Tec Empresarial
P-ISSN: 1659-2395; E-ISSN: 1659-3359

FEATURE EMBEDDING IN QUANTUM DISCRETE TRANSFORM FOR OBJECT DETECTION AND CLASSIFICATION USING SENSOR DATA

[1]Sandip Vaijanath Kendre, [2]Anand Singh Rajawat, [3]Amol Potgantwar
[1]Ph.D Research scholar , School of Computer Science & Engineering & Technology, Sandip University, Nashik, Maharashtra , India
[2,3]Professor , School of Computer Science & Engineering & Technology, Sandip University, Nashik, Maharashtra , India
[1] sandipkendre@gmail.com, [2]anandsingh.rajawat@sandipuniversity.edu.in [3] amol.potgantwar@sitrc.org.

Abstract: Object detection and classification in the data collected from the sensor networks is challenging task. This paper presents novel mechanisms Quantum Fourier Transform (QFT) and Quantum Discrete Transform (QDT) which can be utilized for object detection and classification using sensor data. The quantum Fourier transform with its mathematical modelling and example is described in this paper. The QFT is utilized to propose quantum Fourier transform sampler algorithm. The mathematical elaboration with example is elaborated. An overview of proposed quantum discrete transform with proposed quantum discrete transform algorithm, and proposed designed quantum circuit for quantum discrete transform are described in this paper. The proposed quantum Fourier transform sampler algorithm is evaluated using the open source Qiskit platform. The evaluation of the algorithm is also extended to the real quantum hardware IBM Quantum by utilizing the IBM quantum API. For the evaluation of the proposed approach on the real quantum hardware the inverse QFT is utilized.
Keywords: Object Detection, Quantum Computing, Quantum Transforms, Quantum Fourier Transforms, Quantum Discrete Transforms

1 Introduction

The application of quantum technology for remote sensing has been considered for at least last 20 years. An active imaging information transmission technology for satellite-borne quantum remote sensing is proposed in past, providing solutions and technical basis for realizing active imaging technology relying on quantum mechanics principles. Quantum technology is also used in interferometric synthetic aperture radars. A residue connection problem in the phase unwrapping procedure as quadratic unconstrained binary optimization problem, can be solved by using the D-Wave quantum annealer. A quantum annealer application has been explored in past for subset feature selection and the classification of hyperspectral images. In this chapter quantum discrete transform is proposed and analyzed which can be used for real-time object detection in distinct fields.

Sensor data has consistently leveraged technological and computational advances helping in developing novel techniques to characterize and model the human environment [Rodriguez-Donaire et al., 2020; Sudmanns et al., 2019]. Given that many remote sensing missions are currently operative, carrying on board multispectral, hyperspectral, and radar sensors, and the improved capabilities in transmitting and saving a continuously increasing volume of sensor data, nowadays estimated in over 150 terabytes per day, the amount of data from sensor applications have reached impressive volumes so that they are referred to as Big Data. At the same time, advances in computational technologies and analysis methodologies have also progressed to accommodate larger and higher resolution datasets. Data classification techniques are constantly being improved to keep up with the ever expanding stream of Big Data, and as a consequence, artificial intelligence (AI) techniques are becoming increasingly necessary tools.

Given the need to help expand the processing techniques to deal with these high-resolution Big Data, sensor data processing is now looking toward new and innovative computation technologies [Riedel et al., 2021]. This is where quantum computing (QC) will play a fundamental role. Today, there is a number of differing quantum devices, such as programmable superconducting processors, quantum annealers, and photonic quantum computers. However, QC still presents some technological limitations, as reported by [Shettell et al., 2021] with a special concern with noise and limited error correction. Specific algorithms, namely, the noisy intermediate-scale quantum (NISQ) computing algorithms, have been designed to tackle these issues.

Quantum computers promise to efficiently solve important problems that are intractable on a conventional computer. For instance, in quantum systems, due to the exponentially growing physical dimensions, finding the eigen values of certain operators is one such intractable problem, which can be solved by combining a highly reconfigurable photonic quantum processor with a conventional computer.

Another example is the case of the variational quantum eigensolver (VQE) algorithm used to solve combinatorial optimization problems such as finding the ground state energy of a molecule. The algorithm finds a bound to the lowest eigenenergy of a given Hamiltonian. This is, in essence, a kind of cost function, which is defined by the expectation of the molecular Hamiltonian of a given prepared eigenstate. The goal of the VQE is to minimize this cost function by varying the parameters θ used to prepare the ansatz eigenstate often representative of a molecule. This hybrid algorithm prepares and determines eigenenergies through quantum circuits, and then, it varies the parameter classically. By iterating through these classical variations and quantum calculations, a hybrid minimization process is established. This approximation of critical minima is analogous to the gradient descent.

In QC, a qubit or quantum bit is the basic unit of quantum information, i.e., the quantum version of the classic binary bit. A qubit is one of the simplest quantum systems that display the peculiarity of quantum mechanics. Indeed, it is a two-state quantum mechanical system, e.g., an electron in two possible levels (spin up and spin down) or a single photon in one of the two possible states (vertical and horizontal polarization).While in a classical system, a bit can be in one state or the other, qubit exists in a coherent superposition of both states simultaneously, a property that is

fundamental to quantum mechanics. Quantum computers utilize the principles of superposition and entanglement to streamline computation. For every n qubits, 2^n possible states can be represented. This is an exponential improvement with respect to the classical systems, which can only represent n states for every n bits. Moreover, quantum systems exist in a high-dimensional space, known as a Hilbert space, whose inherent properties lend themselves to a complex linear optimization.

A Surrogate Assisted Quantum-behaved Algorithm is presented in [Islam et al., 2022] to obtain a better solution for the well placement optimization problem. The proposed approach utilizes different meta-heuristic optimization techniques such as the Quantum-inspired Particle Swarm Optimization and the Quantum-behaved Bat Algorithm in different implementation phases. Previous work demonstrates that quantum-based techniques, such as the quantum bat algorithm (QBA) and quantum particle swarm optimization algorithm (QPSO) performed better for well placement optimization [Ross, 2019]. A Fourier expression of the quantum radar cross section (QRCS) of a dihedral corner reflector is proposed in [Tian et al., 2021], which avoids the problems caused by atomic sampling on the target-object surface and is a powerful tool for analyzing the scattering characteristics of the reflector. The Radon transform, a classical image-processing tool for fast tomography and denoising is extended to the quantum computing platform [Ma et al., 2022]. A new kind of periodic discrete Radon transform (PDRT), called the quantum periodic discrete Radon transform (QPRT), is proposed.

Quantum machine learning (QML) is an active field of research that seeks to take advantage of the capabilities of both quantum computers and machine learning techniques, adapting the latter to the strengths of the current state of the art in quantum computing. There are many examples that illustrate how quantum computing can be used for anomaly detection [Herr et al., 2021], to train models [Abdelgaber and Nikolopoulos, 2020], and possibly enhance machine learning models, such as quantum support vector machines (QSVMs) [Havlíˇcek et al., 2019], quantum classifiers (QCs) [Yano et al., 2020], and quantum neural networks [Abbas et al., 2021]. Much work has been conducted on synthetic and publicly available datasets from various domains, such as drug discovery [Batra et al., 2021], image classification [Kerenidis and Luongo, 2020], and computational sciences. Comparisons have been made to the classical counterparts of the available QML algorithms [Stein et al., 2021]. In addition, when synthetic data are used for machine learning experiments, there have been provable advantages shown involving synthetic datasets when there is a lack of necessary data [Abufadda and Mansour, 2021].

With the vigorous development of emerging sensor technologies the huge volume of data gets generated every day. Identification of the intended objects from this huge volume of data is tedious task. The comparative analysis of all these reviewed literature is given in table 4.1.

2 Quantum Fourier Transform

The Fourier transform occurs in many different versions throughout classical computing, in areas ranging from signal processing to data compression to complexity theory. The quantum Fourier transform (QFT) is the quantum implementation of the discrete Fourier transform over the

Table 1: Comparison of various methods.

Year	Author's	Methodology	Datasets	Purpose	Evaluation Strategy
2009	Zhao et al.	Data transformations within the	3D Sensor Data and Videos	Motion trajectory of each moving	Computation time
2017	Islam et al.	Surrogate Assisted Quantum-behaved Algorithm, Quantum-inspired	Complex Reservoir	Well placement optimization	Average standard deviation, Effectiveness, Efficiency
2019	Lv et al.	Machine learning and dedicated-	3D sensors data and	Position corrections, a local map,	Computation time
2022	Sebastianelli et al.	Hybrid Quantum	EuroSAT dataset	Remote Sensing	Precision, Recall, F1 Score
2022	Ma et al.	Quantum periodic	Images	Denoising and fast line	Computation time
2022	Otsu et al.	Spatial-Importance-	3D Sensor Data, KITTI	Real-Time Object	Computation time

amplitudes of a wave function. It is part of many quantum algorithms, most notably Shor's factoring algorithm and quantum phase estimation. The discrete Fourier transform acts on a vector $(x_0, x_1, \ldots\ldots, x_{N-1})$ and maps it to the vector $(y_0, y_1, \ldots\ldots, y_{N-1})$ according to the formula

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j \omega_N^{jk}$$

Where, $\omega_N^{jk} = e^{\frac{2\pi jk}{N}}$

On the similar ground, the quantum Fourier transform acts on a quantum state $|X\rangle = \sum x_j |j\rangle$ and maps it to the quantum state $|Y\rangle = \sum x_k |k\rangle$ according to the formula

$$|j\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega_N^{jk} |k\rangle$$

The QFT is also represented by using the unitary matrix:

$$\mathrm{QFT}_U = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \omega_N^{jk} |k\rangle \langle j|$$

The quantum Fourier transform (QFT) transforms between two bases, the computational (Z) basis,

and the Fourier basis. The H-gate is the single-qubit QFT, and it transforms between the Z-basis states $|0\rangle$ and $\langle 1|$ to the X-basis states $|+\rangle$ and $\langle -|$. In the same way, all multi-qubit states in the computational basis have corresponding states in the Fourier basis. The QFT is simply the function that transforms between these bases.

$$|\mathrm{State}_{\mathrm{in}_{\mathrm{Computational}}}(\mathrm{Time})_{\mathrm{Basis}}\rangle \overrightarrow{\mathrm{QFT}} |\mathrm{State}_{\mathrm{in}_{\mathrm{Fourier}}}(\mathrm{Frequency})_{\mathrm{Basis}}\rangle \mathrm{QFT}|x\rangle = |\tilde{x}\rangle$$

In the computational basis, numbers in binary are stored using the states $|0\rangle$ and $\langle 1|$ as shown in figure 1. The frequency with which the different qubits change; the leftmost qubit flips with every increment in the number, the next with every 2 increments, the third with every 4 increments, and so on. In the Fourier basis, numbers are stored using different rotations around the Z-axis as shown in figure 2.

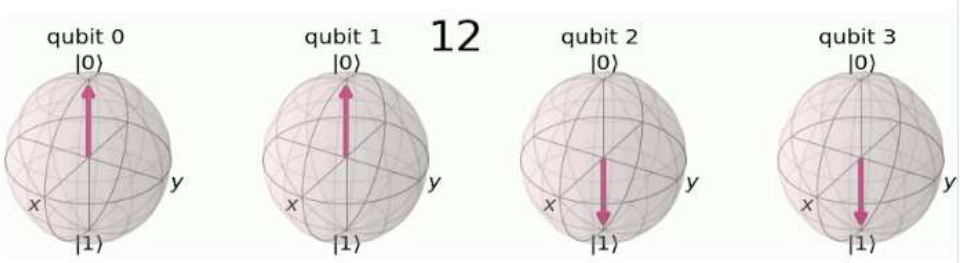

Figure 1: Computational Basis.

The number to be stored dictates the angle at which each qubit is rotated around the Z-axis. In the state $|\tilde{0}\rangle$, all qubits are in the state $|+\rangle$. To encode the state $|\tilde{12}\rangle$on 4 qubits, rotate the leftmost qubit by $\frac{12}{2^n} = \frac{12}{16}$ full turns ($\frac{12}{16} \times 2\pi$radians). The next qubit is turned double this ($\frac{24}{16} \times 2\pi$radians, or $\frac{24}{16}$ full turns), this angle is then doubled for the qubit after, and so on. The frequency with which

each qubit changes. The leftmost qubit (qubit 0) in this case has the lowest frequency, and the rightmost the highest.

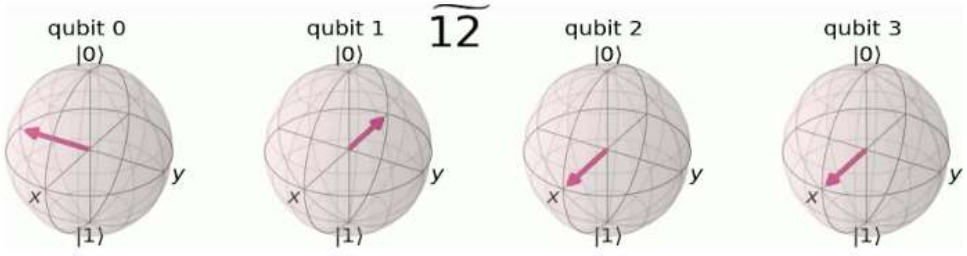

Figure 2: Fourier Basis.

The QFT operator as defined above acts on a single qubit state$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. If $x_0 = \alpha, x_1 = \beta$,andN $= 2$,then

$$y_0 = \frac{1}{\sqrt{2}}\left(\alpha e^{\left(2\pi i\frac{0\times 0}{2}\right)} + \beta e^{\left(2\pi i\frac{1\times 0}{2}\right)}\right) = \frac{1}{\sqrt{2}}(\alpha + \beta)$$

$$y_1 = \frac{1}{\sqrt{2}}\left(\alpha e^{\left(2\pi i\frac{0\times 1}{2}\right)} + \beta e^{\left(2\pi i\frac{1\times 1}{2}\right)}\right) = \frac{1}{\sqrt{2}}(\alpha - \beta)$$

$$\text{QFT}_U|\psi\rangle = \frac{1}{\sqrt{2}}(\alpha + \beta)|0\rangle + \frac{1}{\sqrt{2}}(\alpha - \beta)|1\rangle$$

This operation is exactly the result of applying the Hadamard operator (HAD) on the qubit:

$$\text{HAD} = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}$$

If HADoperator is applied to the state$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, the new state obtained is

$$\frac{1}{\sqrt{2}}(\alpha + \beta)|0\rangle + \frac{1}{\sqrt{2}}(\alpha - \beta)|1\rangle = \tilde{\alpha}|0\rangle + \tilde{\beta}|1\rangle$$

Hadamard gate performs the discrete Fourier transform for $N = 2$,on the amplitudes of the state.

3 Proposed Quantum Fourier Transform Sampler Algorithm

A transformation for $N = 2^2$, QFT_Nacting on the state $|x\rangle = |x_1x_2.....x_n\rangle$ where $x_1$is the most significant bit, is shown below:

$$\mathrm{QFT}_{N|x\rangle} = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} \omega_N^{\mathrm{xy}} |y\rangle$$

$$= \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{\frac{2\pi \mathrm{ixy}}{2^n}} |y\rangle \text{ ,as} \omega_N^{\mathrm{xy}} = e^{\frac{2\pi \mathrm{ixy}}{N}} \text{ and} \mathrm{N} = 2^n$$

Rewriting in fractional binary notation $y = y_1 y_2 \ldots\ldots\ldots y_n, \frac{y}{2^n} = \sum_{k=1}^{n} \frac{y_k}{2^k},$

$$\mathrm{QFT}_{N|x\rangle} = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{2\pi i \left(\sum \frac{y_k}{2^k}\right) x} |y_1 y_2 \ldots\ldots y_n\rangle$$

After expanding the exponential of a sum to a product of exponentials

$$\mathrm{QFT}_{N|x\rangle} = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} \prod_{k=1}^{n} e^{\frac{2\pi \mathrm{ixy}_k}{2^k}} |y_1 y_2 \ldots\ldots y_n\rangle$$

After expanding $\sum_{y=0}^{N-1} = \sum_{y_1=0}^{N-1} \sum_{y_2=0}^{N-1} \sum_{y_3=0}^{N-1} \ldots\ldots\ldots\ldots \sum_{y_n=0}^{N-1}$ and rearranging sum and products

$$\mathrm{QFT}_{N|x\rangle} = \frac{1}{\sqrt{N}} \bigotimes_{k=1}^{n} \left(|0\rangle + e^{\frac{2\pi \mathrm{ix}}{2^k}} |1\rangle \right)$$

$$= \frac{1}{\sqrt{N}} \left(|0\rangle + e^{\frac{2\pi \mathrm{ix}}{2}} |1\rangle \right) \otimes \left(|0\rangle + e^{\frac{2\pi \mathrm{ix}}{2^2}} |1\rangle \right) \otimes \ldots\ldots\ldots \otimes \left(|0\rangle + e^{\frac{2\pi \mathrm{ix}}{2^{n-1}}} |1\rangle \right) \otimes \left(|0\rangle + e^{\frac{2\pi \mathrm{ix}}{2^n}} |1\rangle \right)$$

The circuit that implements QFT makes use of two gates. The first one is a single-qubit Hadamard gate, HADon the single-qubit state $|x_k\rangle$is

$$\mathrm{HAD}|x_k\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle + e^{\left(\frac{2\pi \mathrm{ix}_k}{2^k}\right)} |1\rangle \right)$$

The second is a two-qubit controlled rotation C_{ROT_k} given in block-diagonal form as

$$C_{\mathrm{ROT}_k} = \begin{bmatrix} I & 0 \\ 0 & U_{\mathrm{ROT}_k} \end{bmatrix}$$

With

$$U_{\mathrm{ROT}_k} = \begin{bmatrix} 1 & 0 \\ 0 & e^{\left(\frac{2\pi}{2^k}\right)} \end{bmatrix}$$

The action of C_{ROT_k} on a two-qubit state $|x_i x_j\rangle$ where the first qubit is the control and the second is the target is given by

$$C_{\mathrm{ROT}_k}|0x_j\rangle = |0x_j\rangle$$

$$C_{\mathrm{ROT}_k}|1x_j\rangle = e^{\left(\frac{2\pi \mathrm{i} x_j}{2^k}\right)}|1x_j\rangle$$

Given these two gates, a circuit that implements an n-qubit QFT is shown in figure 3.

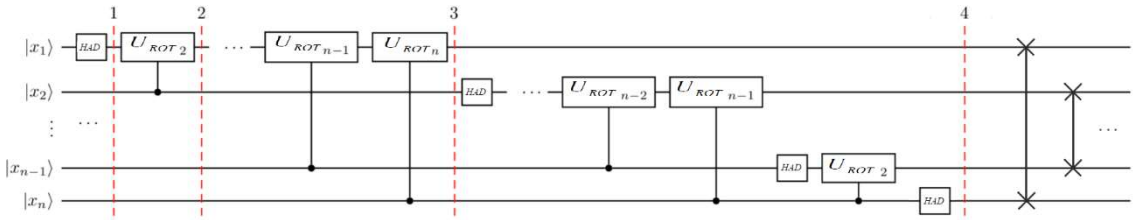

Figure 3: Quantum Circuit for n-qubit Quantum Fourier Transform.

The circuit operates as specified below. Starting with an n-qubit input state $|x_1 x_2 \ldots\ldots x_n\rangle$.

1. After the first Hadamard gate on qubit 1, the state is transformed from the input state to

$$\mathrm{HAD}_1|x_1 x_2 \ldots\ldots x_n\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle + e^{(\pi \mathrm{i} x_1)}|1\rangle\right] \otimes |x_2 x_3 \ldots\ldots x_n\rangle$$

2. After the U_{ROT_2} gate on qubit 1 controlled by qubit 2, the state is transformed to

$$\frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x_2}{2} + \pi \mathrm{i} x_1\right)}|1\rangle\right] \otimes |x_2 x_3 \ldots\ldots x_n\rangle$$

3. After the application of the last U_{ROT_n} gate on qubit 1 controlled by qubit n, the state becomes

$$\frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x_n}{2^{n-1}} + \frac{\pi \mathrm{i} x_{n-1}}{2^{n-2}} + \ldots\ldots\ldots\ldots + \frac{\pi \mathrm{i} x_2}{2} + \pi \mathrm{i} x_1\right)}|1\rangle\right] \otimes |x_2 x_3 \ldots\ldots x_n\rangle$$

With $x = 2^{n-1}x_1 + 2^{n-2}x_2 + \ldots\ldots + 2^1 x_{n-1} + 2^0 x_n$

The state can be modified as

$$\frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x}{2^{n-1}}\right)}|1\rangle\right] \otimes |x_2 x_3 \ldots\ldots x_n\rangle$$

4. After the application of a similar sequence of gates for qubits 2^n, we find the final state to be

$$\frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x}{2^{n-1}}\right)}|1\rangle\right] \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x}{2^{n-2}}\right)}|1\rangle\right] \otimes \ldots\ldots\ldots\ldots \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x}{2^1}\right)}|1\rangle\right]$$
$$\otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x}{2^0}\right)}|1\rangle\right]$$

which is exactly the QFT of the input state as derived with the caveat that the order of the qubits is reversed in the output state.

The 4-qubit QFT circuit $|y_4y_3y_2y_1\rangle = \mathrm{QFT}_8|x_4x_3x_2x_1\rangle$is created as follows:

1. Apply a Hadamard gate to $|x_1\rangle$

$$|\psi_1\rangle = |x_4\rangle \otimes |x_3\rangle \otimes |x_2\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{(\pi \mathrm{i} x_1)}|1\rangle\right]$$

2. Apply a U_{ROT_2}gate to $|x_1\rangle$ depending on $|x_2\rangle$

$$|\psi_2\rangle = |x_4\rangle \otimes |x_3\rangle \otimes |x_2\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x_2}{2}+\pi \mathrm{i} x_1\right)}|1\rangle\right]$$

3. Apply a U_{ROT_3}gate to $|x_1\rangle$ depending on $|x_3\rangle$

$$|\psi_3\rangle = |x_4\rangle \otimes |x_3\rangle \otimes |x_2\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x_3}{2^2}+\frac{\pi \mathrm{i} x_2}{2}+\pi \mathrm{i} x_1\right)}|1\rangle\right]$$

4. Apply a U_{ROT_4}gate to $|x_1\rangle$ depending on $|x_4\rangle$

$$|\psi_4\rangle = |x_4\rangle \otimes |x_3\rangle \otimes |x_2\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x_4}{2^3}+\frac{\pi \mathrm{i} x_3}{2^2}+\frac{\pi \mathrm{i} x_2}{2}+\pi \mathrm{i} x_1\right)}|1\rangle\right]$$

5. Apply a Hadamard gate to $|x_2\rangle$

$$|\psi_5\rangle = |x_4\rangle \otimes |x_3\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{(\pi \mathrm{i} x_2)}|1\rangle\right] \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x_4}{2^3}+\frac{\pi \mathrm{i} x_3}{2^2}+\frac{\pi \mathrm{i} x_2}{2}+\pi \mathrm{i} x_1\right)}|1\rangle\right]$$

6. Apply a U_{ROT_2}gate to $|x_2\rangle$depending on $|x_4\rangle$

$$|\psi_6\rangle = |x_4\rangle \otimes |x_3\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x_4}{2^2}+\frac{\pi \mathrm{i} x_3}{2}+\pi \mathrm{i} x_2\right)}|1\rangle\right] \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x_4}{2^3}+\frac{\pi \mathrm{i} x_3}{2^2}+\frac{\pi \mathrm{i} x_2}{2}+\pi \mathrm{i} x_1\right)}|1\rangle\right]$$

7. Apply a Hadamard gate to $|x_4\rangle$

$$|\psi_7\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle + e^{(\pi \mathrm{i} x_4)}|1\rangle\right] \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x_4}{2}+\pi \mathrm{i} x_3\right)}|1\rangle\right] \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x_4}{2^2}+\frac{\pi \mathrm{i} x_3}{2}+\pi \mathrm{i} x_2\right)}|1\rangle\right] \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{\left(\frac{\pi \mathrm{i} x_4}{2^3}+\frac{\pi \mathrm{i} x_3}{2^2}+\frac{\pi \mathrm{i} x_2}{2}+\pi \mathrm{i} x_1\right)}|1\rangle\right]$$

8. The reverse order of the output state relative to the desired QFT can be observed. Therefore, the order of the qubits must be reversed (in this case swap y_1 and y_4).

A useful form of the QFT for $N = 2^n$is shown above. It can be observed that only the last qubit depends on the values of all the other input qubits and each further bit depends less and less on the input qubits. This becomes important in physical implementations of the QFT, where nearest-neighbor couplings are easier to achieve than distant couplings between qubits. Additionally, as the QFT circuit becomes large, an increasing amount of time is spent doing increasingly slight rotations. It turns out that we can ignore rotations below a certain threshold and still get decent results, this is known as the approximate QFT. This is also important in physical implementations,

as reducing the number of operations can greatly reduce decoherence and potential gate errors. The proposed QFT sampler algorithm for object detection in sensor data is specified below.

Algorithm: QFT Sampler Algorithm

1. Build a circuit that implements the QFT with the qubits upside down, then swap them afterwards
2. if n is 0:
    a. return circuit #Exit function if circuit is empty
3. n -= 1 # Start indexes from 0
4. circuit_ApplyH(n) # Apply the H-gate to the most significant qubit
5. for qubit in range(n):
    a. circuit_ControlledRotation(pi/2**(n-qubit), qubit, n) # For each less significant qubit, apply smaller-angled controlled rotation
6. Perform qft on the first n qubits in circuit without swaps
    a. if n is 0:
        i. return circuit
    b. n -= 1
    c. circuit_ApplyH(n)
    d. for qubit in range(n):
        i. circuit_ControlledRotation(pi/2**(n-qubit), qubit, n)
    e. qft_rotations(circuit, n)
7. QFT on the first n qubits in circuit
    a. qft_rotations(circuit, n)
    b. swap_registers(circuit, n)
    c. return circuit
8. Create a QFT circuit of the correct size
    a. qft_circuit = qft(QuantumCircuit(n), n)
9. Take the inverse of this circuit
    a. invqft_circuit = qft_circuit_inverse()
10. Add it the first n qubits in existing circuit
    a. Circuit_append(invqft_circuit, circuit_qubits[:n])
    b. return circuit_decompose()to see the individual gates

4 An Overview of Proposed Quantum Discrete Transform

The controlled phase rotation gate $C_{U1}$to perform the Quantum Discrete Transform, which performs a phase rotation if both qubits are in the $|11\rangle|11\rangle$ state. The matrix looks the same regardless of whether the most significant bit (MSB) or least significant bit (LSB) is the control qubit as discussed in previous section,

$$C_{U1}(\lambda) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\lambda} \end{pmatrix}$$

The algorithm for the proposed quantum discrete transform is specified below:

Algorithm: Proposed Quantum Discrete Transform

Usages: Qiskit, QuantumCircuit, ClassicalRegister, QuantumRegister, Aer, IBMQ, available_backends, execute, register, get_backend, circuit_drawer, state_fidelity, least_busy, matplotlib, pyplot, numpy, math, plot_histogram, plot_state

1. Use state tomography functions
2. Design Quantum Circuit for Quantum Discrete Transform
    a. Number of qubits: N # e.g. N=10
    b. Quantum Register with N qubits: Q # e.g. QuantumRegister(N, 'Q')
    c. Classical Register with N Classical Bits for Measurement: C # e.g. ClassicalRegister(N, "C")
    d. Build a Quantum Circuit: QCkt #QuantumCircuit(Q, C)
3. Use Different Gates before the Controlled Phase Gate
    a. QCkt.X(Q[0])
    b. QCkt.Y(Q[1])
    c. QCkt.Z(Q[2])
    d. QCkt.S(Q[3])
    e. QCkt.SDG(Q[4])
    f. QCkt.T(Q[5])
    g. QCkt.barrier([])
4. To every qubit add Hadamard Gate
    a. for i in range(N):
        i. QCkt.H(Q[i])
        ii. Use Controlled-Phase Gate cu1 and Entangle i-th qubit with [i+1, n]-th qubit
        iii. For k in range (i+1, N):
            1. QCkt.cu1((2*pi)/(2**(2+k-i-1)),Q[i], Q[j])
5. Gate Swap
    a. for i in range(N):
        i. if i <= (N/2) - 1:
            1. QCkt.swap(Q[i], Q[N-1-i])
        ii. QCkt.barrier()
6. Measure
    a. QCkt.measure(Q, C)
    b. circuit_drawer(QCkt)

7. Evaluate the QCkt for Different Simulation Parameters

The designed quantum circuit for evaluating proposed quantum discrete transform is shown in figure 4.

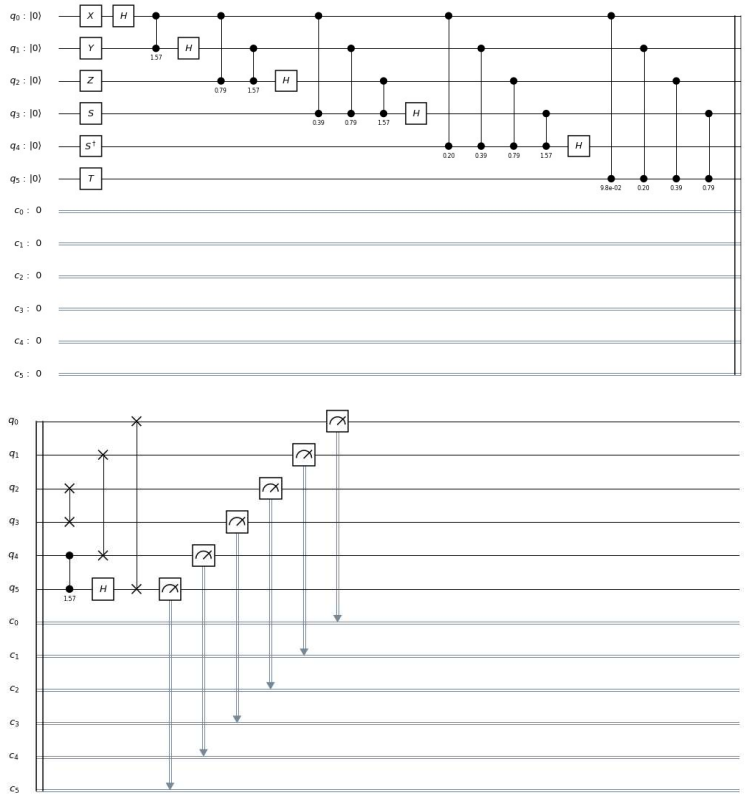

Figure 4: Proposed Designed Quantum Circuit for Quantum Discrete Transform.

5 Results and Evaluation of the Proposed Approach

The proposed approach is simulated on the Darwin OS with 10 CPUs and 32 GB Memory having Python 3.9, and Python Clang compiler. The open source Qiskit library shown in figure 5, https://qiskit.org/ is utilized for the implementation for the proposed approach. The modules available in this library qiskit-terra, qiskit-aer, qiskit-ignis, qiskit-ibmq-provider, and qiskit-aqua are utilized for experimental evaluation. Qiskit is an open-source toolkit for useful quantum computing. It has a production-ready circuit compiler.

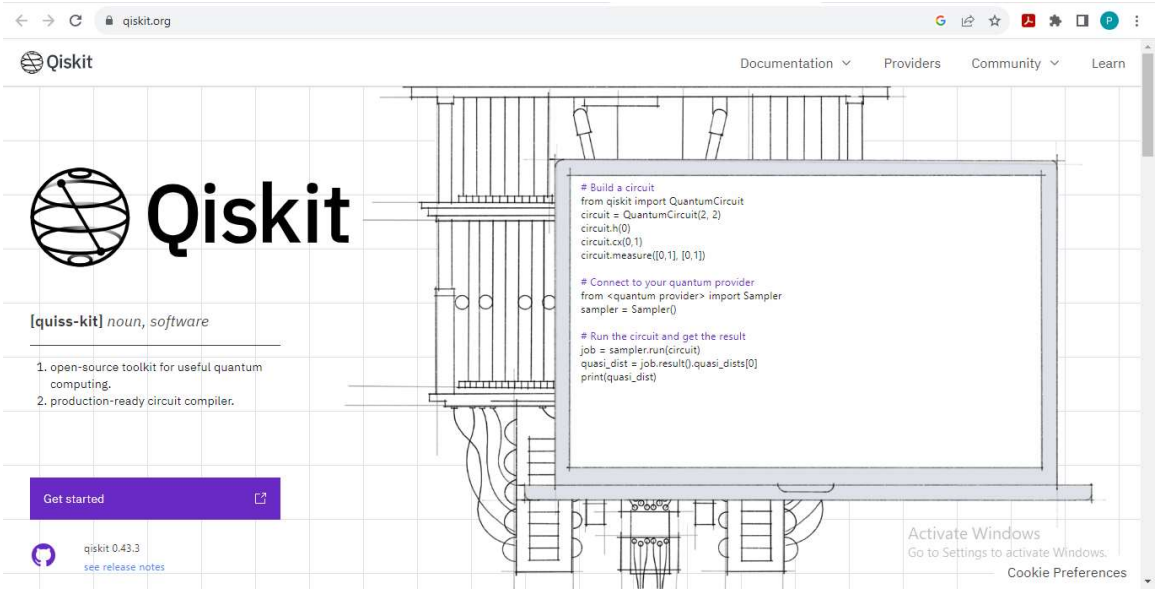

Figure 5: Qiskit used for the implementation of the proposed approach.

In Qiskit, the implementation of the C_{ROT} gate is a controlled phase rotation gate is defined as:

$$\mathrm{CP}(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\theta} \end{bmatrix}$$

The mapping from the C_{ROT_k} gate into the$\mathrm{CP}(\theta)$ gate is found from the following equation:

$$\theta = \frac{2\pi}{2^k} = \frac{\pi}{2^{k-1}}$$

Initially the 3-qubit quantum circuit is generated. Qiskit's least significant bit has the lowest index (0), thus the circuit will be mirrored through the horizontal. First, H-gate is applied to qubit 2, the result is shown in figure 6. To turn an extra quarter turn if qubit 1 is in the state $|1\rangle$, shown in figure 7. Another eighth turn if the least significant qubit (0) is $|1\rangle$ shown in figure 8.

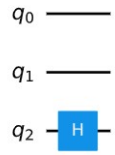

Figure 6: H-gate applied to qubit 2.

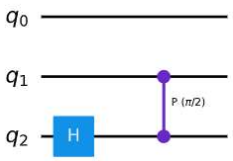

Figure 7: Qubit 1 is in the state$|1\rangle$.

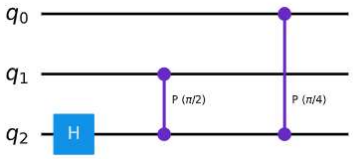

Figure 8: Least significant qubit (0) is|1⟩.

With that qubit taken care of, it can be ignored and the process is repeated, using the same logic for qubits 0 and 1, shown in figure 9. The qubits 0 and 2 are swapped to complete the QFT shown in figure 10.

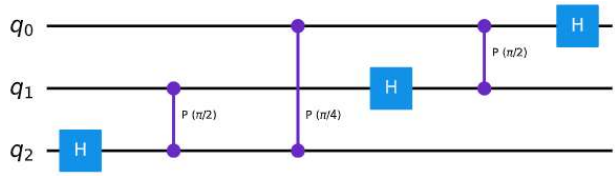

Figure 9: Quantum circuit after repeating the process, logic for qubits 0 and 1.

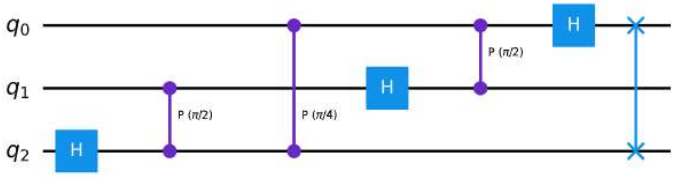

Figure 10: Qubits 0 and 2 swapped.

A general circuit is created for the QFT in Qiskit. Qiskit allows creating large general circuits like this. It is easier to build a circuit that implements the QFT with the qubits upside down, then swap them afterwards; the function is created that rotates qubits correctly. The most significant qubit (the qubit with the highest index) are correctly rotated as shown in figure 11. The widget can be used to see how this circuit scales with the number of qubits in circuit.

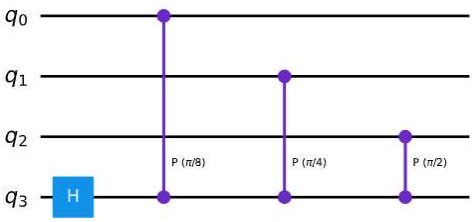

Figure 11: Most significant qubit rotated.

This is the first part of our QFT. The most significant qubit is correctly rotated, correctly rotate the second most significant qubit. Then deal with the third most significant, and so on. At the end of qft_rotations() function, the same code is used to repeat the process on the next n-1 qubits as shown in figure 12.

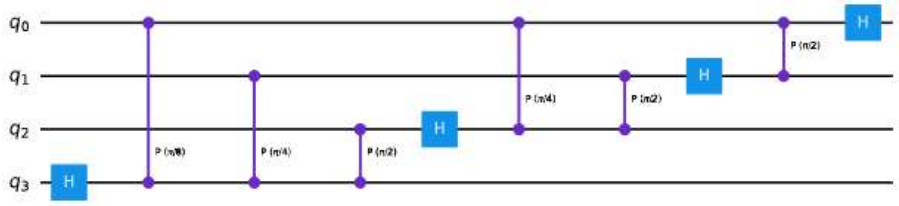

Figure 12: All most significant qubit rotated.

The swaps are added at the end of the QFT function to match the definition of the QFT. The generalized circuit shown in figure 13 for the quantum Fourier transform is generated. For verifying the circuit working properly or not, a number 5 (101 in binary) is encoded in the

computational basis. The encoded bits are shown in figure 14. The qubit's states are verified using the aer simulator shown in figure 15.

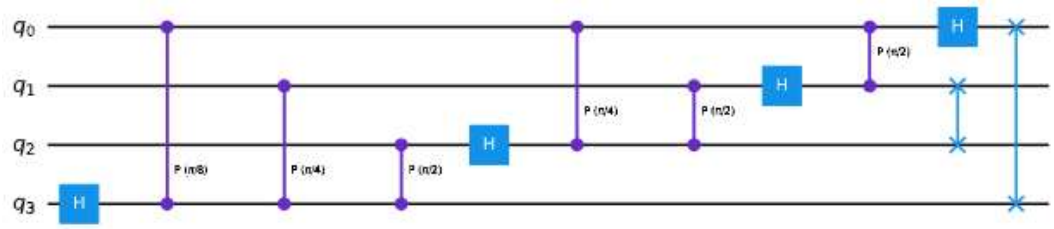

Figure 13: Generalized circuit for QFT.

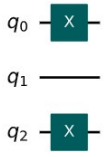

Figure 14: Generalized circuit for encoded bits.

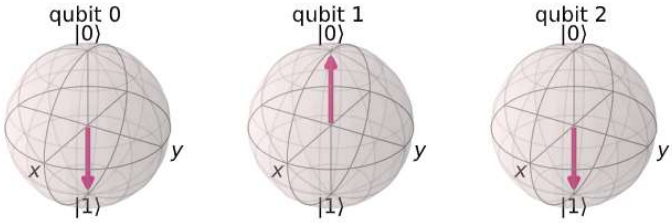

Figure 15: Qubit's states on the aer simulator.

The proposed sampled QFT is applied, as shown in figure 16 and the final state of qubits is analyzed, shown in figure 17.

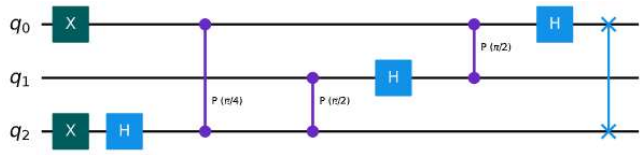

Figure 16: QFT circuit for encoding 5.

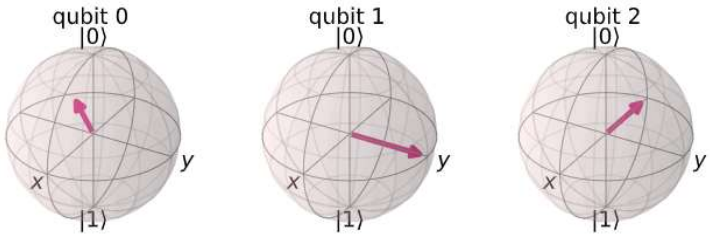

Figure 17: Final state of qubits.

It can be observed that the proposed sampler QFT algorithm has worked correctly. Compared the state $|\tilde{0}\rangle = |{+}{+}{+}\rangle$, Qubit 0 has been rotated by $\frac{5}{8}$ of a full turn, qubit 1 by $\frac{10}{8}$ full turns (equivalent to $\frac{1}{4}$of a full turn), and qubit 2 by $\frac{20}{8}$full turns (equivalent to $\frac{1}{2}$of a full turn).

When the circuit is evaluated on a real device, the results would be completely random, since all qubits are in equal superposition of $|0\rangle$and $|1\rangle$. To demonstrate and investigate the QFT working on real hardware, instead of creating the state as specified above, the QFT is executed in reverse,

and the output is verified, it is the state $|\tilde{5}\rangle$ as expected. Firstly, Qiskit is used to easily reverse QFT operation. Qbits are kept in state $|\tilde{5}\rangle$as shown in figure 18. The circuit after application of the inverse QFT is shown in figure 19. The probabilities for the encoding of the digits from 0 to 7 from computational space to quantum space is shown in figure 20. The highest probability outcome is for the digit 5.

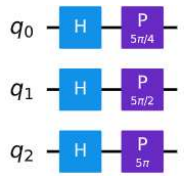

Figure 18: Qbits in state$|\tilde{5}\rangle$.

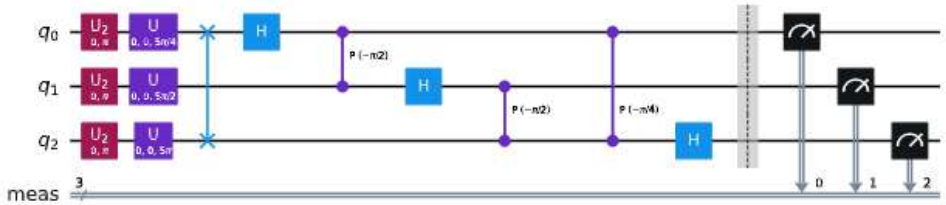

Figure 19: Quantum circuit after inverse QFT.

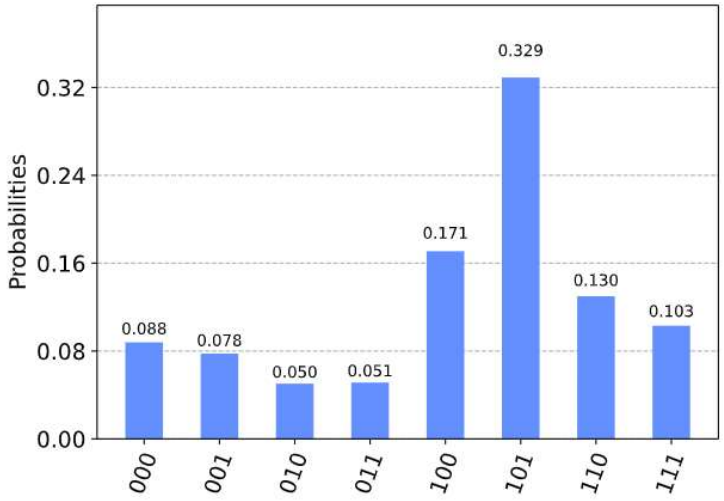

Figure 20: Probabilities of encoding the digits.

The implementation of the proposed QDT algorithm is carried out in Anaconda Python 3.8 distribution with Intel i3 11th generation processor, 8 GB RAM and Windows 11 OS. The python libraries and functions used to implement the proposed algorithm are: qiskit, IBMQ: IBM Quantum System One, Aer, Quantum Circuits, ClassicalRegister, QuantumRegister, available_backends, execute, get_backend, circuit_drawer, state_fidelity, least_busy, plot_histogram, plot_state, numpy, matplotlib, and math. The simulation results are shown in the screenshot shown in figure 21.

```
COMPLETED
{'000000': 131, '000001': 142, '000010': 127, '000011': 127, '000100': 128, '000101': 154, '000110': 129, '000111': 117, '00100
0': 108, '001001': 119, '001010': 135, '001011': 131, '001100': 131, '001101': 138, '001110': 141, '001111': 119, '010000': 12
8, '010001': 122, '010010': 131, '010011': 141, '010100': 134, '010101': 143, '010110': 123, '010111': 129, '011000': 118, '011
001': 136, '011010': 127, '011011': 130, '011100': 135, '011101': 123, '011110': 108, '011111': 132, '100000': 133, '100001': 1
33, '100010': 149, '100011': 126, '100100': 123, '100101': 128, '100110': 134, '100111': 116, '101000': 132, '101001': 122, '10
1010': 120, '101011': 135, '101100': 140, '101101': 111, '101110': 125, '101111': 119, '110000': 124, '110001': 118, '110010':
140, '110011': 137, '110100': 134, '110101': 128, '110110': 136, '110111': 125, '111000': 114, '111001': 125, '111010': 137, '1
11011': 141, '111100': 117, '111101': 98, '111110': 114, '111111': 121}
```

Figure 21: Simulation results.

The relationship between superposition and counts is shown in screenshot of figure 22.

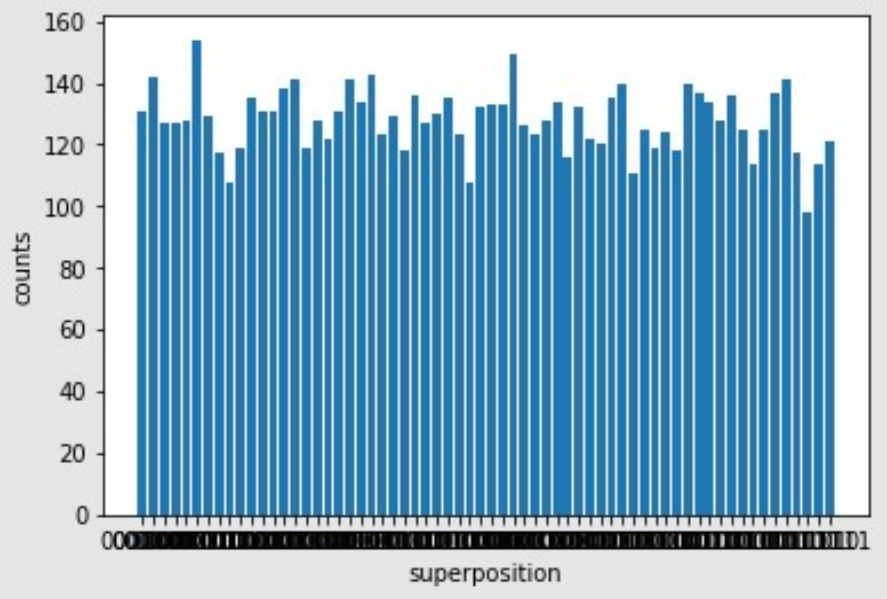

Figure 22: The distribution of superposition and counts.

The real machine used for quantum computation is shown in following screenshot shown in figure 23.

```
Available backends:

[<IBMQBackend('ibmqx4') from IBMQ()>,
 <IBMQBackend('ibmqx5') from IBMQ()>,
 <IBMQBackend('ibmqx2') from IBMQ()>,
 <IBMQBackend('ibmq_16_melbourne') from IBMQ()>,
 <IBMQBackend('ibmq_qasm_simulator') from IBMQ()>]
```

Figure 23: Used Real Machines as Backend.

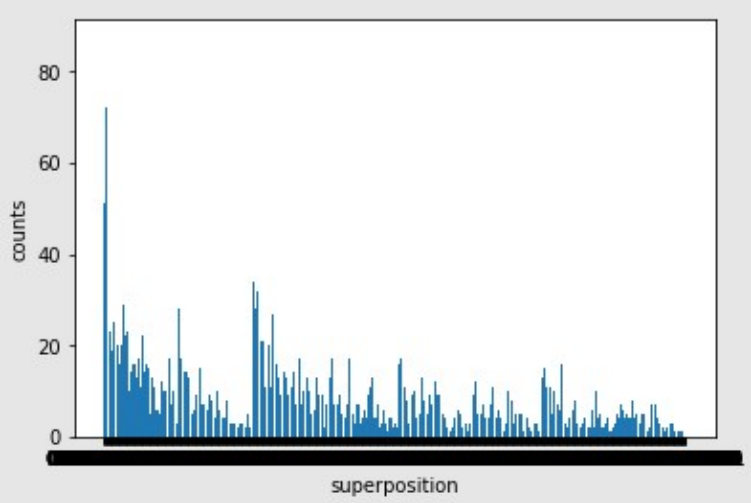

Figure 24: 10 qubits and same initial state.

Maximum number of shots considered to run the experiment are 8192. The maximum number of credits spent on executions are 3. After evaluation the ibmq_16_melbourne backend is identified as the best backend. For 10 qubits and same initial state the results are shown in figure 24. For 6 qubits and same initial state the results are shown in figure 25. For 6 qubits and differential initial state 1.0 results are shown in figure 26. For 6 qubits and differential initial state 2.0 results are shown in figure 27.

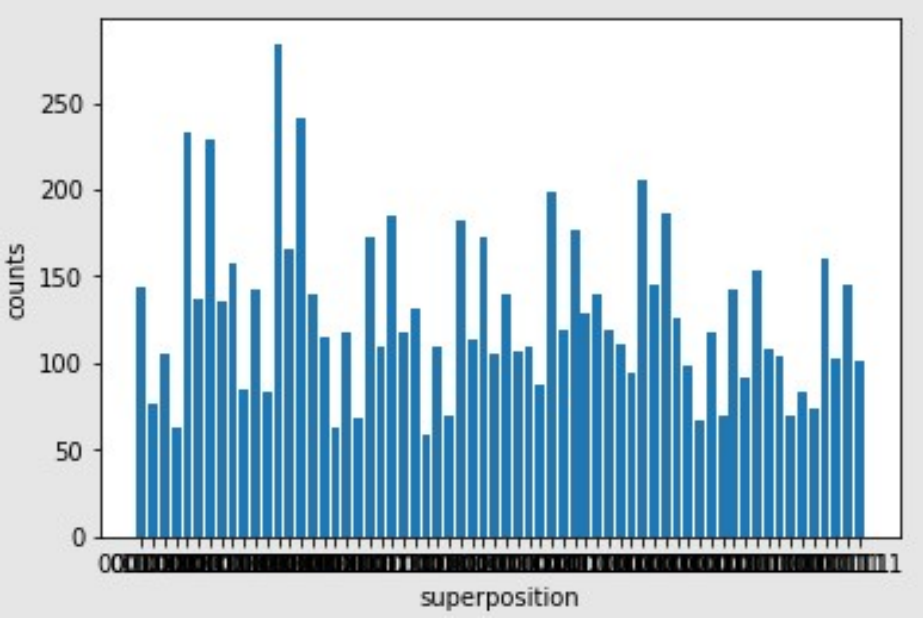

Figure 25: 6 qubits and same initial state.

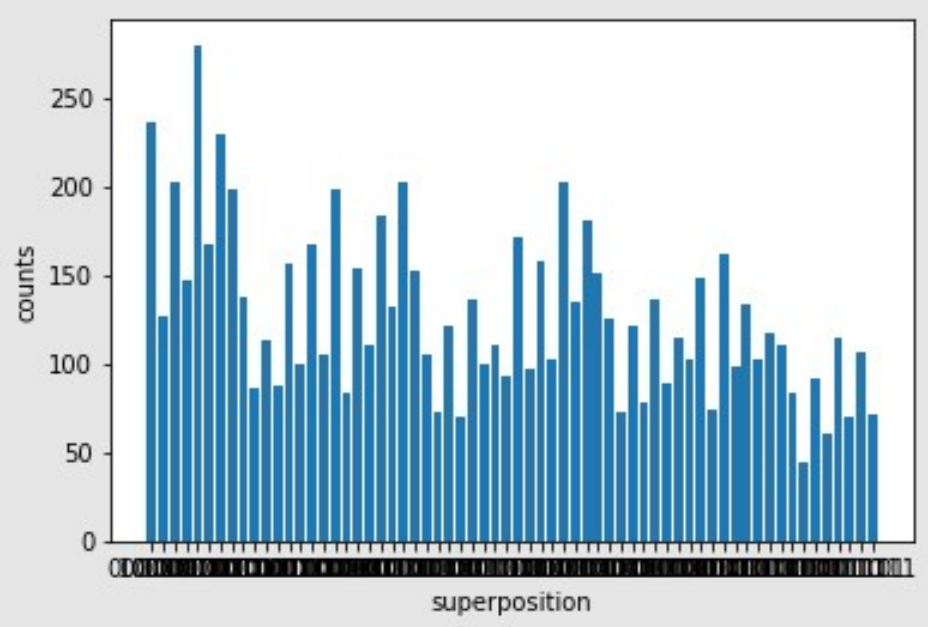

Figure 26: 6 qubits and differential initial state 1.0.

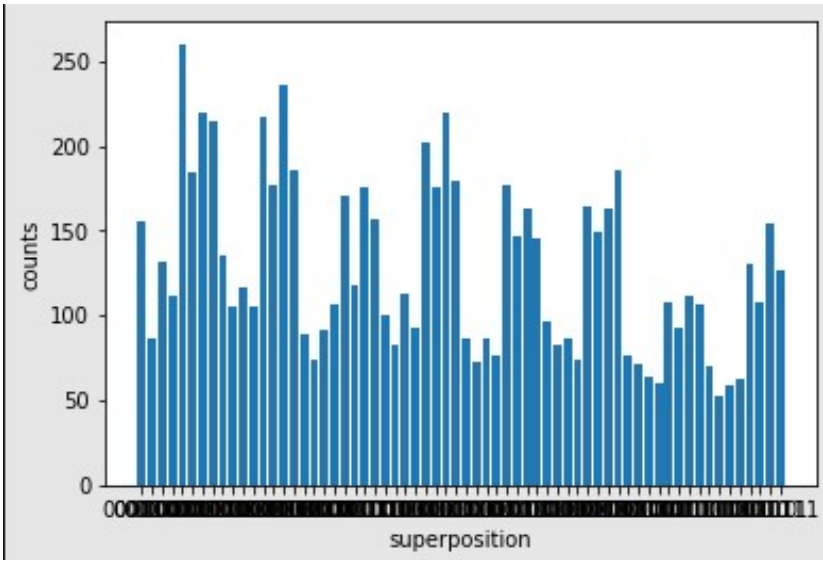

Figure 27: 6 qubits and differential initial state 2.0.

The quantum computation is less uniform if more qubits are taken. It also produces more errors. The simulations carried out are uniformly distributed. In the real quantum machine hardware uniform distribution is major concern. A uniform distribution out of the designed quantum circuit is obtained as proper qubits has been initialized and linear combination of Hadamard and control phase shift is applied which rotates states along a plane on the sphere that has always the values of 0 or 1. If the qubits are initialized with different values, the uniform distribution will get minimized. The proposed QDT is utilized in next chapter with sensor data fusion integration for object detection for sensor data classification

6 Conclusion

The quantum Fourier transform with its mathematical modelling and example is described in this paper. The QFT is utilized to propose quantum Fourier transform sampler algorithm. The mathematical elaboration with example is elaborated in this paper. The proposed quantum Fourier transform sampler algorithm is evaluated using the open source Qiskit platform. The evaluation of the algorithm is also extended to the real quantum hardware IBM Quantum by utilizing the IBM quantum API. For the evaluation of the proposed approach on the real quantum hardware the inverse QFT is utilized. The proposed approach is evaluated for the encoding of the digits 0 to 7. The highest probability outcome is for the digit 5. The proposed quantum Fourier transform sampler algorithm can be extended for object detection and for proposing the quantum discrete transform for object detection for sensor data classification. An overview of proposed quantum discrete transform with proposed quantum discrete transform algorithm, and proposed designed quantum circuit for quantum discrete transform is presented in this paper. The implementation details and results evaluation are presented in this paper. Regarding the quantum component, this work is aim at increasing the proportion of quantum processing in the hybrid approach. Indeed, more complex quantum mechanisms are expected to enhance the object detection capabilities. In particular, quantum computing could be examined to incorporate spatial information and invariance in the processing. More fundamentally, the understanding of the quantum discrete transforms will represent the key to design better models, develop deep quantum learning, and eventually implement it to many real-life applications. The proposed QDT can be utilized in future with sensor data fusion integration for object detection for sensor data classification.

Conflict of Interest

On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

[1] [Rodriguez-Donaire et al., 2020] S. Rodriguez-Donaire et al., “Earth observation technologies: Low-end market disruptive innovation,” in Satellites Missions and Technologies for Geosciences. Rijeka, Croatia: InTech, Jan. 2020.

[2] [Sudmanns et al., 2019] Sudmanns M. et al. (2019), Big earth data: Disruptive changes in earth observation data management and analysis. Int. J. Digit. Earth, vol. 13, no. 7, pp. 832–850, Jul. 2019.

[3] [Riedel et al., 2021] M. Riedel, G. Cavallaro, and J. Benediktsson, “Practice and experience in using parallel and scalable machine learning in remote sensing from HPC over cloud to quantum computing,” in Proc. IEEE Int. Geosci. Remote Sens. Symp., 2021, pp. 1571–1574.

[4] [Shettell et al., 2021] N. Shettell, W. J. Munro, D. Markham, and K. Nemoto, "Practical limits of error correction for quantum metrology," New J. Phys., vol. 23, no. 4, 2021, Art. no. 043038.

[5] [Islam et al., 2022] Jahedul Islam, Amril Nazir, Md. Moinul Hossain, Hitmi Khalifa Alhitmi, Muhammad Ashad Kabir, and Abdul-Halim Jallad, "A Surrogate Assisted Quantum-behaved Algorithm for Well Placement Optimization," IEEE Access, 2022,

[6] [Ross, 2019] O. H. M. Ross, "A review of quantum-inspired meta-heuristics: going from classical computers to real quantum computers," IEEE Access, 2019.

[7] [Tian et al., 2021] Zhifu Tian, Di Wu, and Tao Hu, "Fourier Expression of the Quantum Radar Cross Section of a Dihedral Corner Reflector," IEEE PHOTONICS JOURNAL, VOL. 13, NO. 4, AUGUST 2021.

[8] [Ma et al., 2022] GUANGSHENG MA, HONGBO LI, AND JIMAN ZHAO, "Quantum Radon Transforms and Their Applications," IEEE Transactions on Quantum Engineering, VOLUME 3, 2022.

[9] [Herr et al., 2021] D. Herr, B. Obert, and M. Rosenkranz, "Anomaly detection with variational quantum generative adversarial networks," Quantum Sci. Technol., vol. 6, no. 4, Jul. 2021, Art. no. 045004.

[10] [Abdelgaber and Nikolopoulos, 2020] N. Abdelgaber and C. Nikolopoulos, "Overview on quantum computing and its applications in artificial intelligence," in Proc. IEEE 3rd Int. Conf. Artif. Intell. Knowl. Eng., 2020, pp. 198–199.

[11] [Havlíˇcek et al., 2019] V. Havlíˇcek et al., "Supervised learning with quantum-enhanced feature spaces," Nature, vol. 567, no. 7747, pp. 209–212, 2019.

[12] [Yano et al., 2020] H. Yano, Y. Suzuki, R. Raymond, and N. Yamamoto, "Efficient discrete feature encoding for variational quantum classifier," in Proc. IEEE Int. Conf. Quantum Comput. Eng., 2020, pp. 11–21.

[13] [Abbas et al., 2021] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, "The power of quantum neural networks," Nature Comput. Sci., vol. 1, no. 6, pp. 403–409, Jun. 2021.

[14] [Batra et al., 2021] K. Batra et al., "Quantum machine learning algorithms for drug discovery applications," J. Chem. Inf. Model., vol. 61, no. 6, pp. 2641–2647, 2021, doi: 10.1021/acs.jcim.1c00166.

[15] [Kerenidis and Luongo, 2020] I. Kerenidis and A. Luongo, "Classification of the MNIST data set with quantum slow feature analysis," Phys. Rev. A, vol. 101, no. 6, 2020, Art. no. 062327, doi: 10.1103/PhysRevA.101.062327.

[16] [Stein et al., 2021] S. A. Stein et al., "A hybrid system for learning classical data in quantum states," in Proc. IEEE Int. Perform., Comput., Commun. Conf., 2021, pp. 1–7, doi: 10.1109/IPCCC51483.2021.9679430.

[17] [Abufadda and Mansour, 2021] M. Abufadda and K. Mansour, "A survey of synthetic data generation for machine learning," in Proc. 22nd Int. Arab Conf. Inf. Technol., 2021, pp. 1–7.

[18] [Zhao et al., 2009] Zhao H., Cui J., Zha H., Katabira K., Shao X., and Shibasaki R. (2009), Sensing an intersection using a network of laser scanners and video cameras. IEEE Intell. Transp. Syst. Mag., vol. 1, no. 2, pp. 31-37, Sep. 2009.

[19] [Islam et al., 2017] Islam J., Nazir A., Hossain Md. M., Alhitmi H. K., Kabir M. A., and Jallad A. (2017), A Surrogate Assisted Quantum-behaved Algorithm for Well Placement Optimization. IEEE Access, 2017.

[20] [Lv et al., 2019] Lv B., Xu H., Wu J., Tian Y., Zhang Y., Zheng Y., Yuan C., and Tian S. (2019), LiDAR-enhanced connected infrastructures sensing and broadcasting high-resolution traffic information serving smart cities. IEEE Access, vol. 7, pp. 79895-79907, 2019.

[21] [Sebastianelli et al., 2022] Sebastianelli A., Zaidenberg D. A., Spiller D., Saux B. L., and Ullo S. L. (2022), On Circuit-Based Hybrid Quantum Neural Networks for Remote Sensing Imagery Classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, Vol. 15, pp. 565- 580, 2022.

[22] [Ma et al., 2022] Ma G., Li H., and Zhao J. (2022), Quantum Radon Transforms and Their Applications, IEEE Transactions on Quantum Engineering, Volume 3, 2022.

[23] [Otsu et al., 2022] RYO OTSU, RYOICHI SHINKUMA, TAKEHIRO SATO, EIJI OKI, DAIKI HASEGAWA, AND TOSHIKAZU FURUYA, “Spatial-Importance-Based Computation Scheme for Real-Time Object Detection From 3D Sensor Data,” IEEE Access, VOLUME 10, pp. 5672- 5680, 2022.