

量子线性卷积及其在图像处理中的应用

刘兴奥^{1,2} 周日贵^{1,2} 郭文字^{1,2}

摘要 线性卷积在图像处理中发挥着重要作用,但是在处理海量高分辨率图像时,求解线性卷积会消耗许多计算资源.为此,本文就量子线性卷积及其在图像处理问题中的应用开展相关研究,首先提出单通道,单位步长,零补充情况下的量子一维和二维线性卷积,然后实现多通道,非单位步长,非零补充的情况,最后将量子二维线性卷积应用于量子图像平滑,量子图像锐化和量子图像边缘检测.通过理论分析证明了量子线性卷积的空间复杂度 $O(\log M)$ 和时间复杂度 $O(\log^2 M)$ 较经典线性卷积有指数级下降,且基于 Qiskit 的仿真实验成功验证了量子线性卷积和量子图像处理算法的正确性和可行性.

关键词 量子线性卷积,量子图像平滑,量子图像锐化,量子图像边缘检测

引用格式 刘兴奥,周日贵,郭文字.量子线性卷积及其在图像处理中的应用.自动化学报,2022,48(6):1504-1519

DOI 10.16383/j.aas.c210637

Quantum Linear Convolution and Its Application in Image Processing

LIU Xing-Ao^{1,2} ZHOU Ri-Gui^{1,2} GUO Wen-Yu^{1,2}

Abstract Linear convolution plays an important role in image processing, but when processing massive high-resolution images, solving linear convolution will consume a lot of computing resources. To this end, this paper conducts related research on quantum linear convolution and its application in image processing problems. We first propose the quantum one-dimensional and two-dimensional linear convolution in the case of single channel, unit stride and zero padding, and then implement the case of multi channel, non-unit stride and non-zero padding. Finally, we apply the quantum two-dimensional linear convolution to quantum image smoothing, quantum image sharpening and quantum image edge detection. Through theoretical analysis, it is proved that the space complexity $O(\log M)$ and time complexity $O(\log^2 M)$ of quantum linear convolution decrease exponentially compared with classical linear convolution, and the simulation experiment based on Qiskit successfully verifies the correctness and feasibility of the quantum linear convolution and quantum image processing algorithms.

Key words Quantum linear convolution, quantum image smoothing, quantum image sharpening, quantum image edge detection

Citation Liu Xing-Ao, Zhou Ri-Gui, Guo Wen-Yu. Quantum linear convolution and its application in image processing. *Acta Automatica Sinica*, 2022, 48(6): 1504-1519

线性卷积是科学与工程上的重要工具,在图像处理中发挥了重要作用,例如基于均值滤波器和高斯滤波器实现图像平滑^[1],基于拉普拉斯算子实现图像锐化^[1],基于索伯梯度算子实现图像边缘检测,使用基于对称式扩充卷积的残差网络进行图像去噪^[2].对于 $M \times M$ 图像和 $N \times N$ 滤波器,其卷积操作的时间复杂度是 $O(M^2 N^2)$.虽然经典计算能求出线性

卷积结果,但是在处理海量高分辨率图像时,求解线性卷积会消耗大量计算资源.量子计算为解决该问题提供了一种新的解决方案.

量子力学的量子叠加和量子纠缠特性使得量子计算在处理某些特定问题上具有显著的速度优势,例如大数因子分解算法 (Shor 算法)^[3],数据库搜索算法 (Grover 算法)^[4],线性方程求解算法 (HHL 算法)^[5],粗糙集核属性求解算法^[6],映射量子模型^[7],孪生支持向量机^[8]和量子主成分分析^[9-10].目前关于量子线性卷积的研究集中在量子一维卷积 (Quantum one-dimensional convolution, QOC) 和量子二维卷积 (Quantum two-dimensional convolution, QTC).Lomont^[11]证明,如果使用振幅编码将两个一维序列制备成两个量子态,那么在允许使用线性算子 (酉算子和测量算子) 和辅助量子比特的情况下,基于量子力学不能计算它们的量子循环卷积

收稿日期 2021-07-08 录用日期 2021-11-02

Manuscript received July 8, 2021; accepted November 2, 2021

上海市科技项目 (20040501500) 资助

Supported by Shanghai Science and Technology Project (20040501500)

本文责任编辑 刘成林

Recommended by Associate Editor LIU Cheng-Lin

1. 上海海事大学信息工程学院 上海 201306 2. 上海海事大学智能信息处理与量子智能计算研究中心 上海 201306

1. College of Information Engineering, Shanghai Maritime University, Shanghai 201306 2. Research Center of Intelligent Information Processing and Quantum Intelligent Computing, Shanghai 201306

$|c\rangle \otimes |g\rangle$, 其中 $|c\rangle$ 表示卷积结果, $|g\rangle$ 表示垃圾项. 闫茜茜等^[12] 提出量子一维窄卷积, 首先使用振幅编码信息, 利用量子态张量积性质获取振幅的乘积, 接着使用置换矩阵实现振幅的置换, 然后使用哈达玛门 (Hadamard, H) 量子门进行加法计算, 最终获得量子态 $|c\rangle \otimes |0\rangle + |g\rangle \otimes |0\rangle^\perp$, $|c\rangle$ 是卷积结果. 随后闫茜茜等^[13] 又提出量子二维窄卷积, 其实现过程同量子一维窄卷积相似. 另外, 量子图像滤波^[14-17] 和量子图像边缘检测^[18-29] 等算法使用量子算术计算线性卷积. 虽然上面三种方案都可以实现量子线性卷积, 但各有不足. 闫茜茜等提出的量子一维和二维线性卷积方法不适用于求解量子线性宽卷积和量子线性等宽卷积, 量子图像滤波和量子图像边缘检测中的量子二维线性卷积算法需要消耗大量的量子比特资源.

本文研究内容包括量子线性卷积的实现和应用两方面. 首先提出单通道, 单位步长, 零补充情况的量子一维宽卷积 (Quantum one-dimensional wide convolution, QOWC), 量子一维等宽卷积 (Quantum one-dimensional equal-width convolution, QOEC), 量子一维窄卷积 (Quantum one-dimensional narrow convolution, QONC), 量子二维宽卷积 (Quantum two-dimensional wide convolution, QTWC), 量子二维等宽卷积 (Quantum two-dimensional equal-width convolution, QTEC), 量子二维窄卷积 (Quantum two-dimensional narrow convolution, QTNC). 然后实现多通道, 非单位步长, 非零补充情况的 QOWC, 同样适用于 QOEC, QONC, QTWC, QTEC, QTNC. 最后基于量子二维线性卷积实现量子图像平滑 (Quantum image smoothing, QISM), 量子图像锐化 (Quantum image sharpening, QISH) 和量子图像边缘检测 (Quantum image edge detection, QIED) 算法, 并在 Qiskit 上进行仿真实验. 理论分析证明了在时间和资源消耗方面量子线性卷积相比于经典线性卷积呈指数下降.

本文后续部分组织结构如下. 第 1 节介绍线性卷积和量子计算的相关知识. 第 2 节实现 QOWC, QOEC, QONC, QTWC, QTEC, QTNC. 第 3 节实现 QTC 在 QISM, QISH 和 QIED 上的应用. 第 4 节总结和展望. 附录 A 给出置换电路的优化方法. 附录 B 给出量子态制备方法.

1 预备知识及符号说明

1.1 线性卷积

为避免因读者研究背景不同导致的对相关术语

理解存在部分歧义的问题, 本文将在此节中对相关学术名词及符号做合理规范, 读者的阅读请务必参考此规范开展. 根据维度 d , 补充 p , 步长 s , 通道数 c 参数的不同, 线性卷积存在多种变体. 本节重点介绍单通道, 零补充, 单位步长的一维, 二维线性卷积. 在单通道, 单位步长, 零补充情况下, 线性卷积根据补零的数量又可以划分成宽线性卷积 (宽卷积), 等宽线性卷积 (等宽卷积) 和窄线性卷积 (窄卷积).

1.1.1 一维线性卷积

令输入序列 $\mathbf{X} = [x_0, x_1, \dots, x_{M-1}]$, 卷积核 $\mathbf{Y} = [y_0, y_1, \dots, y_{N-1}]$, $M > N$. 当 $M < N$, 则将 \mathbf{Y} 作为输入序列, \mathbf{X} 作为卷积核.

一维宽卷积结果是 $\mathbf{Z}_w = [z_0, z_1, \dots, z_{M+N-2}]$, 满足 $z(i) = \sum_{j=0}^{M-1} x_j y_{i-j}$, $i = 0, 1, \dots, M+N-2$. 展开成矩阵形式可以写成 $\mathbf{U}_w \mathbf{X}^T = \mathbf{Z}_w^T$,

$$\begin{bmatrix} y_0 & 0 & \dots & 0 \\ y_1 & y_0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ y_{M-1} & y_{M-2} & \dots & y_0 \\ \vdots & \vdots & \ddots & y_0 \\ 0 & 0 & 0 & y_{N-1} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{M-1} \end{bmatrix} = \begin{bmatrix} z_0 \\ z_1 \\ \vdots \\ z_{M-1} \\ \vdots \\ z_{M+N-2} \end{bmatrix} \quad (1)$$

其中 $\mathbf{U}_w \in \mathbf{R}^{(M+N-1) \times M}$.

一维等宽卷积结果是 $\mathbf{Z}_e = [z_0, z_1, \dots, z_{M-1}]$, 满足 $z(i) = \sum_{j=0}^{M-1} x_j y_{-j+i+(N-1)/2}$, $i = 0, 1, \dots, M-1$. 展开成矩阵形式可以写成 $\mathbf{U}_e \mathbf{X}^T = \mathbf{Z}_e^T$.

一维窄卷积结果是 $\mathbf{Z}_n = [z_0, z_1, \dots, z_{M-N}]$, 满足 $z(i) = \sum_{j=0}^{M-1} x_j y_{-j+i+N-1}$, $i = 0, 1, \dots, M-N$. 展开成矩阵形式可以写成 $\mathbf{U}_n \mathbf{X}^T = \mathbf{Z}_n^T$.

1.1.2 二维线性卷积

二维卷积是一维卷积的扩展, 可以看成是在行和列上分别做一维卷积. 在不影响上下文理解的情况下, 我们使用与一维卷积相同的变量表示物理含义. 令输入序列 $\mathbf{X} = [x_{i_1, i_2}]_{M \times M}$, 卷积核 $\mathbf{Y} = [y_{j_1, j_2}]_{N \times N}$, $M > N$. 当 $M < N$ 时, 将 \mathbf{Y} 作为输入序列, \mathbf{X} 作为卷积核即可.

二维宽卷积结果 $\mathbf{Z}_w = [z_{k_1, k_2}]_{(M+N-1) \times (M+N-1)}$ 满足 $z_{k_1, k_2} = \sum_{i_1=0}^{M-1} \sum_{i_2=0}^{M-1} x_{i_1, i_2} y_{k_1-i_1, k_2-i_2}$, 展开成矩阵形式是 $\mathbf{U}_w \mathbf{X} = \mathbf{Z}_w$, \mathbf{X} 和 \mathbf{Z}_w 分别是 \mathbf{X} 和 \mathbf{Z}_w 的行向量化 (矩阵按行展开成向量形式), 即

$$\begin{bmatrix} y_{0,0} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & y_{N-1,N-1} \end{bmatrix} \begin{bmatrix} x_{0,0} \\ x_{0,1} \\ \vdots \\ x_{M-1,M-1} \end{bmatrix} = \begin{bmatrix} z_{0,0} \\ \vdots \\ z_{0,M+N-2} \\ \vdots \\ z_{M+N-2,0} \\ \vdots \\ z_{M+N-2,M+N-2} \end{bmatrix} \quad (2)$$

二维等宽卷积结果是 $\mathbf{Z}_e = [z_{k_1, k_2}]_{M \times M}$ 满足 $z_{k_1, k_2} = \sum_{i_1=0}^{M-1} \sum_{i_2=0}^{M-1} x_{i_1, i_2} y_{k_1-i_1+(N-1)/2, k_2-i_2+(N-1)/2}$, 展开成矩阵形式是 $\mathbf{U}_e \mathbf{X} = \mathbf{Z}_e$.

二维窄卷积结果是 $\mathbf{Z}_n = [z_{k_1, k_2}]_{(M-N+1) \times (M-N+1)}$ 满足 $z_{k_1, k_2} = \sum_{i_1=0}^{M-1} \sum_{i_2=0}^{M-1} x_{i_1, i_2} y_{k_1-i_1+N-1, k_2-i_2+N-1}$, 展开成矩阵形式是 $\mathbf{U}_n \mathbf{X} = \mathbf{Z}_n$.

1.2 量子计算

量子计算模型有很多, 例如量子电路模型, 量子绝热计算, 量子图灵机, 拓扑量子计算等^[30], 这些量子计算模型是等价的. 本文采用量子电路模型, 其模型易理解且接受范围最广. 量子比特用于存储信息, 它除了可以表示 $|0\rangle$, $|1\rangle$, 还可以是 $|0\rangle$ 和 $|1\rangle$ 的叠加态. 常用的基础门有单量子比特门 X, Y, Z, H, S, T 和双量子比特门受控非门 (Controlled-not gate, CNOT). 这些基础门之间存在等价性关系, 例如 $XZX = -Z$. 任意一种酉操作能以任意精度分解成一系列基础门. 利用这些等价关系和分解方法可以简化复杂的量子电路^[31]. 量子测量是不可逆的过程, 它会引发量子态坍缩, 通过量子态层析, 量子过程层析, 量子压缩感知等技术可以重构出量子态和量子操作的相关信息^[31-32]. 本文使用时间复杂度和空间复杂度来衡量量子电路的性能. 下面给出几种后续行文必需的量子操作:

引理 1. (量子随机存取存储器 (Quantum random access memory, QRAM)^[33-36]) 量子随机存取

存储器可以以相干量子叠加形式执行内存访问. 如果量子计算机需要访问一个叠加的存储单元, 地址寄存器 a 必须包含一个地址的叠加 $\sum_j \psi_j |j\rangle_a |0\rangle$, 量子随机存取存储器将在 $O(\log(MN))$ 时间内, 返回数据寄存器 d 中与地址寄存器相关的数据的叠加 $\sum_j \psi_j |j\rangle_a |D_j\rangle_d$.

引理 2. (量子振幅放大 (Quantum amplitude amplification, QAA)^[37]) 设 U 是酉操作, a 是期望量子态的振幅. 令 $a > 0$, $m = \lfloor \pi/4\theta_a \rfloor$, 其中 θ_a 满足 $\sin^2(\theta_a) = a$, $0 < \theta_a \leq \pi/2$. 如果在量子系统中执行酉操作 $Q^m U |0\rangle$ 并测量它, 那么测量结果是期望量子态的概率至少是 $\max(1-a, a)$.

引理 3. (实数量子模拟数字转换器 (Real quantum analog-to-digital conversion, Real QADC)^[38]) 设 \tilde{r}_j 是一串 m 比特数字 $\tilde{r}_j^{(1)}, \dots, \tilde{r}_j^{(m)}$, 写成 $\sum_{k=1}^m \tilde{r}_j^{(k)} 2^{-k}$, 它近似于 c_j 的实数部分. 一个 m 比特实数量子模拟数字转换器可以将模拟量子态 $\sum_{j=1}^N c_j |j\rangle |0\rangle^{\otimes m}$ 转化成数字量子态 $\frac{1}{\sqrt{N}} \sum_{j=1}^N |j\rangle |\tilde{r}_j\rangle$. 该转换器需要使用 $O(1/\varepsilon)$ 受控 U_A 操作和 $O((\log^2 N)/\varepsilon)$ 个单、双量子比特门, 并以 $1 - O(\text{poly}(\varepsilon))$ 的保真度输出结果, 其中 ε 表示输出精度.

2 量子线性卷积算法

本节研究单通道, 单位步长, 零补充情况下的量子一维和二维线性卷积, 包括宽卷积, 等宽卷积和窄卷积. 第 2.1 节实现了量子一维宽线性卷积, 分析了它的时间复杂度和空间复杂度, 并将其扩展到量子一维等宽卷积, 量子一维窄卷积. 第 2.2 节实现了量子二维宽线性卷积, 分析了它的时间复杂度和空间复杂度, 并将其扩展到量子二维等宽卷积, 量子二维窄卷积. 我们在第 2.3 节讨论了多通道, 非单位步长, 非零补充的情况.

2.1 量子一维线性卷积

如式 (1) 所示, 一维宽卷积计算过程可以写成矩阵形式 $\mathbf{U}_w \mathbf{X}^T = \mathbf{Z}_w^T$, 其中 $\mathbf{U}_w \in \mathbf{R}^{(M+N-1) \times M}$ 是一般矩阵. 为了能使量子计算加速求解计算过程, 需要对输入序列和卷积核变换. 在行向量 \mathbf{X} 的前端补 $N-1$ 个零, 后端补 $N-1+L_t-L_w$ 个零构造出行向量 \mathbf{X}_w :

$$\mathbf{X}_w = \left[\underbrace{0, \dots, 0}_{N-1}, \underbrace{x_0, x_1, \dots, x_{M-1}}_M, \underbrace{0, \dots, 0}_{N-1}, \underbrace{0, \dots, 0}_{L_t-L_w} \right] \quad (3)$$

其中 $L_w = M + 2N - 2$, $L_t = 2^{\lceil \log L_w \rceil}$. 令

$$\mathbf{R}_0 = [y_{N-1}, y_{N-2}, y_{N-3}, \dots, 0, 0] \quad (4)$$

把 \mathbf{R}_0 作为首行向量构造出一个 $L_t \times L_t$ 的循环矩阵 \hat{U}_w :

$$\begin{bmatrix} \overbrace{y_{N-1} \cdots y_1}^{N-1} & \overbrace{y_0 0 \cdots 0}^M & \overbrace{0 \cdots 0}^{N-1} & \overbrace{0 \cdots 0}^{L_t-L_w} \\ \vdots & \vdots & \vdots & \vdots \\ 0 \cdots y_{N-1} & y_{N-2} y_{N-3} \cdots 0 & 0 \cdots 0 & 0 \cdots 0 \\ 0 \cdots 0 & y_{N-1} y_{N-2} \cdots 0 & 0 \cdots 0 & 0 \cdots 0 \\ 0 \cdots 0 & 0 y_{N-1} \cdots 0 & 0 \cdots 0 & 0 \cdots 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 \cdots 0 & 0 0 \cdots y_{N-1} & y_{N-2} \cdots y_0 & 0 \cdots 0 \\ 0 \cdots 0 & 0 0 \cdots 0 & y_{N-1} \cdots y_1 & y_0 \cdots 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 \cdots 0 & 0 0 \cdots 0 & 0 \cdots 0 & y_{N-1} \cdots 0 \\ \vdots & \vdots & \vdots & \vdots \\ y_{N-2} \cdots y_2 & y_1 0 \cdots 0 & 0 \cdots 0 & 0 \cdots y_{N-1} \end{bmatrix} \quad (5)$$

根据循环矩阵的性质, 矩阵 \hat{U}_w 可以简写成:

$$\begin{aligned} \hat{U}_w &= y_{N-1} * \mathbf{I} + y_{N-2} * \mathbf{P}^{L_t-1} + \cdots + y_0 * \mathbf{P}^{L_t-N+1} = \\ & y_{N-1} * \mathbf{I} + y_{N-2} * \mathbf{P}^\dagger + \cdots + y_0 * (\mathbf{P}^\dagger)^{N-1} = \\ & \sum_{i=0}^{N-1} y_{N-1-i} (\mathbf{P}^\dagger)^i \end{aligned} \quad (6)$$

当 $i > N$ 时, $y_{N-1-i} = 0$, $\mathbf{P} \in \mathbf{R}^{L_t \times L_t}$ 是一个置换矩阵, 矩阵形式是:

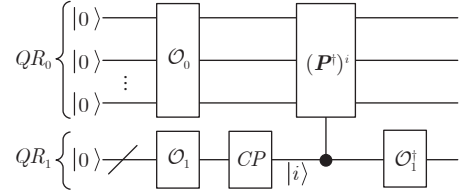
$$\mathbf{P} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \quad (7)$$

\mathbf{P}^\dagger 是它的共轭转置矩阵. 那么一维宽卷积计算过程可以改写成 $\hat{U}_w \mathbf{X}^T = \mathbf{Z}_w^T$, 其中

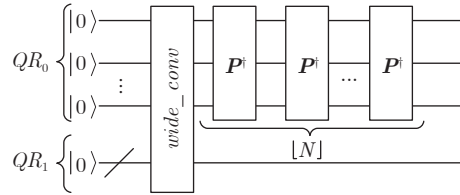
$$\mathbf{Z}_w = \left[\underbrace{z_0, z_1, \cdots, z_{M+N-2}}_{M+N-1}, \underbrace{0, \cdots, 0}_{L_t-M-N+1} \right] \quad (8)$$

行向量 \mathbf{Z}_w 的前 $M+N-1$ 个元素是所求的宽卷积结果, 后 $L_t-M-N+1$ 个元素都是 0, 本文中称之为垃圾元素. 将修改后的输入序列 \mathbf{X}_w 与首行向量 \mathbf{R}_0 存储到量子随机存取存储器中.

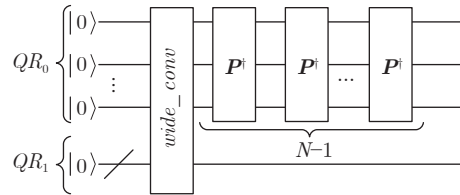
循环矩阵 \hat{U}_w 可以写成置换矩阵的线性组合 (见式 (6)), 基于该性质, 我们使用线性酉组合方法 (Linear combination of unitaries, LCU)^[39] 来求解 $\hat{U}_w \mathbf{X}_w^T = \mathbf{Z}_w^T$. 图 1(a) 给出了量子电路实现过程, 需要注意的是, 图中上面的量子比特是低位,



(a) 量子一维宽卷积实现电路, 其中 $\mathcal{O}_0, \mathcal{O}_1$ 是量子态制备操作, CP 是受控相位操作, \mathbf{P}^i 是置换操作
(a) Quantum circuit of one-dimensional wide convolution, in which \mathcal{O}_0 and \mathcal{O}_1 are the quantum state preparation operation, CP is the controlled phase operation, and \mathbf{P}^i is the permutation operation



(b) 量子一维等宽卷积实现电路, 其中 $wide_conv$ 是量子一维宽卷积实现电路, \mathbf{P}^i 是置换矩阵, N 是卷积核的尺寸
(b) Quantum circuit of one-dimensional equal-width convolution, in which $wide_conv$ is the quantum one-dimensional wide convolution realization circuit, \mathbf{P}^i is the permutation matrix, N is the size of the convolution



(c) 量子一维窄卷积实现电路, 其中 $wide_conv$ 是量子一维宽卷积实现电路, \mathbf{P}^i 是置换矩阵, N 是卷积核的尺寸
(c) Quantum circuit of one-dimensional narrow convolution, in which $wide_conv$ is the quantum one-dimensional wide convolution realization circuit, \mathbf{P}^i is the permutation matrix, N is the size of the convolution kernel

图 1 量子一维线性卷积实现电路

Fig.1 Quantum circuit of one-dimensional linear convolution

下面的量子比特表示高位. 量子算法具体描述如下: 首先初始化两个量子寄存器 $QR_1 \otimes QR_0 = |0\rangle^{\otimes \lceil \log N \rceil} \otimes |0\rangle^{\otimes \log L_t}$. 接着, 令 $\xi_1 = \sum_{i=0}^{N-1} |y_{N-1-i}\rangle$, $\|\mathbf{X}_w\| = \sum_{j=0}^{L_t-1} X_{w,j}^2$, 两个酉操作 \mathcal{O}_0 和 \mathcal{O}_1 分别表示通过量子随机存取存储器将经典数据 \mathbf{X}_w 和 \mathbf{Y} 映射到量子态中:

$$\mathcal{O}_0 : |0\rangle^{\otimes \log L_t} \mapsto |\mathbf{X}_w\rangle = \frac{1}{\sqrt{\|\mathbf{X}_w\|}} \sum_{j=0}^{L_t-1} \mathbf{X}_{w,j} |j\rangle$$

$$\mathcal{O}_1 : |0\rangle^{\otimes \lceil \log N \rceil} \mapsto |\mathbf{R}_0\rangle = \frac{1}{\sqrt{\xi_1}} \sum_{i=0}^{N-1} \sqrt{|y_{N-1-i}|} |i\rangle \quad (9)$$

在 $QR_1 \otimes QR_0$ 上分别执行酉操作 $\mathcal{O}_1 \otimes \mathcal{O}_0$:

$$\frac{1}{\sqrt{\xi_1}} \sum_{i=0}^{N-1} \sqrt{|y_{N-1-i}|} |i\rangle |\mathbf{X}_w\rangle \quad (10)$$

设 $e^{i\theta_i} = y_{N-1-i}/|y_{N-1-i}|$ 表示 \mathbf{R}_0 中每个元素的符号位信息, 在 QR_1 上执行受控相位操作 (CP):

$$\frac{1}{\sqrt{\xi_1}} \sum_{i=0}^{L_1-1} e^{i\theta_i} \sqrt{|y_{N-1-i}|} |i\rangle |\mathbf{X}_w\rangle \quad (11)$$

执行受控置换操作 (P^\dagger) ^{i} , 其量子电路详见附录 A:

$$\frac{1}{\sqrt{\xi_1}} \sum_{i=0}^{L_1-1} e^{i\theta_i} \sqrt{|y_{N-1-i}|} |i\rangle (P^\dagger)^i |\mathbf{X}_w\rangle \quad (12)$$

最后执行逆操作 \mathcal{O}_1^\dagger :

$$\begin{aligned} & \frac{1}{\xi_1} |0\rangle \sum_{i=0}^{L_1-1} e^{i\theta_i} |y_{N-1-i}| P^{L_1-i} |\mathbf{X}_w\rangle + |\perp\rangle = \\ & \frac{1}{\xi_1} |0\rangle U_w |\mathbf{X}_w\rangle + |\perp\rangle = \\ & \frac{1}{\xi_1 \|\mathbf{X}_w\|} |0\rangle \mathbf{Z}_w + |\perp\rangle \end{aligned} \quad (13)$$

其中子空间 $|\perp\rangle$ 正交于子空间 $\frac{1}{\xi_1} |0\rangle \sum_{i=0}^{L_1-1} U_w |\mathbf{X}_w\rangle$, 满足 $|0\rangle \langle 0| \otimes \mathbf{I} |\perp\rangle = 0$.

使用测量算子 $(|0\rangle^{\otimes \lceil \log N \rceil} \langle 0|^{\otimes \lceil \log N \rceil}) \otimes \mathbf{I}$ 对量子系统进行测量, 当量子寄存器 QR_1 测量结果是 $|0\rangle^{\otimes \lceil \log N \rceil}$ 时, 那么量子寄存器 QR_0 存储的量子态是 $\mathbf{Z}_w/\xi_1 \|\mathbf{X}_w\|$, 它与期望的卷积结果 \mathbf{Z}_w 成正比例关系. 由式 (13) 可得量子寄存器 QR_1 中测量得到 $|0\rangle^{\lceil \log N \rceil}$ 的概率是 $\|\mathbf{U}_w |\mathbf{X}_w\rangle\|^2/\xi_1^2$, 可以看出概率依赖于卷积核元素和输入序列元素, 与卷积核元素绝对值的和成反比, 与输入序列的范数成正比. 为提高量子寄存器 QR_1 是 0 的结果, 可使用振幅放大算法 (QAA).

基于量子一维宽卷积的实现框架, 有两种不同的方法可以实现量子一维等宽卷积和量子一维窄卷积. 第一种方法是在量子一维宽卷积的量子电路上再增加若干个置换操作, 对于量子一维等宽卷积而言, 额外增加 $\lfloor N \rfloor$ 个 P^\dagger 操作 (图 1(b) 所示), 对于量

子一维窄卷积而言, 额外增加 $N-1$ 个 P^\dagger 操作 (图 1(c) 所示). 第二种方法是仿照量子一维宽卷积算法, 对于量子一维等宽卷积, 重新制备 \mathbf{X}_e 和 \widehat{U}_e , 对于量子一维窄卷积, 重新制备 \mathbf{X}_n 和 \widehat{U}_n , 使用它们替代 \mathbf{X}_w 和 \widehat{U}_w . 方法一是基于量子一维宽卷积, 会保留一些垃圾元素, 而方法二不会保留垃圾元素.

通常输入序列的尺寸 M 和卷积核的尺寸 N 满足 $M \gg N$. 量子一维线性卷积算法的空间复杂度是 $O(\log M)$. 由于酉操作 \mathcal{O}_0 和 \mathcal{O}_1 的时间复杂度分别是 $O(\log M)$ 和 $O(\log N)$, 受控相位 CP 操作的时间复杂度是 $O(\log N)$, 受控置换操作的时间复杂度是 $O(\log^2 M)$, 逆操作 \mathcal{O}_1^\dagger 的时间复杂度是 $O(\log N)$, 故量子一维线性卷积算法的时间复杂度是 $O(\log^2 M)$.

相对于经典卷积时间复杂度 $O(M)$, 量子一维线性卷积算法实现了指数加速. 文献 [12] 从卷积定义角度出发, 首先以卷积核长度向输入序列截取对应长度进行内积计算, 之后保持输入序列不变, 卷积核进行一步移位操作, 直到短向量最后一个元素与输入序列最后一个元素对齐并计算其内积, 最终得到量子一维窄卷积结果. 它的时间复杂度是 $O(\log^2 M)$, 但算法无法扩展到量子一维宽卷积和等宽卷积. 而本文从线性代数的角度出发, 首先给出了量子一维宽卷积算法, 并且可以很容易地将其推广到量子一维等宽卷积和量子一维窄卷积上.

2.2 量子二维线性卷积

如式 (2) 所示, 二维宽卷积可以写成矩阵形式 $U_w \mathbf{X} = \mathbf{Z}_w$, 其中 U_w 是一般矩阵, 为了使量子计算能够求解该线性方程组, 我们需要重新调整该数据结构形式. 在矩阵 \mathbf{X} 四周进行补 0 操作得到矩阵 $\widehat{\mathbf{X}}$, 然后将其向量化

$$\widehat{\mathbf{X}} = [0, \dots, x_{0,0}, x_{0,1}, \dots, x_{M-1,M-1}, \dots, 0]^T \quad (14)$$

令

$$\mathbf{R}_0 = [y_{N-1,N-1}, \dots, y_{0,0}, 0, \dots, 0, \dots, 0] \quad (15)$$

其中前 N^2 个元素来自于卷积核 \mathbf{Y} 行向量化, 把 \mathbf{R}_0 作为首行向量, 构造循环矩阵

$$\begin{bmatrix} y_{N-1,N-1} & \cdots & y_{0,0} & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & y_{N-1,N-1} & y_{N-1,N-2} & \cdots & 0 & \cdots & 0 \\ 0 & \cdots & 0 & y_{N-1,N-1} & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & y_{N-1,N-1} & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ y_{N-1,N-2} & \cdots & 0 & 0 & \cdots & 0 & \cdots & y_{N-1,N-1} \end{bmatrix} \quad (16)$$

其中, $\widehat{\mathbf{X}} \in \mathbf{R}^{L_t^2}$, $\widehat{\mathbf{U}}_w \in \mathbf{R}^{L_t^2 \times L_t^2}$, $L_t = 2^{\lceil \log(M+2N-2) \rceil}$. 于是二维宽卷积可以重新写成线性方程 $\widehat{\mathbf{U}}_w \widehat{\mathbf{X}} = \widehat{\mathbf{Z}}_w$, 其中

$$\widehat{\mathbf{Z}}_w = [z_{0,0}, \dots, z_{N-1,N-1}, z_{N-1,N-2}, \dots, z_{M+N-2,M+N-2}, \dots, z_{L_t,L_t}]^T \quad (17)$$

将 $\widehat{\mathbf{X}}$, \mathbf{R}_0 相关信息存储到量子随机存取存储器.

$\widehat{\mathbf{U}}_w \widehat{\mathbf{X}} = \widehat{\mathbf{Z}}_w$ 求解过程与量子一维线性卷积中的方法相似, 实现电路也是与图 1(a) 相似. 不同之处主要在于, 一方面, 图中酉操作 \mathcal{O}_0 和 \mathcal{O}_1 重新定义为

$$\begin{aligned} \mathcal{O}_0 : |0\rangle^{\otimes 2\log L_t} &\mapsto \frac{1}{\sum_i |\widehat{X}_i|^2} \sum_{i=0}^{L_t^2-1} \widehat{X}_i |i\rangle \\ \mathcal{O}_1 : |0\rangle^{\otimes 2\lceil \log N \rceil} &\mapsto \frac{1}{\sum_j |R_{0,j}|} \sum_{j=0}^{2^{\lceil \log N \rceil}-1} \sqrt{R_{0,j}} |j\rangle \end{aligned} \quad (18)$$

其中 \widehat{X}_i 表示 $\widehat{\mathbf{X}}$ 的第 i 个元素, $R_{0,j}$ 表示向量 \mathbf{R}_0 中第 j 个元素. 另一方面, 图中受控相位操作和置换操作根据卷积核和输入序列的结构需要做出对应的调整.

使用测量算子 $(|0\rangle^{\otimes 2\lceil \log N \rceil} \langle 0|^{\otimes 2\lceil \log N \rceil}) \otimes \mathbf{I}$ 对量子寄存器进行测量, 当 QR_1 测量结果是 $|0\rangle^{2\lceil \log N \rceil}$ 时, QR_0 存储的量子态与我们想要获取的卷积结果 \mathbf{Z}_w 成正比. 为提高 QR_1 获取结果是 0 的概率, 可使用振幅放大算法 (QAA).

量子二维宽卷积的实现方法可以直接扩展到量子二维等宽卷积和量子二维窄卷积的量子实现上, 有两种不同的实现方法. 第一种方法是重新制备 $\widehat{\mathbf{X}}_e$ 和 $\widehat{\mathbf{U}}_e$ ($\widehat{\mathbf{X}}_e$ 和 $\widehat{\mathbf{U}}_e$) 以替代 $\widehat{\mathbf{X}}_w$ 和 $\widehat{\mathbf{U}}_w$. 第二种方法是基于量子二维宽卷积的量子电路, 分别在水平方向向左平移 $\lfloor N \rfloor$ ($N-1$) 次, 垂直方向向上平移 $\lfloor N \rfloor$ ($N-1$) 次, 这种平移操作可以使用 P^\dagger 操作来实现.

通常输入序列的尺寸 M 和卷积核的尺寸 N 满足 $M \gg N$. 量子二维线性卷积的空间复杂度是 $O(2\log M)$, 时间复杂度是 $O(4\log^2 M)$. 相对于经典二维线性卷积时间复杂度 $O(M^2)$, 量子二维线性卷积展现了其指数加速的优势. 文献 [13] 给出了量子二维窄卷积算法, 输入序列的量子态与卷积核的量子态进行张量积计算, 使用两种置换操作共同完成量子卷积振幅置换, 对置换后的卷积核量子态的量子比特执行 H 门操作, 输入序列量子态的量子比特

执行单位矩阵操作, 对卷积核量子态进行测量. 该算法的空间复杂度和时间复杂度与本文相同, 但其思考角度与本文不同, 且其实现不适用于量子二维宽卷积和量子二维等宽卷积.

2.3 量子卷积算法讨论及扩展

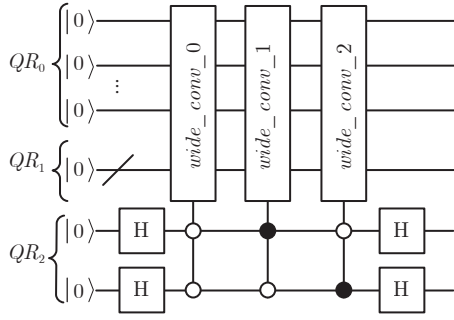
上文介绍了单通道, 单位步长, 零填充情况下的量子一维宽卷积, 等宽卷积, 窄卷积, 量子二维宽卷积, 等宽卷积, 窄卷积的量子实现. 下面我们重点讨论量子一维宽卷积算法的实现和扩展, 其他几种量子卷积实现和扩展与它相似.

在没有对输入序列进行填充的情况下, 一维宽线性卷积可以写成线性方程 $\mathbf{U}_w \mathbf{X}^T = \mathbf{Z}_w^T$ 形式 (见式 (1)), 其中 $\mathbf{U}_w \in \mathbf{R}^{(M+N-1) \times M}$. 有三种量子算法可以求解这样的线性方程. 第一种方法是使用类 HHL 算法^[5,40], 首先将矩阵 \mathbf{U}_w 变成厄米矩阵 $\widetilde{\mathbf{U}}_w$:

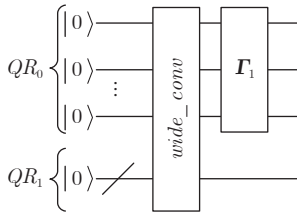
$$\begin{bmatrix} 0 & \mathbf{U}_w \\ \mathbf{U}_w^\dagger & 0 \end{bmatrix} \quad (19)$$

然后对该厄米矩阵进行有效哈密顿模拟 $e^{-i\widetilde{\mathbf{U}}_w t}$, 最后使用类 HHL 算法的量子电路实现. 这种方法要求哈密顿量可以有效模拟, 且需要使用到量子相位估计算法. 第二种方法是使用交换测试 (swap test)^[41], 将矩阵 \mathbf{U}_w 的每一行与列向量 \mathbf{X}^T 做内积计算, 由于内积结果隐藏在概率中, 所以需要进行大量的量子测量. 文献 [42] 中给出了一种使用 swap test 的量子矩阵乘法算法, 但它同样需要用到量子相位估计算法. 第三种方法是基于线性酉组合方法 (LCU). 虽然三种方法都可以实现量子计算求解, 但是前两种方法时间复杂度高且均以高昂的量子比特资源为代价. 因此本文使用第三种方法. 此外, 线性酉组合方法的实现有多种, 但是相对于其他 LCU 方法, 本文使用的 LCU 方法^[39] 构造简单, 易于实现, 且时间复杂度低, 所需量子比特数少.

本文提出的量子一维宽线性卷积适用于扩展到多通道, 非单位步长, 非零补充的情况. 以三通道为例, 量子电路实现如图 2(a) 所示, 在量子一维宽卷积的量子电路的基础上, 额外再添加一个两量子比特的量子寄存器 QR_2 , 此时 QR_0 用于存储输入序列信息, QR_1 用于存储卷积核信息, QR_2 用于存储通道信息. 实现步骤如下, 首先应用两个 H 门在 QR_2 , 接着根据通道信息在图像上使用不同的卷积核, 最后再在 QR_2 使用两个 H 门, 实现卷积结果融合. 以步长为 2 为例, 量子电路见图 2(b) 在量子一维宽卷积的量子电路基础上, 再增加一个置换矩阵:



(a) 三通道的量子一维宽卷积实现电路
(a) Quantum circuit of one-dimensional wide convolution in the case of three channels



(b) 步长为2的量子一维宽卷积实现电路, Γ_1 是置换矩阵
(b) Quantum circuit of one-dimensional wide convolution in the case of stride of 2, in which Γ_1 is the permutation matrix

图 2 三通道/两步长的量子一维宽卷积实现电路

Fig. 2 Three-channel/two-stride quantum one-dimensional wide convolution realization circuit

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \quad (20)$$

图 A1(d) 给出了 $\Gamma_1 \in \mathbf{R}^{8 \times 8}$ 时的量子电路. 以补充值是 1 为例, 此时式 (3) 改写成

$$\mathbf{X}_w = \left[\underbrace{1, \cdots, 1}_{N-1}, \underbrace{x_0, x_1, \cdots, x_{M-1}}_M, \underbrace{1, \cdots, 1}_{N-1}, \underbrace{0, \cdots, 0}_{L_1-L_w} \right] \quad (21)$$

3 量子卷积在图像处理中的应用

本节研究量子二维线性卷积在量子图像中的应用, 例如量子图像平滑, 量子图像锐化和量子图像边缘检测, 旨在验证量子卷积算法本身的正确性及基于量子卷积实现量子图像处理算法的可行性. 首

先分别给出它们的量子算法, 然后在 Qiskit^[43] 上使用 QasmSimulator 进行模拟, 最后将它们和已有的量子图像处理算法比较. 仿真平台的参数是操作系统 Ubuntu 20.04.2 LTS 64 位, 处理器 CPU I9-10900X 10 核 20 进程 3.70 GHz, 内存是 62.5 GB, Qiskit 版本是 0.26.2.

3.1 量子图像平滑

图像平滑常见操作有模糊处理和降噪处理, 通常用于图像预处理任务中. 模糊处理可在大目标之前去除图像中的一些琐碎细节, 连接直线或曲线的缝隙. 通过线性滤波和非线性滤波模糊处理, 可以降低噪声. 本节研究量子图像平滑, 其基本流程是首先编码量子图像和卷积核 (见附录 B), 接着使用量子二维线性卷积实现量子图像平滑.

以图 3 为例, 图 3(a) 是含有泊松噪声的实验图像 I_p , 图 3(c) 是含有高斯噪声 (均值为 0, 方差为 0.01) 的实验图像 I_g , 它们的尺寸是 60×60 . 图 3(b) 是均值滤波器, 图 3(d) 是高斯滤波器, 它们的尺寸是 3×3 . 量子图像平滑算法对应的量子电路图如图 4(a) 所示, 最上面的 10 个量子比特 (QR_0) 用于存储量子图像信息, 最下面的 4 个量子比特 (QR_1) 用于存储卷积核信息, 最后一个是经典寄存器 (CR), 它有四个比特用于存储量子测量后坍塌的经典信息. 需要特别指出, 不同于前面描述的量子卷积的量子电路, 由于本节使用的两个平滑滤波器的元素都是正数, 因此在量子电路设计过程中, 受控相位操作被省略. 我们使用 Qiskit 的 QasmSimulator 模拟量子图像平滑算法, 其中参数 *shots* 设置为 1 024, 实现代码见^{1,2}. 模拟结果输出状态向量 (状态向量的前 1024 个元素表示平滑滤波后的图像) 以及量子寄存器 QR_1 是全 0 的概率.

图 4(b) 表明了均值滤波时输出 0 的概率是 0.935, 图 4(c) 显示了量子模拟器输出的均值滤波后的图像, 图 4(d) 显示了经典计算机使用 SciPy³ 输出的均值滤波后的图像. 两者在视觉上没有差异且抑制了泊松噪声. 同样地, 图 4(e) 表明了高斯滤波器时输出 0 的概率是 0.905, 图 4(f) 显示了量子模拟器输出的高斯滤波后的图像, 图 4(g) 显示了经典计算机使用 SciPy 输出的高斯滤波后的图像, 两者在视觉上没有差异且能抑制了高斯噪声. 相比于均方误差 (Mean-square error, MSE) 和峰值信噪比 (Peak signal-to-noise ratio, PSNR), 结构相似性 (Structural similarity, SSIM) 的评价性能有明

¹ https://github.com/liuxingao/SimulateCode/blob/main/NORMAL/mean_filter.ipynb

² https://github.com/liuxingao/SimulateCode/blob/main/NORMAL/gaussian_filter.ipynb

³ SciPy 包: <https://docs.scipy.org/doc/scipy/reference/>

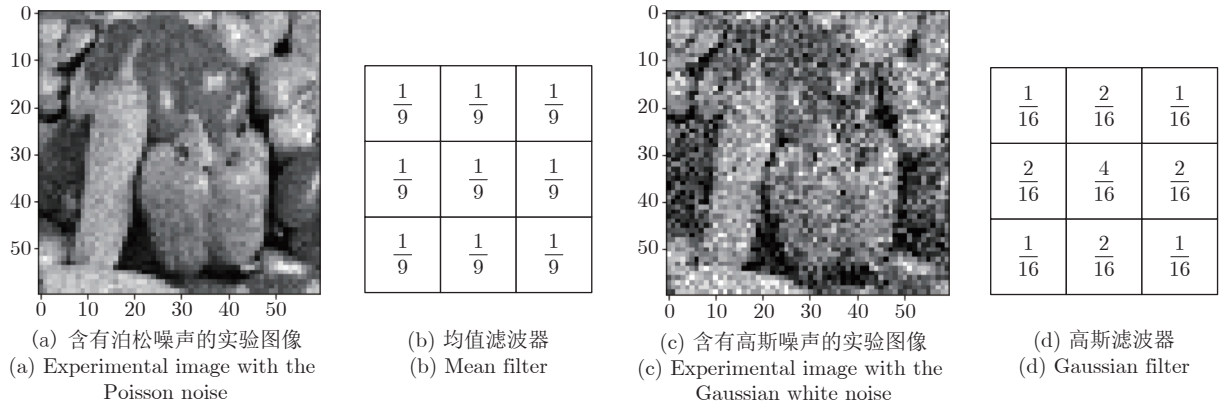


图 3 量子图像平滑的实验图像和平滑滤波器

Fig.3 Experimental image and smoothing filter for quantum image smoothing

显提高^[44],所以本文使用 SSIM 评估两幅图像的相似性. 均值滤波后的两幅图像的 SSIM 是 0.999 856 7, 高斯滤波后的两幅图像的 SSIM 是 0.999 923 3, 这表明两幅图像相同, 同时, 验证了量子图像平滑算法和量子卷积算法的可行性和正确性.

3.2 量子图像锐化

图像锐化可以补偿图像的轮廓, 增强图像的边缘及灰度跳变的部分, 使图像变得清晰. 本节研究量子图像锐化, 其基本步骤是首先对图像和卷积核编码, 接着使用量子二维卷积算法对图像滤波, 需要注意的是, 与平滑滤波算子全是正数元素不同, 锐化滤波算子存在负数元素, 需要使用受控相位操作给负数元素项增加相对相位, 可以得到量子态:

$$|\psi_0\rangle = \frac{1}{\xi_1 \|\mathbf{X}_w\|} |0\rangle \mathbf{Z}_w + |\perp\rangle = \frac{1}{\xi_1 \|\mathbf{X}_w\|} |0\rangle \sum_{i,j} z_{w,i,j} |ij\rangle + |\perp\rangle \quad (22)$$

其中 $\mathbf{Z}_w = \sum_{(i,j)} z_{w,i,j} |ij\rangle$ 是卷积滤波后的结果. 然后使用 Real QADC 方法将卷积结果从振幅提取到基态中, 可以得到量子态:

$$|\psi_1\rangle = \frac{1}{L_t} |0\rangle \sum_{i,j} |ij\rangle \left| \frac{z_{w,i,j}}{\xi_1 \|\mathbf{X}_w\|} \right\rangle + |\perp\rangle \quad (23)$$

最后添加额外辅助量子比特, 使用量子比较器^[45] 分别比较 $\frac{z_{w,i,j}}{\xi_1 \|\mathbf{X}_w\|}$ 与 0, $\frac{z_{w,i,j}}{\xi_1 \|\mathbf{X}_w\|}$ 与 $\frac{255}{\xi_1 \|\mathbf{X}_w\|}$, 如果 $\frac{z_{w,i,j}}{\xi_1 \|\mathbf{X}_w\|}$ 小于 0, 则将其置为 0; 如果 $\frac{z_{w,i,j}}{\xi_1 \|\mathbf{X}_w\|}$ 大于 $\frac{255}{\xi_1 \|\mathbf{X}_w\|}$, 则将其置为 $\frac{255}{\xi_1 \|\mathbf{X}_w\|}$.

以图 5 为例, 图 5(a) 是实验图像, 尺寸是 60×60 , 图 5(b) 是锐化算子, 尺寸是 3×3 . 经典计算机模拟

量子计算时, 它的内存资源消耗和运行时间随量子电路深度、电路宽度变化呈指数增长. 受现有量子实验设备以及仿真平台环境的约束, 我们只能使用 Qiskit 仿真实验步骤 1, 至于步骤 2、3 将在步骤一的基础上按照逻辑进行经典运算, 实现代码见⁴. 量子图像锐化算法步骤一对应的量子电路图如图 6(a) 所示, 未优化分解之前量子电路的电路宽度是 20, 电路深度是 23. 值得注意的是, 受控相位操作中, 除了量子态 (下标是 1, 3, 5, 7) 的相对相位翻转, 还有其他量子态 (下标是 2, 4, 6, 8) 的相对相位也偏转成负值. 相对相位对量子算法的结果影响很大, 因此需要额外的受控相位门将量子态 (下标是 2, 4, 6, 8) 的相对相位恢复成正相位. 我们使用 Qiskit 的 QasmSimulator 模拟量子图像锐化算法, 其中参数 *shots* 设置为 1024. 模拟结果输出状态向量 (状态向量的前 1024 个元素表示锐化后的图像) 以及量子寄存器 QR_1 是全 0 的概率. 图 6(b) 表明输出 0 的概率是 0.026, 图 6(c) 显示了量子模拟器输出的锐化后图像, 图 6(d) 显示了经典计算机使用 SciPy 输出的锐化后图像. 两者在视觉上没有差异, 且两幅图像的 SSIM 是 1.0, 这表明两幅图像相同, 同时, 验证了量子图像锐化算法和量子卷积算法的可行性和正确性.

3.3 量子图像边缘检测

边缘检测是图像处理和计算机视觉中的基本问题, 边缘检测的目的是标识数字图像中亮度变化明显的点. 图像边缘检测可大幅度地减少数据量, 剔除不相关的信息, 保留图像重要的结构属性. 本节研究量子图像边缘检测, 使用水平和垂直方向的 Sobel 算子, 最终获得的梯度图像是 $Z_{xy} = \sqrt{Z_x^2 + Z_y^2}$

⁴ https://github.com/liuxingao/SimulateCode/blob/main/NORMAL/sharpen_filter.ipynb

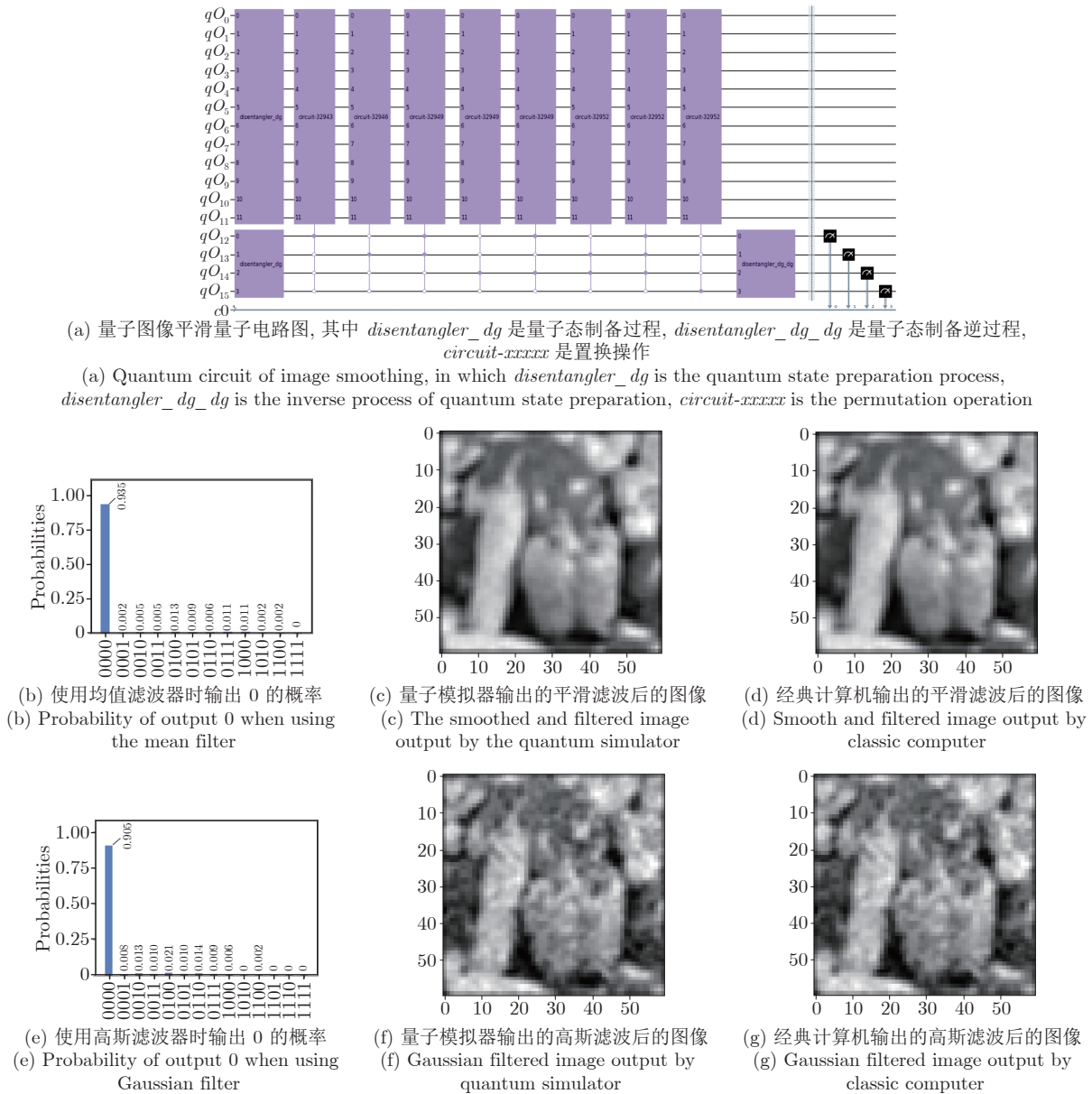


图 4 量子图像平滑的仿真电路和仿真结果

Fig.4 Simulation circuit and simulation results of quantum image smoothing

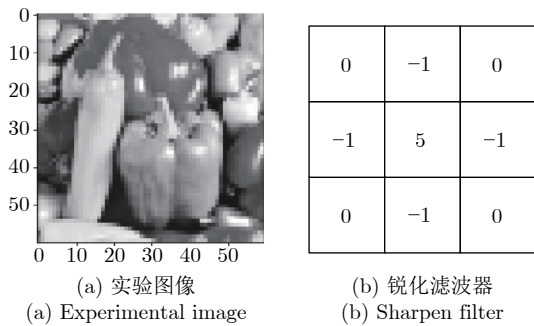


图 5 量子图像锐化的实验图像和锐化滤波器

Fig.5 Experimental image and sharpening filter for quantum image sharpening

量子图像边缘检测的实现过程如下:

步骤 1. 准备 6 个量子寄存器, 最左边表示第 1 个量子寄存器, 最右边表示第 6 个量子寄存器, q_1 表示存储卷积核所需量子比特数, q_2 表示存储图像所需量子比特数:

$$|\psi\rangle = |0\rangle |0\rangle |0\rangle^{\otimes q_1} |0\rangle^{\otimes q_2} |0\rangle^{\otimes q_1} |0\rangle^{\otimes q_2} \quad (24)$$

步骤 2. 对第 2 个量子寄存器执行 H 门操作:

$$|\psi\rangle = \frac{1}{\sqrt{2}} \left[|0\rangle |0\rangle |0\rangle^{\otimes q_1} |0\rangle^{\otimes q_2} |0\rangle^{\otimes q_1} |0\rangle^{\otimes q_2} + |0\rangle |1\rangle |0\rangle^{\otimes q_1} |0\rangle^{\otimes q_2} |0\rangle^{\otimes q_1} |0\rangle^{\otimes q_2} \right] \quad (25)$$

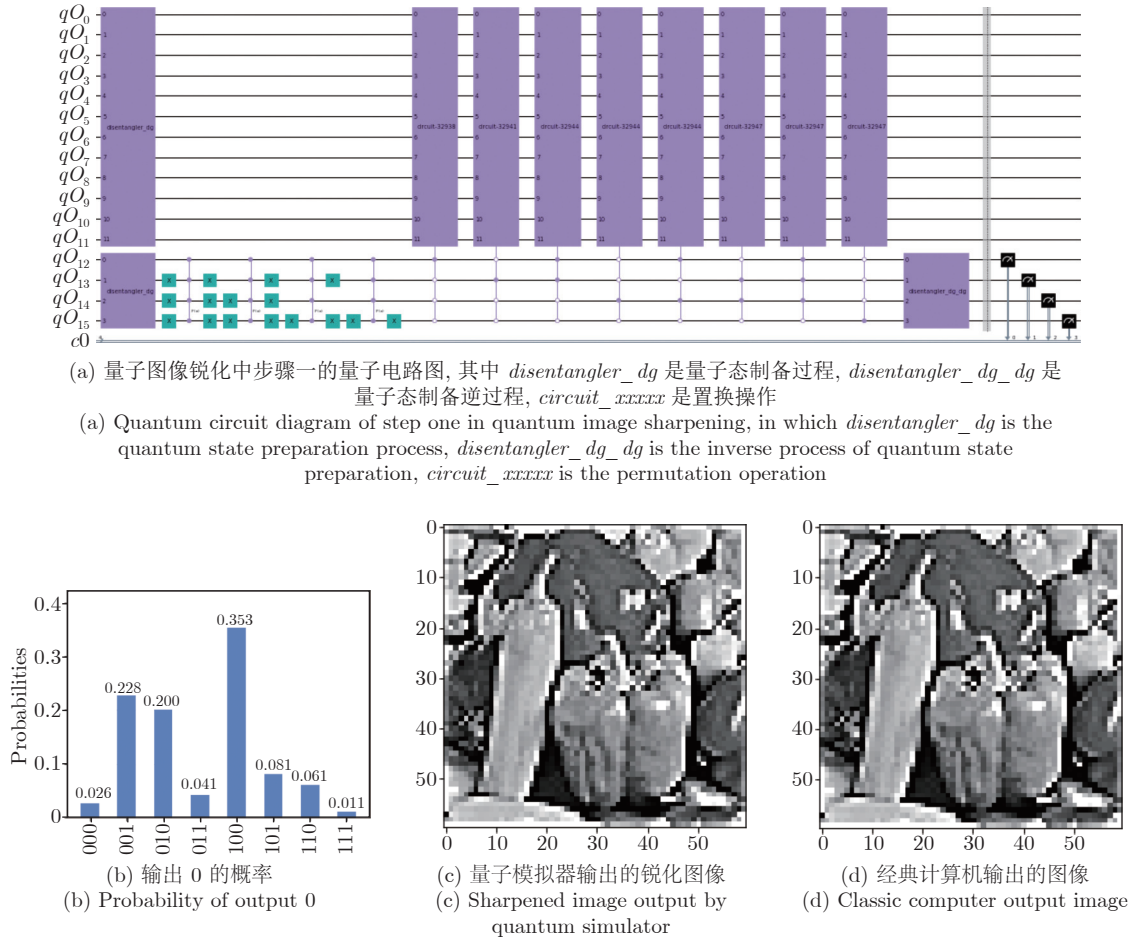


图 6 量子图像锐化的仿真电路和仿真结果

Fig. 6 Simulation circuit and simulation results of quantum image sharpening

步骤 3. 当第 2 个量子寄存器是 $|0\rangle$ 时, 对第 3 个和第 4 个量子寄存器执行量子卷积操作, 卷积核是水平方向 Sobel 算子 (Sobelx), 在第 5 和 6 个量子寄存器执行同样的量子卷积操作; 当第 2 个量子寄存器是 $|1\rangle$ 时, 对第 3 个和第 4 个量子寄存器执行量子卷积操作, 卷积核是垂直方向 Sobel 算子 (Sobely), 在第 5 和 6 个量子寄存器上执行同样的量子卷积操作:

$$|\psi\rangle = \frac{1}{\sqrt{2}} \left\{ |0\rangle |0\rangle \left[\frac{1}{\xi_x^2 \|\mathbf{X}_w\|^2} |0\rangle \sum_{x_1} Z_{x_1} |x_1\rangle |0\rangle \times \sum_{x_2} Z_{x_2} |x_2\rangle + |G\rangle \right] + |0\rangle |1\rangle \left[\frac{1}{\xi_y^2 \|\mathbf{X}_w\|^2} |0\rangle \times \sum_{y_1} Z_{y_1} |y_1\rangle |0\rangle \sum_{y_2} Z_{y_2} |y_2\rangle + |G\rangle \right] \right\} \quad (26)$$

步骤 4. 第 2 个量子寄存器执行 H 门操作:

$$|\psi\rangle = \frac{1}{2} \left\{ |0\rangle |0\rangle \left[\frac{1}{\xi_x^2 \|\mathbf{X}_w\|^2} \sum_{x_1} \sum_{x_2} Z_{x_1} Z_{x_2} \times |0\rangle |x_1\rangle |0\rangle |x_2\rangle + \frac{1}{\xi_y^2 \|\mathbf{X}_w\|^2} \sum_{y_1} \sum_{y_2} Z_{y_1} Z_{y_2} \times |0\rangle |y_1\rangle |0\rangle |y_2\rangle \right] + |G\rangle \right\} \quad (27)$$

步骤 5. 把第 6 个量子寄存器中量子比特作为控制比特, 第 4 个量子寄存器中的量子比特作为目标比特, 执行 q_2 个受控非门操作:

$$|\psi\rangle = \frac{1}{2\xi^2 \|\mathbf{X}_w\|^2} |0\rangle |0\rangle |0\rangle |0\rangle |0\rangle \left[\sum_{\rho} (Z_{x,\rho}^2 + Z_{y,\rho}^2) |\rho\rangle \right] + |G\rangle \quad (28)$$

其中, $x = x_1 = x_2$, $y = y_1 = y_2$, $\xi = \xi_x = \xi_y$, $\rho = x = y = 0, \dots, L_t - 1$, 该量子态对应的状态向量的前 L_t 个元素就是 Z_{xy}^2 .

步骤 6. 使用 Real QADC 方法将振幅添加到基态中:

$$|\psi\rangle = \frac{1}{2\xi^2 L_t \|\mathbf{X}_w\|^2} \sum_{\rho} |Z_{x,\rho}^2 + Z_{y,\rho}^2\rangle |0\rangle |0\rangle |0\rangle |0\rangle |\rho\rangle + |G\rangle \quad (29)$$

步骤 7. 使用量子比较器^[45]比较第 1 个量子寄存器与预设的阈值 $thre$, 如果大于阈值则第 1 个量子寄存器设置为 1, 否则第 1 个量子寄存器置为 0:

$$|\psi\rangle = \frac{1}{2\xi^2 L_t \|\mathbf{X}_w\|^2} \left[\sum_{Z_{x,\rho}^2 + Z_{y,\rho}^2 > thre} |1\rangle |0\rangle |0\rangle |0\rangle |0\rangle |\rho\rangle + \sum_{Z_{x,\rho}^2 + Z_{y,\rho}^2 \leq thre} |0\rangle |0\rangle |0\rangle |0\rangle |0\rangle |\rho\rangle \right] + |G\rangle \quad (30)$$

以图 7 为例, 图 7(a) 是实验图像, 尺寸是 12×12 , 图 7(b) 是水平方向的 Sobel 算子, 尺寸是 3×3 , 图 7(c) 是垂直方向的 Sobel 算子, 尺寸是 3×3 . 受仿真平台环境约束, 我们只使用 Qiskit 仿真实现步骤 1 至 5, 步骤 6、7 将在此基础上按逻辑进行经典运算, 实现代码见⁵. 图 8(a) 给出了量子图像边缘检测步骤 1 到 5 的量子电路图, 使用 Qasm-Simulator 对它进行模拟, 参数 $shots$ 设置为 1024. 模拟结果输出状态向量 (状态向量的前 256 个元素是边缘检测后的图像) 以及量子寄存器 QR_1 是 0 的概率. 图 8(b) 显示了输出是 0 的概率, 图 8(c) 显示量子模拟器输出的图像, 图 8(d) 显示了经典计算机使用 SciPy 输出的图像. 两者在视觉上没有差异且两幅图像的 SSIM 是 1.0, 这说明两幅图像相同, 同时, 验证了量子卷积算法的正确性和量子图像边缘检测算法的可行性.

量子图像边缘检测的效果受许多因素影响, 例如算子, 方向, 阈值等. 本节提出的基于水平垂直两个方向 Sobel 算子的量子图像检测算法可以很容易扩展到四个方向 Sobel 算子.

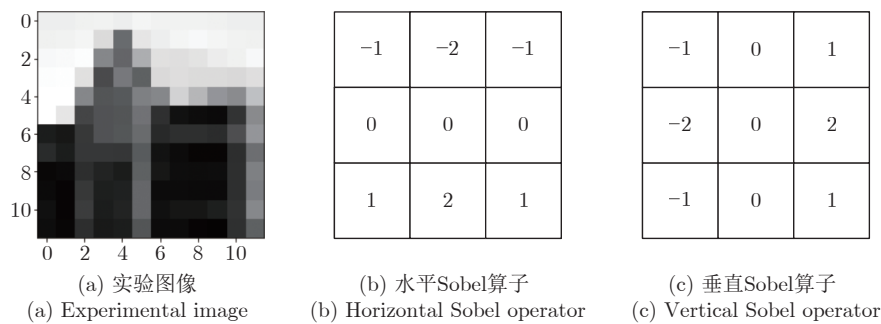


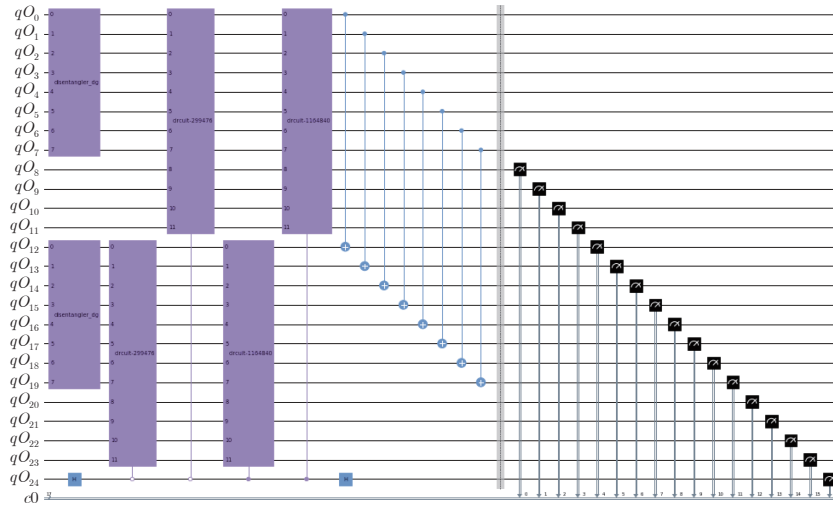
图 7 量子图像边缘检测的实验图像和 Sobel 算子

Fig. 7 Experimental image and Sobel operator of quantum image edge detection

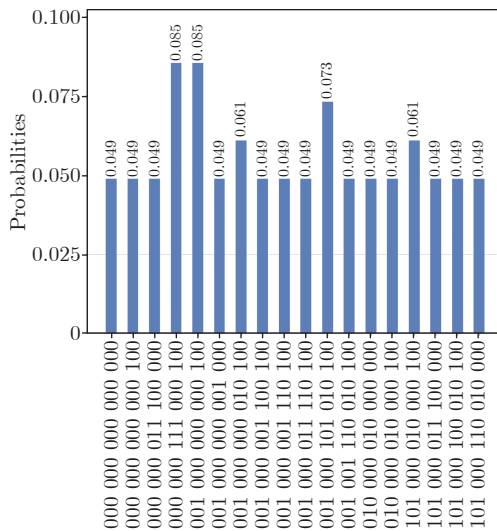
3.4 量子图像处理的讨论

基于量子卷积操作的量子图像处理算法已被广泛研究, 主要涉及量子图像滤波, 量子图像特征提取, 量子图像边缘检测. 这些算法实现量子卷积操作的思路大致如下, 首先获取邻域像素值, 接着根据滤波器的元素, 使用量子算术实现卷积计算. 其中邻域像素值获取方法可以细分为三类. 假设卷积核算子的尺寸为 3×3 , 图像尺寸是 $M \times M$, $M = 2^m$, 图像灰度值范围是 $[0, 2^q - 1]$, $q = 1$ 表示二值图像, $q = 8$ 表示灰度图像, $q = 24$ 表示彩色图像. 第一类方法是制备 9 份完全相同的量子图像, 将循环操作分别作用于 8 份量子图像, 坐标信息相同的量子态构成邻域像素. 第二类方法是制备 1 份量子图像和用于存储像素值的 8 个量子寄存器, 依次使用循环操作和复制操作来获取相邻像素值. 第三类方法是制备 1 份量子图像和用于存储量子图像的 8 个量子寄存器, 依次使用量子比较器和复制操作获取相邻像素值. 如表 1 所示, 这些算法的空间复杂度和时间复杂度都比较高. 此外, 第二类算法存在错误, 量子图像的位置信息和像素值信息是纠缠的, 移位后再复制的像素值都是相同的, 通过理论推导或 Qiskit 仿真可以验证. 文献^[27]中给出的量子图像检测算法只适用于二值图像, 因此这里不做比较.

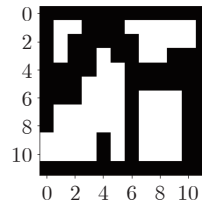
量子图像平滑算法需要 $2m + 4$ 个量子比特, 时间复杂度是 $O(4m^2)$; 量子图像锐化算法需要 $2m + 2\log(1/\varepsilon) + 5$ 个量子比特, 时间复杂度是 $O(4m^2/\varepsilon + 16\log(1/\varepsilon) - 3)$, ε 是计算精度; 量子图像边缘检测需要 $4m + 2\log(1/\varepsilon) + 7$ 个量子比特, 时间复杂度是 $O(16m^2/\varepsilon + 16\log(1/\varepsilon) - 3)$. 本文提出的量子图像滤波 (平滑和锐化) 算法, 量子图像边缘检测算法相比于其对应的经典算法, 在存储能力和计算效率上都实现了指数加速; 对比表 1 中量子算法, 在时间复杂度方面处于相同数量级, 在消耗的量子比特数方面占有明显优势, 使得它更有希望在目前的量



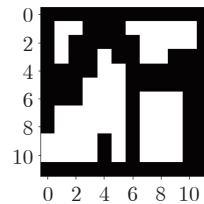
(a) 量子图像边缘检测步骤1-5的仿真电路图, 其中 *disentangler_dg* 是量子态制备过程, *circuit-xxxx* 是二维量子卷积操作
 (a) The simulation circuit diagram of quantum image edge extraction steps 1-5, in which *disentangler_dg* is the quantum state preparation process, *circuit-xxxx* is the 2D quantum convolution operation



(b) 输出 0 的概率
 (b) Probability of output 0



(c) 量子模拟器输出的图像
 (c) Image output by quantum simulator



(d) 经典计算机输出的图像
 (d) Classic computer output image

图 8 量子图像边缘检测的仿真电路和仿真结果

Fig.8 Simulation circuit and simulation results of quantum image edge detection

子计算机上实现. 但是基于本文提出的量子二维卷积操作的量子图像处理算法不适用于非线性量子图像滤波情况^[46-50].

4 总结及未来工作

本文提出了单通道, 单位步长和零补充情况下的 QOWC, QOEC, QONC, QTWC, QTEC, QTNC. 当通道, 步长, 补充值等参数发生变化时, 其对应的算法实现只需轻微调整. 理论分析证明量子线性卷积的空间复杂度 $O(\log M)$ 和时间复杂度 $O(\log^2 M)$ 较经典线性卷积有指数级下降. 另外, 本文基于量子线性卷积和振幅编码方式提出了 QISM, QISH,

QIED 三种量子图像处理算法. 理论分析证明了提出的量子图像处理算法相比于经典算法, 在存储能力和计算效率上都实现了指数加速, 相比于其他编码方式的量子图像处理算法, 在消耗的量子比特数方面占有明显优势.

研究过程中我们发现两种情况, 一是量子图像锐化的获取概率较低, 为 0.026, 二是最高获取概率 0.367 对应的量子态也能重构出锐化效果的量子图像. 因此, 未来思考如何提高获取概率以及探究最高概率对应量子态能重构出锐化效果量子图像的背后原因. 另外, 量子卷积操作是量子神经网络的重要步骤, 如量子卷积网络和量子生成对抗网络^[51],

表 1 已被提出的量子图像滤波和量子图像特征边缘检测算法. 假设卷积核算子的尺寸为 3×3 , 图像尺寸是 $M \times M$, $M = 2^m$, q 表示图像的灰度值范围

Table 1 Quantum image filtering and quantum image feature edge detection algorithms have been proposed. Suppose the size of the convolution kernel operator is 3×3 , and the image size is $M \times M$, $M = 2^m$, q represents the range of gray values

图像处理	图像编码	卷积核算子	邻域获取	空间复杂度	时间复杂度	论文
图像滤波(平滑,锐化)	NEQR	任意卷积核	第一种	$18m + 9q$	$O(9m^2 + 9q^2)$	[14]
	NEQR	Gaussian核	第一种	$20m+10q + 1$	$O(m^2 2^{2m} + q^2 2^{2m})$	[15]
	NEQR	任意卷积核	第一种	$18m + 9q$	$O(9m^2 + 9q^2)$	[16]
量子图像边缘检测	NEQR	Sobel	第二种	$4m+18q$	$O(m^2 + q^2)$	[18]
	NEQR	Kirsch	第一种	$18m + 9q$	$O(m^2 + 2q^{+3})$	[19]
	NEQR	Zero-cross	第二种	$2m + 9q$	$O(m^2 + q^2)$	[20]
	GQIR	Sobel	第一种	$18m + 9q$	$O(m^2 + 2^{q+5} + q^2)$	[21]
	NEQR	Prewitt	第一种	$18m + 9q$	$O(m^2 + 2^{q+3})$	[22]
	NEQR	Sobel	第二种	$2m + 9q$	$O(m^2 + 2^{q+4})$	[23]
	NEQR	Sobel	第二种	$2m + 9q$	$O(10m^2 + 4q^2)$	[24]
	NEQR	Sobel	第一种	$18m + 9q$	$O(qm^2 + 9q^2)$	[25]
	FRQ	Sobel	第二种	$2m + 9q$	$O(10m^2 + q^2)$	[26]
	FRQI	Sobel	第二种	$4m + 10$	$O(10m^2 + q^2)$	[28]
	NEQR	Canny	第三种	$18m + 9q$	$O(48m^2 + 4q^2)$	[29]

这些都是未来值得深入研究的方向.

附录A 置换矩阵的量子电路及优化

任意置换矩阵的量子电路可使用量子可逆逻辑方法获得^[52-53]. 置换矩阵 P 的量子电路如图 A1(a) 所示, P^\dagger 的量子电路如图 A1(b) 所示, $\Gamma_1 \in \mathbf{R}^{8 \times 8}$ 的量子电路如图 A1(d) 所示, 置换矩阵 Q_8 (文献 [12, 13] 使用它置换振幅) 的正确量子电路应该是如图 A1(c) 所示. 量子电路图中, 低位量子比特在上面, 高位量子比特在下面. 由文献 [54] 可以推导出下面的结论:

推论 1. 假设输入量子态是 $|a\rangle = |a_{n-1}a_{n-2} \cdots a_1a_0\rangle$, 执行 $b = b_{n-1}b_{n-2} \cdots b_0$ 次置换操作 $P^\dagger \in \mathbf{R}^{2^n \times 2^n}$ 后, 那么输出的量子态 $|\hat{a}\rangle = (P^\dagger)^b |a\rangle = \sum_{k=0}^{n-1} (a_k - b_k) 2^k \bmod 2^{k+1}$. 当 b 可以写成指数形式 $b = 2^m$, $m = 0, 1, 2, \dots, n-1$ 时, $|\hat{a}\rangle = (a_h - 1) \bmod 2^{n-m} + a_l$, 其中 a_h 和 a_l 满足 $a = a_h * 2^m + a_l$.

证明. 当 b 不可以写成指数形式时,

$$\begin{aligned}
 (a - b) \bmod 2^{2n} &= \left[\sum_{k=0}^{2n-1} (a_k - b_k) \times 2^k \right] \bmod 2^{2n} = \\
 & \left[\bigvee_{k=0}^{2n-1} (a_k - b_k) \times 2^k \right] \wedge (2^{2n} - 1) = \\
 & \bigvee_{k=0}^{2n-1} \left[(a_k - b_k) \times 2^k \wedge (2^{2n} - 1) \right] = \\
 & \bigvee_{k=0}^{2n-1} \left[(a_k - b_k) \times 2^k \wedge (2^{k+1} - 1) \right] = \\
 & \sum_{k=0}^{2n-1} \left[(a_k - b_k) \times 2^k \bmod 2^{k+1} \right] \quad (A1)
 \end{aligned}$$

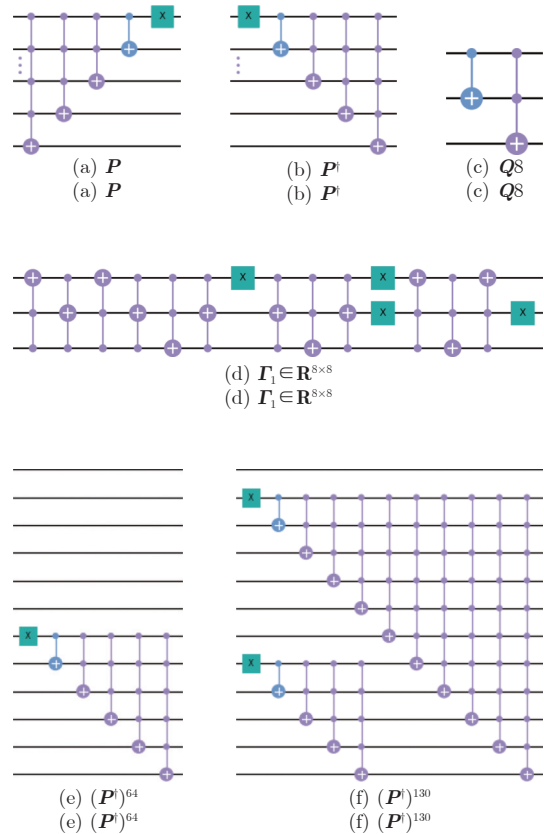


图 A1 置换矩阵的量子电路

Fig. A1 Quantum circuit of permutation matrix

其中 \vee 表示或运算, \wedge 表示与运算. 当 b 可以写成指数形式时,

$$\begin{aligned}
(a-b) \bmod 2^{2n} &= \\
[(a_h - 1) \times 2^m + a_l] \bmod 2^{2n} &= \\
[(a_h - 1) \times 2^m \vee a_l] \wedge (2^{2n} - 1) &= \\
[(a_h - 1) \times 2^m \wedge (2^{2n} - 1)] \vee [a_l \wedge (2^{2n} - 1)] &= \\
[(a_h - 1) \wedge (2^{2n-m} - 1)] \vee [a_l \wedge (2^{2n} - 1)] &= \\
(a_h - 1) \bmod 2^{2n-m} + a_l & \quad (A2)
\end{aligned}$$

□

推论 1 可用于简化量子电路, 降低电路深度, 例如 $\mathbf{P}^\dagger \in \mathbf{R}^{64 \times 64}$ 时, $(\mathbf{P}^\dagger)^{64}$ 对应的量子电路可以简化成图 A1(e), $(\mathbf{P}^\dagger)^{130}$ 对应的量子电路可以简化成图 A1(f), 它们将用于 Qiskit 仿真实验中, 可以简化量子电路, 缩短模拟仿真时间。

附录B 量子态制备和量子图像编码

根据已知的经典信息制备相应的量子态, 是量子算法设计过程中重要的步骤. 一方面, 量子态制备过程有时会统治整个量子算法的时间复杂度, 另一方面, 量子态制备方法决定了后续采取什么样的量子操作. 量子态的一般形式可以写成 $|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\varphi}\sin\frac{\theta}{2}|1\rangle$, 经典信息可以存储在基态 $|0\rangle, |1\rangle$, 相对相位 φ , 还有振幅 $\cos\frac{\theta}{2}, \sin\frac{\theta}{2}$ 中. 根据存储位置的不同, 量子态制备方法分为三类, 基态编码 (Basic code, BC)^[33, 55-56], 相位编码 (Phase code, PC)^[57] 和振幅编码 (Amplitude code, AC)^[26, 57-60]. 对于相同类型的量子态制备方法, 不同的研究领域里使用的定义存在差异.

在量子图像处理研究领域, 量子图像编码方法有基态编码, 如新型数字图像增强量子表示 (A novel enhanced quantum representation of digital images, NEQR)^[55], 广义量子图像表示 (The generalized quantum image representation, GQIR)^[61], 振幅编码, 如灵活表示的量子图像 (A flexible representation of quantum images, FRQI)^[58], 量子概率图像编码 (Quantum probability image encoding, QPIE)^[27], n-qubit 正常任意叠加态 (n-qubit normal arbitrary superposition state, NASS)^[57]. 其中 NEQR, FRQI, NASS 最为常用, 三种方法各有优劣^[62]. 假设图像的尺寸是 $M \times M$, 灰度值的范围是 $[0, 2^q - 1]$. NEQR 时间复杂度 ($O(2qM^2 \log M + 1)$) 低, 酉操作方便, 但空间复杂度 ($O(2 \log M + q)$) 高; FRQI 的时间复杂度 ($O(2M^2 \log M)$) 和空间复杂度 ($O(2 \log M + 1)$) 都低, 但是酉操作不方便; NASS 的空间复杂度 ($O(2 \log M)$) 低, 但是它的时间复杂度 ($O(2M^2 \log^2 M)$) 高. 为了充分利用三者的优劣, 有时我们会将三种编码变法进行转换, 文献 [63] 中首先使用 QDAC^[38] 方法将 NEQR 转化成 FRQI, 然后在 FRQI 基础上实现频域滤波.

QRAM 在量子机器学习中被使用^[10]. 此处我们考虑使用 QRAM 方法来降低量子图像编码的时间复杂度. 首先将图像通过 QRAM 存储制备成量子态 $\frac{1}{M^2} \sum_i |i\rangle |\psi_i\rangle$, $|i\rangle$ 表

示位置信息, $|\psi_i\rangle$ 表示像素值信息, 时间复杂度是 $O(2 \log M)$, 然后使用 QADC 方法^[38] 制备量子态 $\frac{1}{\sqrt{\sum_i \psi_i^2}} \sum_i \psi_i |i\rangle$, 时间复杂度是 $O(2 \log M)$, 故总的复杂度是 $O(4 \log M)$.

References

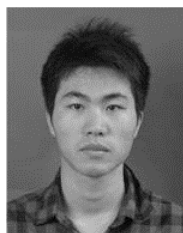
- Gonzalez R C, Woods R E [Author], Ruan Qiu-Qi, Ruan Yu-Zhi [Translator]. *Digital Image Processing* (Third edition). Beijing: Publishing House of Electronics Industry, 2017. (Gonzalez R C, Woods R E [著], 阮秋琦, 阮宇智 [译]. 数字图像处理. 第3版. 北京: 电子工业出版社, 2017.)
- Gai Shan, Bao Zhong-Yun. High noise image denoising algorithm based on deep learning. *Acta Automatica Sinica*, 2020, **46**(12): 2672-2680 (盖杉, 鲍中运. 基于深度学习的高噪声图像去噪算法. 自动化学报, 2020, **46**(12): 2672-2680)
- Shor P W. Algorithms for quantum computation: Discrete logarithms and factoring. In: Proceedings of the 35th Annual Symposium on Foundations of Computer Science. Santa Fe, USA: IEEE, 1994. 124-134
- Grover L K. A fast quantum mechanical algorithm for database search. In: Proceedings of the 28th Annual ACM Symposium on Theory of Computing. Philadelphia, USA: Association for Computing Machinery, 1996. 212-219
- Harrow A W, Hassidim A, Lloyd S. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 2009, **103**(15): Article No. 150502
- Duan Long-Zhen, Xie Xu-Ming, Qiu Tao-Rong, Yang Shu-Qing. A quantum algorithm of the computation of core in rough set. *Acta Automatica Sinica*, 2020, **46**(8): 1753-1758 (段隆振, 谢旭明, 邱桃荣, 杨舒晴. 基于量子计算的粗糙集核属性求解算法. 自动化学报, 2020, **46**(8): 1753-1758)
- Huang H Y, Broughton M, Mohseni M, Babbush R, Boixo S, Neven H, et al. Power of data in quantum machine learning. *Nature Communications*, 2021, **12**: Article No. 2631
- Ye Z K, Li L Z, Situ H Z, Wang Y Y. Quantum speedup of twin support vector machines. *Science China Information Sciences*, 2020, **63**(8): Article No. 189501
- Huang Yi-Ming, Lei Hang, Li Xiao-Yu. A survey on quantum machine learning. *Chinese Journal of Computers*, 2018, **41**(1): 145-163 (黄一鸣, 雷航, 李晓瑜. 量子机器学习算法综述. 计算机学报, 2018, **41**(1): 145-163)
- Biamonte J, Wittek P, Pancotti N, Rebentrost P, Wiebe N, Lloyd S. Quantum machine learning. *Nature*, 2017, **549**(7671): 195-202
- Lomont C. Quantum convolution and quantum correlation algorithms are physically impossible [Online], available: <https://arxiv.org/abs/quant-ph/0309070>, November 5, 2021
- Yan Xi-Xi, Wang Peng-Cheng, Liu Xing-Yun. One-dimensional quantum convolution calculation. *Computer Engineering and Applications*, 2020, **56**(8): 55-59 (闫茜茜, 王鹏程, 刘兴云. 一维量子卷积计算. 计算机工程与应用, 2020, **56**(8): 55-59)
- Yan Xi-Xi, Wang Peng-Cheng, Liu Xing-Yun. Two dimensional quantum convolution calculation. *Journal of Quantum Optics*, 2020, **26**(4): 327-333 (闫茜茜, 王鹏程, 刘兴云. 二维量子卷积计算. 量子光学学报, 2020, **26**(4): 327-333)
- Yuan S Z, Lu Y L, Mao X F, Luo Y, Yuan J G. Improved quantum image filtering in the spatial domain. *International Journal of Theoretical Physics*, 2018, **57**(3): 804-813
- Li P C, Liu X D, Xiao H. Quantum image weighted average filtering in spatial domain. *International Journal of Theoretical*

- Physics*, 2017, **56**(11): 3690–3716
- 16 Yuan S Z, Mao X F, Zhou J, Wang X F. Quantum image filtering in the spatial domain. *International Journal of Theoretical Physics*, 2017, **56**(8): 2495–2511
- 17 Wang Z B, Xu M Z, Zhang Y N. Review of quantum image processing. *Archives of Computational Methods in Engineering*, 2022, **29**(2): 737–761
- 18 Chetia R, Boruah S M B, Sahu P P. Quantum image edge detection using improved Sobel mask based on NEQR. *Quantum Information Processing*, 2021, **20**(1): Article No. 21
- 19 Xu P G, He Z X, Qiu T H, Ma H Y. Quantum image processing algorithm using edge extraction based on Kirsch operator. *Optics Express*, 2020, **28**(9): 12508–12517
- 20 Fan P, Zhou R G, Hu W W, Jing N H. Quantum image edge extraction based on Laplacian operator and zero-cross method. *Quantum Information Processing*, 2019, **18**(1): Article No. 27
- 21 Zhou R G, Liu D Q. Quantum image edge extraction based on improved Sobel operator. *International Journal of Theoretical Physics*, 2019, **58**(9): 2969–2985
- 22 Zhou R G, Yu H, Cheng Y, Li F X. Quantum image edge extraction based on improved Prewitt operator. *Quantum Information Processing*, 2019, **18**(9): Article No. 261
- 23 Fan P, Zhou R G, Hu W W, Jing N H. Quantum image edge extraction based on classical Sobel operator for NEQR. *Quantum Information Processing*, 2019, **18**(1): Article No. 24
- 24 Chetia R, Boruah S M B, Roy S, Sahu P P. Quantum image edge detection based on four directional Sobel operator. In: Proceedings of the 8th International Conference on Pattern Recognition and Machine Intelligence. Tezpur, India: Springer, 2019. 532–540
- 25 Yuan S Z, Venegas-Andraca S W, Wang Y C, Luo Y, Mao X F. Quantum image edge detection algorithm. *International Journal of Theoretical Physics*, 2019, **58**(9): 2823–2833
- 26 Lu Z Y, Wang X M, Shang J Z, Luo Z R, Sun C F, Wu G H. A multimedia image edge extraction algorithm based on flexible representation of quantum. *Multimedia Tools and Applications*, 2019, **78**(17): 24067–24082
- 27 Yao X W, Wang H Y, Liao Z Y, Chen M C, Pan J, Li J, et al. Quantum image processing and its application to edge detection: Theory and experiment. *Physical Review X*, 2017, **7**(3): Article No. 031041
- 28 Zhang Y, Lu K, Gao Y H. QSobel: A novel quantum image edge extraction algorithm. *Science China Information Sciences*, 2015, **58**(1): 1–13
- 29 Hu Wen-Wen, Zhou Ri-Gui, Fan Ping, Li Yao-Chong. Quantum image edge detection based on Canny algorithm. *Journal of Zhengzhou University (Natural Science Edition)*, 2020, **52**(4): 37–52
(胡文文, 周日贵, 范萍, 李尧翀. 基于Canny算法的量子图像边缘检测. 郑州大学学报(理学版), 2020, **52**(4): 37–52)
- 30 Fan P, Zhou R G, Jing N H, Li H S. Geometric transformations of multidimensional color images based on NASS. *Information Sciences*, 2016, **340-341**: 191–208
- 31 Nielsen M A, Chuang I L. *Quantum Computation and Quantum Information*. New York: Cambridge University Press, 2010. 171–211
- 32 Tang Ya-Ru, Cong Shuang, Yang Jing-Bei. On-line state estimation of one-qubit system. *Acta Automatica Sinica*, 2020, **46**(8): 1592–1599
(唐雅茹, 丛爽, 杨靖北. 单量子比特系统状态的在线估计. 自动化学报, 2020, **46**(8): 1592–1599)
- 33 Giovannetti V, Lloyd S, Maccone L. Quantum random access memory. *Physical Review Letters*, 2008, **100**(16): Article No. 160501
- 34 Giovannetti V, Lloyd S, Maccone L. Architectures for a quantum random access memory. *Physical Review A*, 2008, **78**(5): Article No. 052310
- 35 Arunachalam S, Gheorghiu V, Jochym-O'Connor T, Mosca M, Srinivasan P V. On the robustness of bucket brigade quantum RAM. *New Journal of Physics*, 2015, **17**: Article No. 123010
- 36 Kerenidis I, Prakash A. A quantum interior point method for LPs and SDPs. *ACM Transactions on Quantum Computing*, 2020, **1**(1): Article No. 5
- 37 Brassard G, Hoyer P, Mosca M, Tapp A. Quantum amplitude amplification and estimation [Online], available: <https://arxiv.org/abs/quant-ph/0005055v1>, November 5, 2021
- 38 Mitarai K, Kitagawa M, Fujii K. Quantum analog-digital conversion. *Physical Review A*, 2019, **99**(1): Article No. 012301
- 39 Shao C P, Li Y, Li H B. Quantum algorithm design: Techniques and applications. *Journal of Systems Science and Complexity*, 2019, **32**(1): 375–452
- 40 Zhou S S, Wang J B. Efficient quantum circuits for dense circulant and circulant like operators. *Royal Society Open Science*, 2017, **4**(5): Article No. 160906
- 41 Buhrman H, Cleve R, Watrous J, de Wolf R. Quantum fingerprinting. *Physical Review Letters*, 2001, **87**(16): Article No. 167902
- 42 Shao C P. Quantum algorithms to matrix multiplication [Online], available: <https://arxiv.org/abs/1803.01601v2>, November 5, 2021
- 43 McKay D C, Alexander T, Bello L, Biercuk M J, Bishop L, Chen J, et al. Qiskit backend specifications for openQASM and openPulse experiments [Online], available: <https://arxiv.org/abs/1809.03452v1>, November 5, 2021
- 44 Gao Min-Juan, Dang Hong-She, Wei Li-Li, Wang Hai-Long, Zhang Xuan-De. Combining global and local variation for image quality assessment. *Acta Automatica Sinica*, 2020, **46**(12): 2662–2671
(高敏娟, 党宏社, 魏立力, 王海龙, 张选德. 结合全局与局部变化的图像质量评价. 自动化学报, 2020, **46**(12): 2662–2671)
- 45 Wang Dong, Liu Zhi-Hao, Zhu Wan-Ning, Li Shan-Zhi. Design of quantum comparator based on extended general Toffoli gates with multiple targets. *Computer Science*, 2012, **39**(9): 302–306
(王冬, 刘志昊, 朱皖宁, 李善治. 基于多目标扩展通用Toffoli门的量子比较器设计. 计算机科学, 2012, **39**(9): 302–306)
- 46 Xia H Y, Xiao Y F, Song S X, Li H S. Quantum circuit design of approximate median filtering with noise tolerance threshold. *Quantum Information Processing*, 2020, **19**(6): Article No. 183
- 47 Jiang S X, Zhou R G, Hu W W, Li Y C. Improved quantum image median filtering in the spatial domain. *International Journal of Theoretical Physics*, 2019, **58**(7): 2115–2133
- 48 Li P C, Liu X D, Xiao H. Quantum image median filtering in the spatial domain. *Quantum Information Processing*, 2018, **17**(3): Article No. 49
- 49 Zhao Ya, Guo Jia-Hui, Li Pan-Chi. A median filtering scheme for quantum images. *Journal of Electronics & Information Technology*, 2021, **43**(1): 204–211
(赵娅, 郭嘉慧, 李盼池. 一种量子图像的中值滤波方案. 电子与信息学报, 2021, **43**(1): 204–211)
- 50 Ali A E, Abdel-Galil H, Mohamed S. Quantum image mid-point filter. *Quantum Information Processing*, 2020, **19**(8): Article No. 238
- 51 Situ H Z, He Z M, Wang Y Y, Li L Z, Zheng S G. Quantum generative adversarial network for generating discrete distribution. *Information Sciences*, 2020, **538**: 193–208
- 52 Fijany A, Williams C P. Quantum wavelet transforms: Fast algorithms and complete circuits. In: Proceedings of the 1st NASA International Conference on Quantum Computing and Quantum Communications (QCQC). Palm Springs, USA: Springer, 1999.

10–33

- 53 Wang Dong, Liu Qiang, Li Shan-Zhi. Bidirectional reversible logic circuit synthesis algorithm based on matrix elementary transformation. *Computer Science*, 2014, **41**(9): 18–23 (王冬, 刘强, 李善治. 基于矩阵初等变换的量子可逆逻辑电路双向综合算法. *计算机科学*, 2014, **41**(9): 18–23)
- 54 Le P Q, Iliyasa A M, Dong F Y, Hirota K. Strategies for designing geometric transformations on quantum images. *Theoretical Computer Science*, 2011, **412**(15): 1406–1418
- 55 Zhang Y, Lu K, Gao Y H, Wang M. NEQR: A novel enhanced quantum representation of digital images. *Quantum Information Processing*, 2013, **12**(8): 2833–2860
- 56 Cortese J A, Braje T M. Loading classical data into a quantum computer [Online], available: <https://arxiv.org/abs/1803.01958v1>, November 5, 2021
- 57 Li H S, Zhu Q X, Zhou R G, Li M G, Song L, Ian H. Multidimensional color image storage, retrieval, and compression based on quantum amplitudes and phases. *Information Sciences*, 2014, **273**: 212–232
- 58 Le P Q, Dong F Y, Hirota K. A flexible representation of quantum images for polynomial preparation, image compression, and processing operations. *Quantum Information Processing*, 2011, **10**(1): 63–84
- 59 Grover L, Rudolph T. Creating superpositions that correspond to efficiently integrable probability distributions [Online], available: <https://arxiv.org/abs/quant-ph/0208112v1>, November 5, 2021
- 60 Kerenidis I, Prakash A. Quantum recommendation systems. In: Proceedings of the 8th Innovations in Theoretical Computer Science Conference (ITCS). Berkeley, USA: Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2017. 1–21
- 61 Jiang N, Wang J, Mu Y. Quantum image scaling up based on nearest-neighbor interpolation with integer scaling ratio. *Quantum Information Processing*, 2015, **14**(11): 4001–4026
- 62 Li H S, Song S X, Fan P, Peng H L, Xia H Y, Liang Y. Quantum vision representations and multi-dimensional quantum transforms. *Information Sciences*, 2019, **502**: 42–58
- 63 Liu K, Zhang Y, Wang X P, Lu K. A strategy of quantum image filtering in frequency domain. In: Proceedings of the 2016 International Conference on Applied Mechanics, Electronics and Mechatronics Engineering (AMEME). Beijing, China: DEStech,

2016. 82–87



刘兴奥 上海海事大学信息工程学院博士研究生. 主要研究方向为量子图像处理, 量子机器学习.

E-mail: liuxingao@stu.shmtu.edu.cn
(**LIU Xing-Ao** Ph. D. candidate at the School of Information Engineering, Shanghai Maritime University.

His research interest covers quantum image processing, quantum machine learning.)



周日贵 上海海事大学信息工程学院教授. 主要研究方向为图像处理, 计算机视觉与模式识别. 本文通信作者.

E-mail: rgzhou@shmtu.edu.cn
(**ZHOU Ri-Gui** Professor at School of Information Engineering, Shanghai Maritime University. His research interest covers image processing, computer vision and pattern recognition. Corresponding author of this paper.)



郭文宇 上海海事大学信息工程学院博士研究生. 主要研究方向为量子机器学习, 量子计算和量子变分算法.

E-mail: 202040310006@stu.shmtu.edu.cn
(**GUO Wen-Yu** Ph. D. candidate at School of Information Engineering, Shanghai Maritime University. Her research interest covers quantum machine learning, quantum computing and quantum variational algorithm.)