vec{x}\rangle, |\vec{y}\rangle$ be two quantum states, which can be prepared in time $O(T_{\text{in}})$. There is a quantum algorithm with runtime $O(\frac{T_{\text{in}}}{\epsilon})$ to achieve $|0\rangle \to |s\rangle$, where $|s - \langle\vec{x}|\vec{y}\rangle| \leq \epsilon$.

Proof: First consider two arbitrary normalized quantum states $|\vec{u}\rangle, |\vec{v}\rangle$. Suppose that the state

$$|\vec{\phi}\rangle = \sin\theta|0\rangle|\vec{u}\rangle + \cos\theta|1\rangle|\vec{v}\rangle, \text{where}\vec{\theta} \in (0, \frac{\pi}{2}] \tag{70}$$

can be prepared in time $O(T)$ by the unitary $U : |0\rangle \to |\vec{\phi}\rangle$. Let $G = U(2|0\rangle\langle 0| - \mathbb{I})U^\dagger(Z \otimes \mathbb{I})$, where the matrix $Z := \begin{bmatrix} 1 & \\ & -1 \end{bmatrix}$ acts on the first qubit, and $\mathbb{I}$ is the identity matrix. Then, under the basis $\{|0\rangle|\vec{u}\rangle, |1\rangle|\vec{v}\rangle\}$, one can write

$$G = \begin{pmatrix} \cos 2\theta & \sin 2\theta \\ -\sin 2\theta & \cos 2\theta \end{pmatrix}.$$

The eigenvalues of G are $e^{\pm I2\theta}$ with the corresponding eigenvectors $|\vec{w}_\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle|\vec{u}\rangle \pm I|1\rangle|\vec{v}\rangle)$. By (70), $|\vec{\phi}\rangle = -\frac{I}{\sqrt{2}}(e^{I\theta}|\vec{w}_+\rangle - e^{-I\theta}|\vec{w}_-\rangle)$. So, after performing the quantum phase estimation associated with operator G and initial state $|0\rangle^n|\vec{\phi}\rangle$ where $n = \Theta(\log\frac{1}{\epsilon})$, the result is (an good approximation to) the following state:

$$\frac{-I}{\sqrt{2}}\left[e^{I\theta}|h\rangle|\vec{w}_+\rangle - e^{-I\theta}|-h\rangle|\vec{w}_-\rangle\right] \tag{71}$$

where $h \in [2^n]$ satisfies $|h\pi/2^n - \theta| \leq \epsilon$. The time complexity for the previous procedure is $O(T/\epsilon)$. By adding ancilla qubits, one can continue to compute abs(h), where abs$(h) = h$ is the absolutely function in $[2^n]$ (i.e., if $h \in [2^{n-1}]$, then abs$(h) = h$; otherwise, abs$(h) = 2^n - h$. It always holds that abs$(h) =$ abs$(-h \mod 2^n)$). It can be implemented in time $O(T/\epsilon)$ that

$$\begin{aligned} &|0\rangle \to |0\rangle^n|\phi\rangle|0\rangle^{n-1} \\ &\to \frac{-I}{\sqrt{2}}\left[e^{I\theta}|h\rangle|\vec{w}_+\rangle - e^{-I\theta}|-h\rangle|\vec{w}_-\rangle\right]|\text{abs}(h)\rangle \\ &\xrightarrow{\text{undo}} |\text{abs}(h)\rangle \end{aligned} \tag{72}$$

where abs(h) satisfies $|\sin(\text{abs}(h)\pi/2^n) - \sin\theta| \leq \epsilon$.

Now, we proof this lemma in the case where $|\vec{x}\rangle$ and $|\vec{y}\rangle$ are two real quantum states that can be prepared in time T_{in} by unitaries: $|0\rangle \to |\vec{x}\rangle$ and $|0\rangle \to |\vec{y}\rangle$. The previous method, indeed, provides a quantum algorithm to estimate the inner product $\langle\vec{x}|\vec{y}\rangle$ to precision ϵ in time $O(\frac{T_{\text{in}}}{\epsilon})$. In fact, setting in

(70) the following:

$$|\vec{\phi}\rangle = \frac{1}{2}(|0\rangle(|\vec{x}\rangle + |\vec{y}\rangle) + |1\rangle(|\vec{x}\rangle - |\vec{y}\rangle)) \tag{73}$$

(normalized form of) $|\vec{u}\rangle = |\vec{x}\rangle + |\vec{y}\rangle$, (normalized form of) $|\vec{v}\rangle = |\vec{x}\rangle - |\vec{y}\rangle$, then by calculation, $\sin\theta = \sqrt{(1 + \langle\vec{x}|\vec{y}\rangle)/2}$. Using the unitary capable of preparing $|x\rangle$ and $|y\rangle$ in time $O(T_{\text{in}})$, one can prepare the state of (73) in time $O(T_{\text{in}})$. Then by applying (72) and trigonometric function "sin" [37], one can realize in time $O(\frac{T_{\text{in}}}{\epsilon})$ the following mapping:

$$|0\rangle \to |\widetilde{\sin\theta}\rangle \to |s\rangle$$

where $|\widetilde{\sin\theta} - \sin\theta| \leq \epsilon$ and $|s - \langle\vec{x}|\vec{y}\rangle| \leq \epsilon$.

For the case when $|\vec{x}\rangle$ and $|\vec{y}\rangle$ in (73) are complex vectors, the corresponding $\sin\theta = \sqrt{(1 + (Re\langle\vec{x}|\vec{y}\rangle)/2}$, so that one can compute $\text{Re}\langle x|y\rangle$ using the previous method. The imaginary part of $\langle\vec{x}|\vec{y}\rangle$ can be obtained by computing $\text{Re}\langle\vec{x}|I\vec{y}\rangle$. ■

In Proposition 5, one is given $2N$ quantum states $|\vec{u}_0\rangle$, $|\vec{v}_0\rangle$... $|\vec{u}_{N-1}\rangle$, $|\vec{v}_{N-1}\rangle$, and the preparation unitaries as follows:

$$U = \begin{cases} |k\rangle|0\rangle \to |k\rangle|\vec{u}_k\rangle, & k \in [N] \\ |k\rangle|0\rangle \to |k\rangle|\vec{v}_{k-N}\rangle, & k \in [2N] \setminus [N]. \end{cases}$$

To perform swap tests in parallel, consider the operator $G' = U(\mathbb{I}_{\log 2N \times \log 2N} \otimes (2|0\rangle\langle 0| - \mathbb{I}))U^{-1}$. Proposition 5 follows by applying the phase estimation associated with the operator $G'(Z \otimes \mathbb{I})$ and state $\frac{1}{\sqrt{2N}}\sum_{j\in[2N]}|j\rangle|0\rangle$, where Z acts on the most significant qubit in $|k\rangle$. More details can be found in [41].

ACKNOWLEDGMENT

The authors would like to thank referees for useful comments and constructive suggestions.

REFERENCES

[1] J. Radon, "Über die bestimmung von funktionen durch ihre integralwerte längs gewisser mannigfaltigkeiten," *J. Math. Phys.*, vol. 69, pp. 262–277, 1917, doi: 10.1090/psapm/027/692055.

[2] S. R. Deans, *The Radon Transform and Some of Its Applications*. Mineola, NY, USA: Dover Pub., 2007.

[3] M. N. Do and M. Vetterli, "Image denoising using orthonormal finite ridgelet transform," in *Proc. Wavelet Appl. Signal Image Process. VIII*, 2000, vol. 4119, pp. 831–842, doi: 10.1117/12.408673.

[4] G. Beylkin, "Discrete Radon transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, no. 2, pp. 162–172, Feb. 1987, doi: 10.1109/TASSP.1987.1165108.

[5] P. A. Toft, "The Radon transform-theory and implementation," Ph.D. dissertation, Dept. Math. Model., Sect. Digit. Signal Process., Tech. Univ. Denmark, Kongens Lyngby, Denmark, 1996.

[6] B. T. Kelley and V. K. Madisetti, "The fast discrete Radon transform. I. Theory," *IEEE Trans. Image Process.*, vol. 2, no. 3, pp. 382–400, Jul. 1993, doi: 10.1109/83.236530.

[7] D. P. Lun *et al.*, "Discrete periodic Radon transform based weighted nuclear norm minimization for image denoising," in *Proc. 5th Int. Symp. Comput. Netw.*, 2017, pp. 395–400, doi: 10.1109/CANDAR.2017.88.

[8] C. Carranza, M. Pattichis, and D. Llamocca, "Fast and parallel computation of the discrete periodic Radon transform on GPUs, multicore CPUs and FPGAs," in *Proc. 25th IEEE Int. Conf. Image Process.*, 2018, pp. 4158–4162, doi: 10.1109/ICIP.2018.8451751.

[9] F. Matus and J. Flusser, "Image representation via a finite Radon transform," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 10, pp. 996–1006, Oct. 1993, doi: 10.1109/34.254058.

[10] T. Hsung, D. P. Lun, and W. Siu, "The discrete periodic Radon transform," *IEEE Trans. Signal Process.*, vol. 44, no. 10, pp. 2651–2657, Oct. 1996, doi: 10.1109/78.539055.

[11] I. Svalbe and D. van der Spek, "Reconstruction of tomographic images using analog projections and the digital Radon transform," *Linear Algebra Appl.*, vol. 339, no. 1-3, pp. 125–145, 2001, doi: 10.1016/S0024-3795(01)00487-6.

[12] A. Kingston *et al.*, "Projective transforms on periodic discrete image arrays," *Adv. Imag. Electron. Phys.*, vol. 139, pp. 75–177, 2006, doi: 10.1016/S1076-5670(05)39002-1.

[13] T. M. Khanipov, "Computational complexity lower bounds of certain discrete Radon transform approximations," 2018, *arXiv:1801.01054*.

[14] C. Shao, Y. Li, and H. Li, "Quantum algorithm design: Techniques and applications," *J. Syst. Sci. Complexity*, vol. 32, no. 1, pp. 375–452, 2019, doi: 10.1007/s11424-019-9008-0.

[15] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Rev.*, vol. 41, no. 2, pp. 303–332, 1999, doi: 10.1137/S0097539795293172.

[16] C. Zalka, "Grover's quantum searching algorithm is optimal," *Phys. Rev. A*, vol. 60, no. 4, 1999, Art. no. 2746, doi: 10.1103/PhysRevA.60.2746.

[17] M. E. Nielsen, M. A. Nielsen, and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[18] N. Abura'ed, F. S. Khan, and H. Bhaskar, "Advances in the quantum theoretical approach to image processing applications," *ACM Comput. Surv.*, vol. 49, no. 4, pp. 1–49, 2017, doi: 10.1145/3009965.

[19] G. Beach, C. Lomont, and C. Cohen, "Quantum image processing (QuIP)," in *Proc. 32nd Appl. Imagery Pattern Recognit. Workshop*, 2003, pp. 39–44, doi: 10.1109/AIPR.2003.1284246.

[20] F. Yan, A. M. Iliyasu, and S. E. Venegas-Andraca, "A survey of quantum image representations," *Quantum Inf. Process.*, vol. 15, no. 1, pp. 1–35, 2016, doi: 10.1007/s11128-015-1195-6.

[21] S. E. Venegas-Andraca and S. Bose, "Storing, processing, and retrieving an image using quantum mechanics," in *Proc. Quantum Inf. Comput. Int. Soc. Opt. Photon.*, 2003, pp. 137–147, doi: 10.1117/12.485960.

[22] J. I. Latorre, "Image compression and entanglement," 2005, *arXiv:quant-ph/0510031*.

[23] Y. Zhang, K. Lu, Y. Gao, and M. Wang, "NEQR: A novel enhanced quantum representation of digital images," *Quantum Inf. Process.*, vol. 12, no. 8, pp. 2833–2860, 2013, doi: 10.1007/s11128-013-0567-z.

[24] A. Fijany and C. P. Williams, "Quantum wavelet transforms: Fast algorithms and complete circuits," in *Proc. NASA Int. Conf. Quantum Comput. Quantum Commun.*, 1998, pp. 10–33, doi: 10.1007/3-540-49208-9_2.

[25] Y. Zhang, K. Lu, and Y. Gao, "QSobel: A novel quantum image edge extraction algorithm," *Sci. China Inf. Sci.*, vol. 58, no. 1, pp. 1–13, 2015, doi: 10.1007/s11432-014-5158-9.

[26] X.-W. Yao *et al.*, "Quantum image processing and its application to edge detection: Theory and experiment," *Phys. Rev. X*, vol. 7, no. 3, 2017, Art. no. 0 31041, doi: 10.1103/PhysRevX.7.031041.

[27] S. Caraiman and V. I. Manta, "Histogram-based segmentation of quantum images," *Theor. Comput. Sci.*, vol. 529, pp. 46–60, 2014, doi: 10.1016/j.tcs.2013.08.005.

[28] X. H. Song, S. Wang, S. Liu, A. A. A. El Latif, and X. M. Niu, "A dynamic watermarking scheme for quantum images using quantum wavelet transform," *Quantum Inf. Process.*, vol. 12, no. 12, pp. 3689–3706, 2013, doi: 10.1007/s11128-013-0629-2.

[29] H. S. Li, X. Chen, S. Song, Z. Liao, and J. Fang, "A block-based quantum image scrambling for GNEQR," *IEEE Access*, vol. 7, pp. 233–243, 2019, doi: 10.1109/ACCESS.2019.2942986.

[30] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum algorithm for linear systems of equations," *Phys. Rev. Lett.*, vol. 103, no. 15, p. 150502, 2009, doi: 10.1103/PhysRevLett.103.150502.

[31] H. J. Nussbaumer, "The fast Fourier transform," in *Fast Fourier Transform and Convolution Algorithms*. Berlin, Germany: Springer, 1981, pp. 80–111, doi: 10.1007/978-3-662-00551-4_4.

[32] A. Kingston and I. Svalbe, "Generalised finite Radon transform for n× n images," *Image Vis. Comput.*, vol. 25, no. 10, pp. 1620–1630, 2007, doi: 10.1016/j.imavis.2006.03.002.

[33] A. Kingston, "Orthogonal discrete Radon transform over $p^n \times p^n$ images," *Signal Process.*, vol. 86, no. 8, pp. 2040–2050, 2006, doi: 10.1016/j.sigpro.2005.09.024.

[34] A. Buades, B. Coll, and J.-M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 490–530, 2005, doi: 10.1137/040616024.

[35] D. L. Donoho and J. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, 1994, doi: 10.1093/biomet/81.3.425.

[36] I. Kerenidis and A. Prakash, "Quantum recommendation systems," in *Proc. 8th Innov. Theor. Comput. Sci. Conf.*, 2017, pp. 49:1–49:21, doi: 10.4230/LIPIcs.ITCS.2017.49.

[37] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, "Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics," in *Proc. 51st Annu. ACM SIGACT Symp. Theory Comput.*, 2019, pp. 193–204, doi: 10.1145/3313276.3316366.

[38] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, 2017, doi: 10.1038/nature23474.

[39] B. D. Clader, B. C. Jacobs, and C. R. Sprouse, "Preconditioned quantum linear system algorithm," *Phys. Rev. Lett.*, vol. 110, no. 25, 2013, Art. no. 250504, doi: 10.1103/PhysRevLett.110.250504.

[40] C. Shao, "From linear combination of quantum states to Grover's searching algorithm," 2018, *arXiv:1807.09693*.

[41] C. Shao, "A quantum model for multilayer perceptron," 2018, *arXiv:1808.10561*.

[42] L. Ruiz Perez and J. C. Garcia Escartin, "Quantum arithmetic with the quantum Fourier transform," *Quantum Inf. Process.*, vol. 16, no. 6, 2017, Art. no. 152, doi: 10.1007/s11128-017-1603-1.

[43] C. Wang and L. Wossnig, "A quantum algorithm for simulating non-sparse Hamiltonians," 2018, *arXiv:1803.08273*.

[44] A. Averbuch, R. Coifman, D. Donoho, M. Israeli, and J. Walden, "Fast slant stack: A notion of Radon transform for data in a cartesian grid which is rapidly computible, algebraically exact, geometrically faithful and invertible," *SIAM Sci. Comput.*, vol. 373, no. 3, pp. 192–206, 2001.

[45] D. L. Donoho and A. G. Flesia, "Digital ridgelet transform based on true ridge functions," in *Studies in Computational Mathematics*. Amsterdam, The Netherlands: Elsevier, 2003, pp. 1–30, doi: 10.1016/S1570-579X(03)80029-0.

[46] I. L. Markov and M. Saeedi, "Constant-optimized quantum circuits for modular multiplication and exponentiation," 2012, *arXiv:1202.6614*.

[47] S. Catterall, "Central limit theorem for square roots of sums of i.i.d. random variables." Accessed: Jan. 08, 2020. [Online]. Available: https://stats.stackexchange.com/questions/241504/central-limit-theorem-for-square-roots-of-sums-of-i-i-d-random-variables

[48] A. Ahuja and S. Kapoor, "A quantum algorithm for finding the maximum," 1999, *arXiv:quant-ph/9911082*.

[49] H. Buhrman, R. Cleve, J. Watrous, and R. De Wolf, "Quantum fingerprinting," *Phys. Rev. Lett.*, vol. 87, no. 16, 2001, Art. no. 167902, doi: 10.1103/PhysRevLett.87.167902.