Performance Analysis and Noise Impact of a Novel Quantum KNN Algorithm for Machine Learning

Asif Akhtab Ronggon
Department of Electrical and Electronic Engineering
Bangladesh University of Engineering and Technology
Dhaka, Bangladesh
asifaftab172@gmail.com

Md. Saifur Rahman
Department of Electrical and Electronic Engineering
Bangladesh University of Engineering and Technology
Dhaka, Bangladesh
saifur@eee.buet.ac.bd

May 13, 2025

Abstract

This paper presents a novel quantum K-nearest neighbors (QKNN) algorithm, which offers improved performance over the classical k-NN technique by incorporating quantum computing (QC) techniques to enhance classification accuracy, scalability, and robustness. The proposed modifications focus on optimizing quantum data encoding using Hadamard and rotation gates, ensuring more effective rendering of classical data in quantum states. In addition, the quantum feature extraction process is significantly enhanced by the use of entangled gates such as IsingXY and CNOT, which enables better feature interactions and class separability. A novel quantum distance metric, based on the swap test, is introduced to calculate similarity measures between various quantum states, offering superior accuracy and computational efficiency compared to traditional Euclidean distance metrics. We assess the achievement of the proposed QKNN algorithm on three benchmark datasets: Wisconsin Breast Cancer, Iris, and Bank Note Authentication, and have noted its superior performance relative to both classical k-NN (CKNN) and Quantum Neural Network (QNN). The proposed QKNN algorithm is found to achieve prediction accuracies of 98.25%, 100%, and 99.27% ,respectively, for the three datasets, while the customized QNN shows prediction accuracies of only 97.17%, 83.33%, and 86.18%, respectively. Furthermore, we address the challenges of quantum noise by incorporating a repetition encoding-based error mitigation strategy, which ensures the stability and resilience of the algorithm in noisy quantum environments. The results highlight the potential of the proposed QKNN as a scalable, efficient and robust quantum-enhanced machine learning algorithm, especially in high-dimensional and complex datasets, when traditional approaches frequently fail.

Keywords: Quantum machine learning, quantum K-nearest neighbors, quantum distance computation, quantum feature mapping, quantum noise and error correction, classical KNN, hybrid classical-quantum systems, and quantum neural networks.

1 Introduction

The rapid advancements in QC have spurred significant interest in the development of algorithms that possess the potential of outperforming their classical counterparts across various domains, including machine learning (ML). Quantum Machine Learning (QML) is a highly prospective field of research that seeks to integrate quantum computational principles with traditional ML models to improve their computational efficiency and scalability. Significant attention has been drawn to quantum implementations of acknowledged classical algorithms, including QKNN, owing to their rapid processing of high-dimensional data and more computational efficiency in comparison to their classically implemented equivalents [1].

The CKNN is a broadly used non-parametric ML approach that involves data mining and recognizing patterns. However, CKNN has significant computing hurdles, particularly for large datasets, despite being a simple and efficient algorithm. This approach is computationally expensive for real-time applications because it relies on brute-force computation to find the closest neighbors [2]. It is necessary to find other ways to accelerate the classification process since the performance of CKNN drops drastically as the size of the dataset grows. One possible answer to these problems is QC, which can perform exponentially faster than its conventional counterpart. According to Wiebe [3], this is the cause for the introduction of QKNN, a quantum variation, as a substitute. To enhance the effectiveness of neighbor search and distance computation, QKNN utilizes quantum features such as superposition, entanglement, and quantum parallelism. To be more specific, compared to their conventional counterparts, quantum algorithms may conduct computationally more effective similarity measurements, and quantum systems can encode enormous quantities of information [4]. For high-dimensional classification problems, QKNN is a very desired option since it uses quantum circuits to offer exponential speed-ups compared to the traditional CKNN in certain applications [5].

The present status of quantum technology continues to impede the practical implementation of QKNN, despite its theoretical advantages. Full realization of QML algorithms is not possible owing to decoherence, noise and the constrained qubit count of the recently available quantum hardware. Furthermore, developing faster quantum feature mapping techniques is essential, since efficient quantum feature mapping remains a fundamental challenge. To decide whether the proposed QKNN is practicable, it is crucial to consider its computational complexity, quantum noise resilience and classification excellence, particularly when dealing with practical quantum faults [6]. The major contributions of the current research are as follows:

- A novel QKNN algorithm is introduced, leveraging quantum characteristics to enhance classification effectiveness.
- A QNN architecture has also been developed as part of the research. The accuracy of QKNN is compared with both the developed QNN architecture and CKNN algorithm through rigorous testing on various benchmarking datasets.
- It has also been investigated as to how quantum noise affects the accuracy of QKNN classification and suggests strategies for reducing the effect to make it more resilient to noise.

Here is how the rest of the paper is organized. A thorough literature review of QML, QKNN, and QNN is included in Section II. The description of dataset, algorithmic specifics, and experimental setup is presented in Section III. Classification accuracy, computational efficiency, and noise resistance were compared between the traditional CKNN and QKNN in Section IV, which also covers the findings of the performance assessment. Section V includes the conclusions and findings of this research and suggests further research direction in QML.

2 Related Work

QML is a novel concept that optimizes traditional ML algorithms by using quantum physics. New QC capabilities have demonstrated that quantum algorithms may solve computational problems unique to conventional methods, including data classification, feature selection, and model optimization. The versatility and effectiveness with which QKNN and QNNs categorize large datasets have brought them to the forefront among the existing QML algorithms.

Recent years have noticed an upsurge in interest in potential integration of QC principles with ML algorithms, particularly QKNN and QNN. For large, high-dimensional datasets and multi-classification jobs, the objective of the algorithms influenced by quantum mechanics is to increase computing efficiency while simultaneously improving the classification accuracy. Despite the encouraging findings demonstrated by quantum techniques, there are still several scaling problems and realistic application challenges that persist, particularly in relation to quantum noise and decoherence [7].

An essential part of QML is quantum feature encoding, which maps classical input into quantum states. Several encoding methods that have been examined for the purpose of effectively representing classical information in quantum systems include amplitude encoding, angle encoding, and quantum random access coding (QRAC) [8] [9]. Nguyen et al. [10] conducted a review of quantum visual encoding strategies and identified the Quantum Information Gap (QIG), which results in information loss between the classical and quantum features. The Quantum Information Preserving (QIP) loss function was put forward to reduce this gap, thereby enhancing the effectiveness of QML algorithms. Sakka et al. [11] presented an agentic framework capable of autonomously generating, evaluating, and refining quantum feature maps through the utilization of large language models. This method revealed that the algorithm was worthy of identifying feature maps that surpassed current quantum baselines and attained comparable accuracy relative to classical kernels on datasets, such as MNIST.

Numerous studies have investigated quantum KNN algorithms, demonstrating potential improvements in CKNN methods. Jarir and Quafafou [12] proposed a hybrid quantum-classical KNN approach for text classification, which utilizes quantum algorithms for feature extraction, although their study did not provide direct accuracy comparisons with classical methods. Zardini et al. [13] introduced a quantum KNN algorithm based on Euclidean distance estimation, improving computational efficiency in high-dimensional spaces. However, the study lacked explicit accuracy benchmarks. Li et al. [14] introduce a novel quantum K-nearest neighbors (QKNN) classifier that simultaneously quantumizes both neighbor selection and K value determination by formulating an objective function combining least squares loss and sparse regularization using the HHL algorithm. The methodology employs a quantum circuit using phase estimation, controlled rotation, and inverse phase estimation, with experimental validation via Qiskit and MATLAB demonstrating its effectiveness in optimizing K and neighbor selection.

In addition to KNN, QNN has become a viable strategy for enhancing classification tasks. Quantum-enhanced neural networks leverage QC to offer significant improvements in both speed and accuracy. Berti et al. [15] explore the transformation of classical K-nearest neighbor algorithms into quantum equivalents, focusing on two distinct designs: amplitude encoding and basis encoding. It highlights the impact of these encoding methods on algorithm structure, distance metrics, and performance while addressing challenges in data preparation and the theoretical advantages of quantum algorithms over classical ones. In their study, Wang et al. [16] applied quantum KNN to handwritten digit recognition, achieving 98% accuracy on the MNIST dataset, which outperformed CKNN algorithms achieving an accuracy of around 94%.

Further developments in QML have seen significant progress in QNNs. Bhaskaran and Prasanna [17] performed an accuracy analysis comparing classical and quantum-enhanced KNN algorithms, reporting accuracy improvements of up to 18% in favor of quantum-enhanced methods. Feng et al. [18] proposed an enhanced quantum KNN classification algorithm using quantum polar distance, which demonstrated significant improvements in classification accuracy, especially for high-dimensional data. Xiang et al. [19] explored hybrid quantum-classical convolutional neural networks (QCNNs) for breast cancer diagnosis, showcasing improvements in both accuracy and computational efficiency over the classical models. Zhul et al. [20] propose a fast clustering approach that integrates graph-regularized non-negative matrix factorization with quantum clustering to leverage the strengths of both, enhancing speed and accuracy through a novel quantum-inspired clustering algorithm combining matrix factorization and quantum computing principles.

In summary, while many studies have focused on improving the computational efficiency and scalability of quantum KNN and QNN algorithms, some of which have also demonstrated notable improvements in classification accuracy. However, direct and consistent accuracy comparisons between quantum-enhanced and classical methods across various domains remain an area for further research.

3 Methodology

This section outlines the methodology used to evaluate the performance of the proposed QKNN algorithm. We focus on data preprocessing, implementation of the QKNN algorithm, and comparison with CKNN and QNN. In addition, we discuss the impact of quantum noise on the classification accuracy and present strategies for error mitigation.

3.1 Dataset Description

For evaluating the proposed QKNN algorithm, three benchmark datasets are utilized: Wisconsin Breast Cancer, Iris, and Bank Note Authentication.

Wisconsin Breast Cancer Dataset: This dataset contains 569 instances, each described by 30 features derived from digitized images of fine needle aspirates (FNA) of breast masses. The task involves binary classification to differentiate between benign and malignant tumors [21].

Iris Dataset: Introduced by Fisher [22], the Iris dataset consists of 150 instances characterized by four features (sepal length, petal length, sepal width, and petal width). The classification task distinguishes between three species of iris flowers: versicolor, setosa, and virginica. Its ease of use and neatly divided classes make it a standard benchmark for classification tasks.

Bank Note Authentication Dataset: This dataset comprises 1,372 instances, each with 4 features extracted from images of genuine and forged banknotes. The goal is a binary classification task to distinguish between authentic and counterfeit notes [23].

An overview of the prominent attributes of these datasets is given in Table 1.

Table 1: Summary of Dataset Characteristics.

Dataset	Instances	Features	Classes	Reference
Wisconsin Breast Cancer	569	30	2	[21]
Iris	150	4	3	[22]
Bank Note Authentication	1372	4	2	[23]

3.2 Feature Selection Using the Chi-Square Test

Feature selection is a crucial step in reducing dimensionality, improving model efficiency, and preventing overfitting. In this study, we apply the chi-square (χ^2) test to assess the dependency between each feature and the target class.

The chi-square statistic for each feature F_i is calculated as:

$$\chi^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i} \tag{1}$$

where O_i is the observed frequency, E_i is the expected frequency under the null hypothesis of independence, and k is the number of bins or categories.

For continuous features, discretization is applied to convert them into categorical values, with expected frequencies calculated relying on the notion that there is statistical independence between the feature and the target class.

A significant result (typically with a p-value less than 0.05) specifies that the feature is relevant to the target class and should be retained. Features with high p-values, indicating weak association, are discarded to reduce model complexity and improve generalization [24]. This method has been successfully used in numerous ML applications to select relevant features and enhance model performance [25].

3.3 Quantum Machine Learning

QML is an emerging paradigm that integrates QC principles with ML algorithms. The fundamental advantage of QML arises from quantum phenomena such as superposition, entanglement, and quantum parallelism, which enable certain computations to be performed more efficiently than classical counterparts [6]. These quantum properties provide the potential for QML to outperform classical models, particularly in tasks involving high-dimensional space, optimization, and large-scale data processing.

One of the key elements of QML is quantum data encoding, where classical data is mapped onto quantum states. Unlike classical representations, quantum encoding allows data to exist in a superposition of multiple states simultaneously, enabling parallel computation. This property is further enhanced by quantum feature maps, which transform classical inputs into quantum Hilbert spaces, allowing for more expressive data processing compared to classical kernels [4].

A popular approach in QML involves Variational Quantum Circuits (VQCs), which use quantum gates parameterized by tunable variables. These gates are optimized to learn patterns from data. Hybrid quantum-classical models combine QC with classical optimization techniques, showing promising results in tasks such as classification, clustering, and generative modeling [26]. These hybrid models are particularly useful given the current limitations of noisy intermediate-scale quantum (NISQ) devices, which suffer from limited qubit coherence and gate fidelity.

QML is also advancing in quantum kernel methods and quantum support vector machines (QSVMs). Quantum-enhanced kernels facilitate higher-dimensional feature mappings that are often intractable classically, providing the ability to handle more complex data structures [4]. Additionally, quantum Boltzmann machines and quantum generative adversarial networks (QGANs) have been explored for generative learning tasks such as image synthesis and anomaly detection [27].

Despite its promise, QML faces several challenges. For example, current quantum hardware is limited by factors such as qubit count, error rates, and coherence times, which restrict the practical implementation of large-scale quantum models. Furthermore, efficient quantum-specific optimization techniques and quantum circuit architectures remain active areas of research. One significant bottleneck is quantum data loading, where encoding large classical datasets into quantum states is a nontrivial task [28].

The integration of QC into ML holds the potential for significant advancements in both theoretical foundations and practical applications. As quantum hardware improves, QML is expected to lead to breakthroughs in fields such as drug discovery, materials science, and financial modeling, where complex simulations require computational power beyond classical capabilities [1]. Ongoing research in quantum algorithms, noise mitigation, and quantum-classical hybrid frameworks will be crucial in unlocking the full potential of QML.

In conclusion, QML represents a transformative approach to ML, with the potential to address computational bottlenecks that classical systems struggle with. As both experimental and theoretical developments progress, QML may become a powerful tool for solving problems that were previously considered intractable.

3.4 Classical vs. Quantum Feature Spaces

In classical ML, data is represented as feature vectors in an n-dimensional Euclidean space given by Eqn. (2) as follows:

$$\mathbf{x} = (x_1, x_2, ..., x_n) \in \mathbb{R}^n. \tag{2}$$

Algorithms, like SVMs, rely on kernel methods to map data into higher-dimensional spaces, allowing separation of nonlinearly separable data. However, this mapping increases computational complexity [29].

In contrast, QC encodes data into quantum states via superposition and entanglement. A classical data point $\mathbf{x}$ is mapped to a quantum state represented by Eqn. (3) as follows:

$$|\psi_x\rangle = U(x)\,|0\rangle\,, \tag{3}$$

where $U(x)$ is a unitary operator and $|0\rangle$ represents the quantum state [4]. Quantum feature spaces inherently embed data into exponentially larger spaces, enhancing the ability of the model to capture complex patterns and relationships that are difficult to represent classically.

Quantum parallelism allows for the efficient exploration of high-dimensional spaces, making QML particularly suitable for large, complex datasets. As shown in Figure 1, quantum encoding techniques map data into exponentially large quantum Hilbert spaces, offering a more powerful feature representation compared to the classical methods.

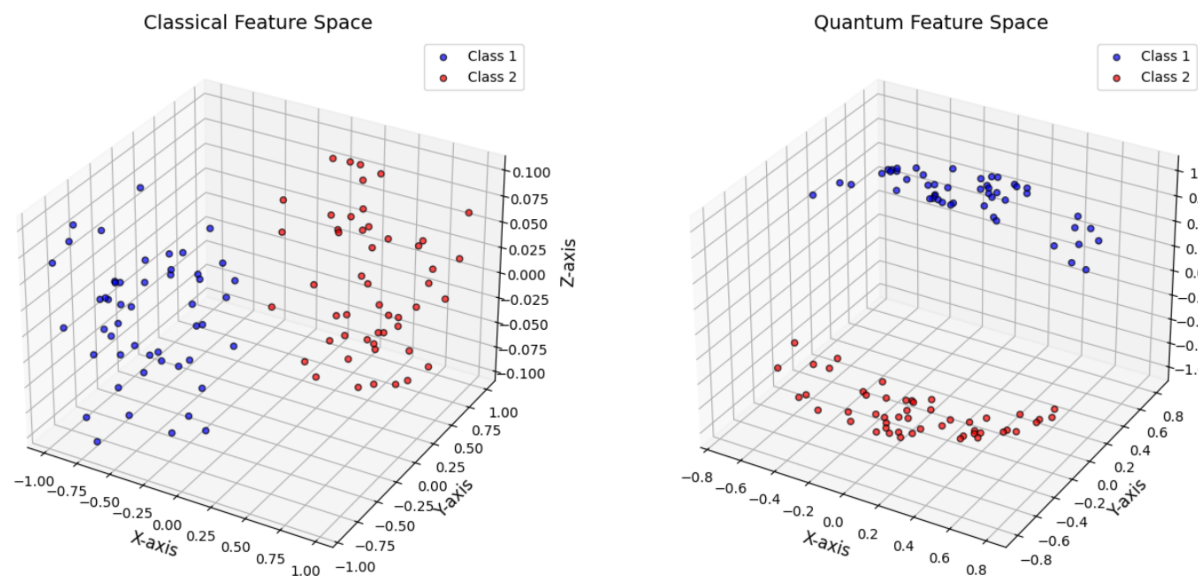


Figure 1: Comparison between classical and quantum feature spaces. Quantum feature spaces leverage exponential space growth for enhanced pattern recognition.

3.5 Quantum Gates and Their Functions

Quantum gates are the fundamental building blocks of QC, analogous to classical logic gates, but they operate on quantum bits (qubits) rather than classical bits. Unlike classical gates, quantum gates are represented by unitary matrices, ensuring reversibility in quantum operations [30]. These gates manipulate quantum states by leveraging principles such as superposition and entanglement, enabling complex computations that classical computers cannot efficiently simulate.

Table 2 presents an overview of commonly used quantum gates, their matrix representations, and a brief description of their functionalities.

Table 2: Common Quantum Gates and Their Functions.

Gate	Matrix Representation	Functionality Description
Hadamard (H)	$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}$	Creates equal superposition of $\|0\rangle$ and $\|1\rangle$ states [30].
Pauli-X (X)	$\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}$	Equivalent to the classical NOT gate; flips $\|0\rangle$ to $\|1\rangle$ and vice versa [30].
Pauli-Y (Y)	$\begin{bmatrix}0 & -i\\i & 0\end{bmatrix}$	Rotates the qubit state around the Y-axis of the Bloch sphere by π radians [31].
Pauli-Z (Z)	$\begin{bmatrix}1 & 0\\0 & -1\end{bmatrix}$	Applies a phase shift of π to the $\|1\rangle$ state [32].
CNOT (CX)	$\begin{bmatrix}1 & 0 & 0 & 0\\0 & 1 & 0 & 0\\0 & 0 & 0 & 1\\0 & 0 & 1 & 0\end{bmatrix}$	A two-qubit gate that flips the target qubit if the control qubit is $\|1\rangle$ [30].
Toffoli (CCX)	$\begin{bmatrix}1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\0 & 0 & 1 & 0 & 0 & 0 & 0 & 0\\0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\\0 & 0 & 0 & 0 & 1 & 0 & 0 & 0\\0 & 0 & 0 & 0 & 0 & 1 & 0 & 0\\0 & 0 & 0 & 0 & 0 & 0 & 0 & 1\\0 & 0 & 0 & 0 & 0 & 0 & 1 & 0\end{bmatrix}$	A controlled-controlled NOT gate used in reversible computing [33].
SWAP	$\begin{bmatrix}1 & 0 & 0 & 0\\0 & 0 & 1 & 0\\0 & 1 & 0 & 0\\0 & 0 & 0 & 1\end{bmatrix}$	Exchanges the quantum states of two qubits [30].
Phase (S)	$\begin{bmatrix}1 & 0\\0 & i\end{bmatrix}$	Applies a phase shift of $\pi/2$ to the $\|1\rangle$ state [32].
T-gate (T)	$\begin{bmatrix}1 & 0\\0 & e^{i\pi/4}\end{bmatrix}$	Introduces a phase shift of $\pi/4$, playing a key role in quantum universality [30].

In the construction of quantum algorithms such as Grover's algorithm for unstructured search and Shor's algorithm for integer factorization, quantum gates are fundamental [33] [34]. One common use of the Hadamard gate is the construction of superpositions, which enable quantum parallelism. One essential part of quantum processing, entangling qubits, requires the CNOT gate [31]. A key component of fault-tolerant QC design is the Toffoli gate, which is classically applicable to all reversible computations.

In quantum circuits, the gates facilitate the precise manipulation of qubit states, thereby allowing for the execution of quantum algorithms that can exceed the performance of their classical counterparts. The flexibility of quantum computers is contingent upon the effective deployment and enhancement of quantum gates as the hardware evolves.

3.6 Proposed Quantum K-Nearest Neighbors (QKNN)

The proposed QKNN algorithm introduces a novel approach to classification by integrating quantum computational principles into the conventional CKNN framework. This quantum-enhanced method leverages quantum state encoding, feature extraction through quantum transformations, and quantum distance computations to improve classification accuracy and efficiency. By exploiting quantum mechanical properties such as superposition and entanglement, QKNN has the potential to process complex datasets more effectively than the classical methods, particularly for high-dimensional and intricate feature spaces. Figure 2 represents the methodology of the proposed QKNN algorithm.

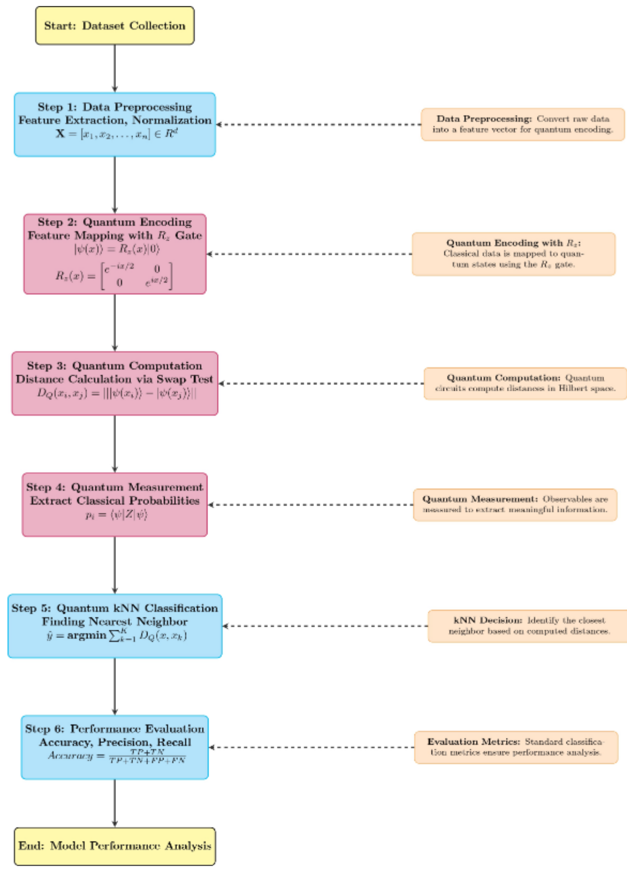


Figure 2: Methodology of the Proposed QKNN Algorithm

Quantum Data Encoding

The first step in the proposed QKNN algorithm involves encoding classical data points into quantum states, enabling the exploitation of quantum computation advantages. Given a training dataset $X_{\text{train}} = \{x_{\text{train}}^{(1)}, x_{\text{train}}^{(2)}, \ldots, x_{\text{train}}^{(N)}\}$ and a test dataset $X_{\text{test}} = \{x_{\text{test}}^{(1)}, x_{\text{test}}^{(2)}, \ldots\}$, each data point is mapped onto a quantum state using Hadamard and rotation gates, ensuring efficient representation and accessibility within the quantum framework.

The Hadamard transformation initializes qubits into an equal superposition state, laying the foundation for parallel quantum computations. The states are given by Eqn. (4) as follows:

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle). \tag{4}$$

Applying Hadamard gates across d qubits produces a uniform superposition state that facilitates efficient quantum data processing. The states are represented by Eqn. (5):

$$|\psi_{\text{init}}\rangle = \frac{1}{\sqrt{2^d}} \sum_{i=0}^{2^d-1} |i\rangle. \tag{5}$$

Rotation gates R_Z are applied to embed feature values into the quantum state. The rotation gate is specified by Eqn. (6) as follows:

$$R_Z(\theta) = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}, \tag{6}$$

where $\theta = 2\pi x_i$ ensures that feature values are normalized within the quantum system. As a result, each classical data point is transformed into the quantum representation:

$$|x_{\text{train}}\rangle = \bigotimes_{i=1}^{d} R_Z(\theta_i) H|0\rangle. \tag{7}$$

This encoding capability enables the quantum system to simultaneously process multiple data points improving computational efficiency.

Quantum Feature Extraction

Quantum transformations enhance data separation and induce structural modifications on encoded information. The IsingXY gate is implemented for establishing entanglement between qubits and enhancing feature interactions, defined as follows:

$$U_{\text{IsingXY}}(\theta) = \exp\left(-i\frac{\theta}{2}(\sigma_x^{(1)}\sigma_x^{(2)} + \sigma_y^{(1)}\sigma_y^{(2)})\right). \tag{8}$$

This gate guarantees that quantum states show higher-order feature connections, hence enabling improved classification capacity. Controlled-NOT (CNOT) gates are also used to lower class overlap and enhance feature space data point individuality. The transformed quantum states are obtained as follows:

$$|\psi_{\text{train}}\rangle \to U|\psi_{\text{train}}\rangle, \quad |\psi_{\text{test}}\rangle \to U|\psi_{\text{test}}\rangle. \tag{9}$$

By executing these quantum modifications, the proposed QKNN algorithm surpasses the CKNN in class separation.

Quantum Distance Calculation and Neighbor Selection

The similarity between a test instance $|\psi_{\text{test}}\rangle$ and training instances $|\psi_{\text{train},i}\rangle$ is evaluated using quantum distance metrics. The swap test is utilized to compute the quantum distance, which is defined as:

$$D(\psi_{\text{train},i}, \psi_{\text{test}}) = \frac{1}{2}\left(1 + |\langle\psi_{\text{test}}|\psi_{\text{train},i}\rangle|^2\right). \tag{10}$$

This metric captures the closeness of quantum states, where lower values indicate higher similarity. The nearest neighbors for the test point are selected based on the smallest quantum distances, ensuring an optimal classification.

Classification via Quantum Nearest Neighbors

After identifying the nearest neighbors, classification is performed through a majority voting mechanism. The predicted class label is determined as follows:

$$C_{\text{pred}}(X_{\text{test}}) = \arg\max_{C_i} \sum_{i=1}^{K} \mathbb{I}(C_{\text{train},i} = C_i), \tag{11}$$

where $\mathbb{I}(\cdot)$ is an indicator function that counts occurrences of each class among the selected neighbors. The class with the highest frequency is assigned to the test instance, ensuring robust classification.

As shown in Figure 3, the proposed QKNN architecture provides a novel approach to improve the efficiency of KNN classification by incorporating the QC principles.

Proposed QKNN Algorithm

As outlined in Algorithm 1, the proposed QKNN algorithm leverages quantum transformations for feature extraction and utilizes quantum distances, determined through the swap test, to identify the nearest neighbors for classification.

Algorithm 1 The Proposed Quantum K-Nearest Neighbors (QKNN) Algorithm

Require: Training dataset $X_{\text{train}}, Y_{\text{train}}$, test instance X_{test}.
Ensure: Predicted class label C_{pred}.
1: Encode classical data into quantum states using Hadamard and rotation gates.
2: Apply quantum transformations (IsingXY, CNOT) for feature extraction.
3: Compute quantum distances using the swap test.
4: Identify nearest neighbors based on minimum quantum distances.
5: Determine the class using majority voting among the selected neighbors.
6: **Return** C_{pred}.

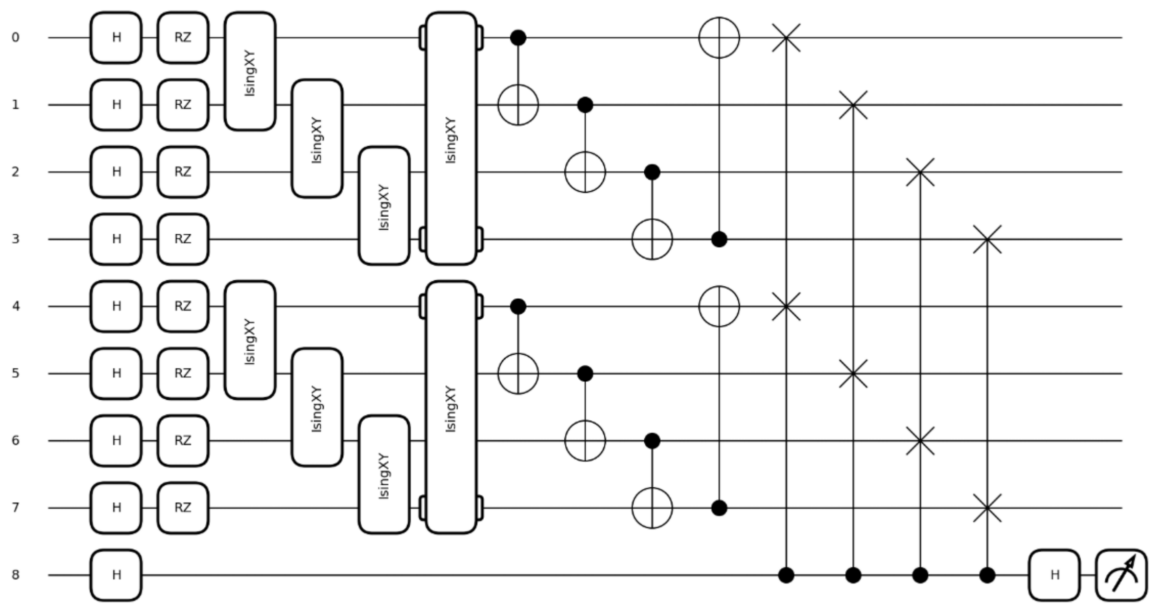


Figure 3: The proposed QKNN architecture.

Table 3 presents a comparative analysis of various existing QKNN approaches, focusing on key factors such as encoding methods, feature extraction techniques, distance metrics, neighbour selection and noise mitigation strategies. In comparison to existing methods, our proposed QKNN algorithm introduces significant performance improvements with more simpler quantum circuit and less circuit depth. Our method aims to optimize the performance and scalability of QKNN models, making them more feasible for real-world applications where computational resources and quantum hardware capabilities are constrained.

Table 3: Methodological Comparison of QKNN Approaches

Study	**Encoding Method**	**Feature Extraction**	**Distance Metric**	**Neighbour Selection**	**Noise Mitigation**
Gao et al. [35]	Amplitude Encoding with Quantum Random Access Memory (QRAM)	Classical	Quantum Mahalanobis Distance	Grover search based on quantum distance	Not addressed
Zardini et al. [13]	Amplitude Encoding	Classical	Quantum Euclidean Distance	Quantum distance based sorting	Not addressed
Jing Li et al. [36]	Binary Encoding Method	Not addressed	Quantum Hamming Distance	Quantum distance based sorting	Not addressed
Feng et al. [18]	Amplitude Encoding	Classical	Quantum Polar Distance	Grover search based on quantum distance	Not addressed
Proposed Work	Hadamard + R_Z Rotation Gates	IsingXY + CNOT gates for entanglement	Swap test	Quantum swap test based sorting	Repetition Encoding

By leveraging quantum-enhanced distance evaluation and feature transformations, the QKNN algorithm provides improved efficiency and performance in classification tasks. The integration of quantum mechanics in ML algorithms enables faster computations and enhanced pattern recognition, making QKNN a promising approach for ML applications in QC environments.

3.7 Quantum Neural Network for Classification

Traditional QNNs combine the computational advantages of QC with the learning capabilities of classical neural networks. QNNs leverage quantum mechanical properties such as superposition, entanglement, and quantum interference to perform ML tasks efficiently [37] [38]. Unlike the classical neural networks that rely on weighted connections between neurons, QNNs utilize parameterized quantum circuits (PQCs) to process and classify data [1] [39].

The fundamental building block of a QNN is a quantum circuit composed of unitary transformations. Given an input state $|\psi_{in}\rangle$, a QNN applies a series of quantum gates parameterized by θ:

$$|\psi_{out}\rangle = U(\theta)|\psi_{in}\rangle, \tag{12}$$

where $U(\theta)$ is a unitary matrix representing the quantum circuit [40]. The trainable parameters θ are optimized using classical gradient-based methods, often through a quantum-classical hybrid approach involving the variational quantum eigensolver (VQE) or quantum natural gradient methods [41].

The capacity of QNNs to encode data into higher-dimensional Hilbert spaces is one of its most important characteristics; this allows for more detailed decision boundary representations than is possible with more conventional methods. The generalized achievement in ML tasks can be enhanced using this quantum-enabled feature space [4].

QNNs use quantum circuits for data measurement, translation, and encoding, which is a new way to tackle classification challenges. This section outlines the theoretical foundations and methodology employed in the proposed quantum classification structure. The QNN architecture with parameterized quantum circuits is illustrated in Figure 4.

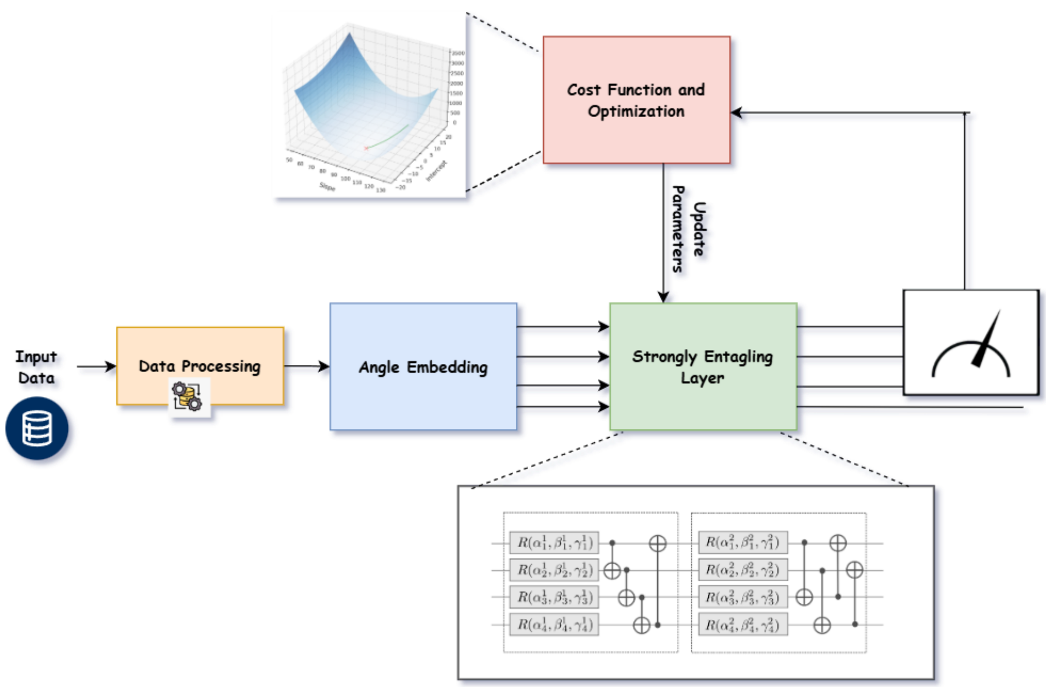


Figure 4: Quantum Neural Network Architecture with Parameterized Quantum Circuits.

Quantum Encoding and Circuit Architecture

Given a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^d$ represents the feature vector and y_i denotes the corresponding label, the QNN begins with quantum state preparation. Classical data is embedded into a quantum state via angle embedding as follows:

$$U_{embed}(x) = \prod_{i=1}^{d} R_Y(x_i)|0\rangle, \tag{13}$$

where $R_Y(x_i) = e^{-ix_iY/2}$ is a single-qubit rotation about the Y-axis [30].

To introduce non-linearity and entanglement, the circuit employs strongly entangling layers as follows:

$$U_{entangle}(\theta) = \prod_{l=1}^{L} \left[\prod_{i=1}^{d} R_Z(\theta_{l,i}) CNOT(i, i+1) \right], \tag{14}$$

where θ are trainable parameters and the $CNOT$ gates that establish qubit entanglement [38].

Quantum Measurement and Output

After transformation through the parameterized quantum gates, measurement in the computational basis is performed. The expectation values of the Pauli-Z operators are used to obtain output features as shown below:

$$\hat{y} = \left[\langle 0|U^\dagger Z_i U|0\rangle \right]_{i=1}^{C}, \tag{15}$$

where C denotes the number of classes [4].

Cost Functions for Classification

The loss function guides the training of quantum parameters. For binary classification, we employ the binary cross entropy (BCE) is given by:

$$L_{binary} = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i) \right]. \tag{16}$$

For multiclass classification, we utilize the categorical cross entropy (CCE) is given by:

$$L_{multi} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{C} y_{i,j} \log \hat{y}_{i,j}, \tag{17}$$

where C represents the total number of classes and $\hat{y}_j$ is obtained via the softmax activation function represented by:

$$\hat{y}_j = \frac{e^{z_j}}{\sum_{k=1}^{C} e^{z_k}}. \tag{18}$$

This formulation allows efficient training of the QNN parameters using gradient-based optimization techniques, enabling effective quantum-based classification of complex datasets.

3.8 Noise Modeling and Error Mitigation in Quantum K-Nearest Neighbors

QC is inherently susceptible to noise due to decoherence and gate imperfections, significantly impacting the reliability of QML algorithms such as the QKNN classifier [6] [30]. In this section, we present a noise model based on Pauli error channels and introduce an error mitigation strategy using repetition codes to enhance computational stability [42] [43].

Quantum Noise Channels

Quantum noise channels describe the probabilistic evolution of quantum states under noise, impacting the accuracy of quantum computations [30] [44]. The primary sources of noise considered in this study include:

- **Bit-Flip Channel (Pauli-X Error):** This channel introduces bit-flip errors, modeled as the application of the Pauli-X gate [6]:
$$\mathcal{E}_X(\rho) = (1-p)\rho + pX\rho X^\dagger. \tag{19}$$
where p is the probability of the occurrence of a bit flip.
- **Phase-Flip Channel (Pauli-Z Error):** This channel induces phase errors, represented by the Pauli-Z operator [44]:
$$\mathcal{E}_Z(\rho) = (1-p)\rho + pZ\rho Z^\dagger. \tag{20}$$

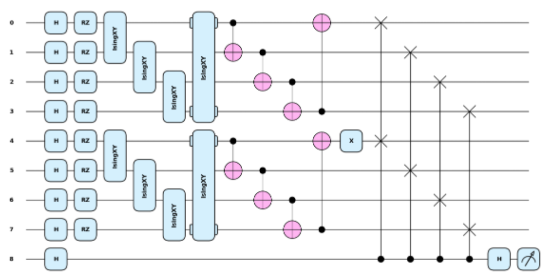
- **Bit-Phase Flip Channel (Pauli-Y Error):** This channel simultaneously introduces bit-flip and phase-flip errors using the Pauli-Y gate [30]:

$$\mathcal{E}_Y(\rho) = (1-p)\rho + pY\rho Y^{\dagger}. \tag{21}$$

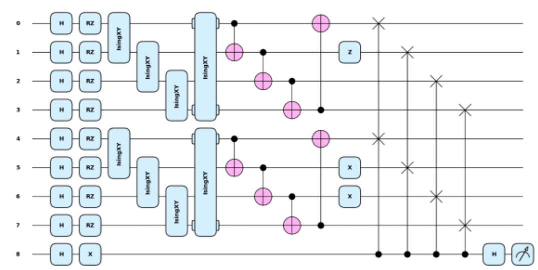
To analyze the robustness of QKNN under noisy conditions, we implement a probabilistic noise model where each qubit in the quantum circuit undergoes one of these errors with probability $\frac{p}{3}$ [6] [42]. The resulting noisy state evolution is given by:

$$\mathcal{E}(\rho) = (1-p)\rho + \frac{p}{3}(X\rho X + Z\rho Z + Y\rho Y). \tag{22}$$

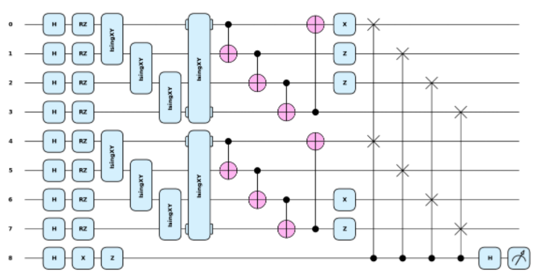
Figure 5 shows the quantum circuits representing various noise models are illustrated for various noise probabilities, ranging from $p = 0.1$ to $p = 0.6$.



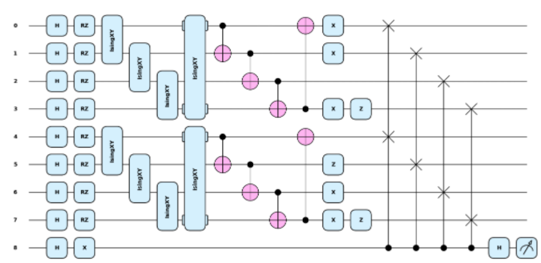
(a) Noise probability $p = 0.1$



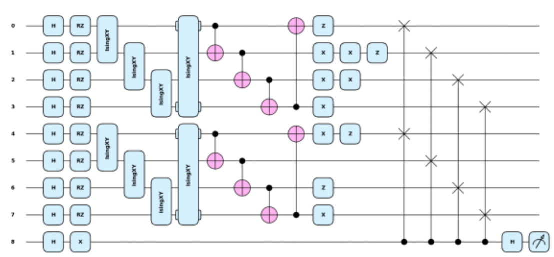
(b) Noise probability $p = 0.2$



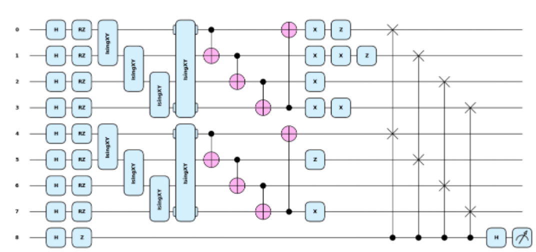
(c) Noise probability $p = 0.3$



(d) Noise probability $p = 0.4$



(e) Noise probability $p = 0.5$



(f) Noise probability $p = 0.6$

Figure 5: Quantum circuits representing various noise models with increasing noise probability.

Error Mitigation via Repetition Codes

To counteract the effects of quantum noise, we employ a n-qubit repetition code, an error mitigation technique that redundantly encodes logical qubits across multiple physical qubits [43].

Encoding and Error Detection: A logical qubit is encoded as:

$$|0\rangle_L = |00\ldots0\rangle, \quad |1\rangle_L = |11\ldots1\rangle. \tag{23}$$

Errors are detected using stabilizer measurements of the form [30]:

$$S_i = Z_i Z_{i+1}, \quad i \in \{1, 2, \ldots, n-1\}. \tag{24}$$

A stabilizer outcome of $+1$ indicates no error, while -1 signals a bit-flip or phase-flip error. Table 4 shows the error syndrome table for a 3-qubit repetition code, illustrating the possible error patterns and their corresponding corrections.

Table 4: Error Syndrome Table for a 3-Qubit Repetition Code.

$\mathbf{S_1}$	$\mathbf{S_2}$	**Error Pattern**	**Correction**
+1	+1	$I \otimes I \otimes I$	No Error
+1	-1	$I \otimes I \otimes X$	Apply X gate on Qubit 1
-1	+1	$X \otimes I \otimes I$	Apply X gate on Qubit 3
-1	-1	$I \otimes X \otimes I$	Apply X gate on Qubit 2

Error Correction: By analyzing the stabilizer syndromes, errors are identified and corrected using a majority voting scheme [45]:

- If the majority of qubits remain in $|0\rangle$, the logical state is $|0\rangle_L$.
- If the majority of qubits are in $|1\rangle$, the logical state is $|1\rangle_L$.

This redundancy-based approach ensures resilience to single-qubit errors, thus improving the reliability of QKNN classification under noisy conditions [6].

The integration of repetition codes into QKNN enables error-resilient quantum state preparation and measurement, which is essential for practical implementations on near-term quantum devices. Future research may explore more advanced error correction strategies, such as surface codes, to further enhance the robustness of QML algorithms [45].

4 Result Analysis and Discussion

In this part, we conduct a thorough evaluation of the classification capabilities of QKNN, CKNN, and QNN on three standard datasets: Breast Cancer, Iris, and Bank Note Authentication. The review provides a comprehensive picture of the success of each model by including important performance parameters, including accuracy, precision, recall, F1 score, and AUC. Furthermore, in order to evaluate the stability of the model, we investigate how quantum noise affects the performance of QKNN.

4.1 Comparative Performance Analysis Across Models

A comparative assessment of classification accuracy across the three models is presented in Table 5. The results reveal that the proposed QKNN outperforms or matches CKNN, showcasing its superior classification capabilities. Although QNN demonstrates competitive performance, it lags behind in performance for some datasets owing to the challenges associated with optimization of the variational quantum circuits.

Table 5: Comparison of QKNN, CKNN, and QNN Accuracies across the Three Datasets.

Dataset	QKNN	CKNN	QNN
Breast Cancer	**0.9825**	0.9298	0.9717
Iris	**1.0000**	**1.0000**	0.8333
Bank Note	**0.9927**	0.9855	0.8618

The findings indicate that the proposed QKNN consistently achieves superior performance over CKNN, particularly in complex datasets such as Breast Cancer and Bank Note Authentication. For the breast cancer dataset, QKNN attains an accuracy of 98.25%, outperforming CKNN (92.98%) and QNN (97.17%). This significant improvement suggests that QKNN effectively leverages quantum feature encoding and entanglement to enhance the classification performance.

For the Iris dataset, both QKNN and CKNN achieve perfect classification accuracy of 100%, indicating that the dataset is inherently linearly separable. However, QNN performs notably worse at 83.33%, demonstrating that the model struggles with capturing the underlying feature space owing to the limitations of quantum variational circuits under small-scale training scenarios.

Similarly, in the Bank Note Authentication dataset, QKNN achieves the highest accuracy of 99.27%, surpassing both CKNN (98.55%) and QNN (86.18%). The lower performance of QNN in this case further underscores the difficulty in training QNNs compared to distance-based quantum-enhanced methods such as QKNN. Figure 6 shows the comparative performance of QKNN, CKNN, and QNN across various datasets.

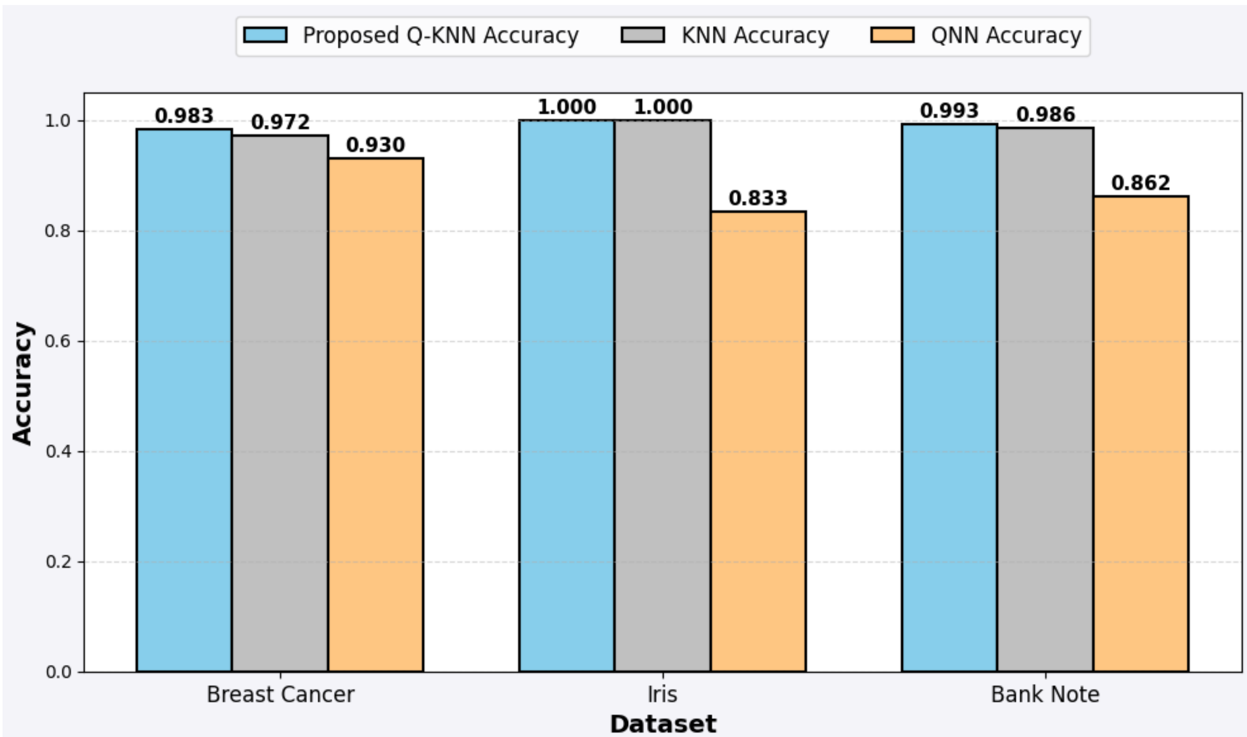


Figure 6: Comparative Performance of QKNN, CKNN, and QNN across the three datasets.

4.2 Key Performance Metrics and Their Implications

To further assess the efficiency of QKNN, we analyze additional performance metrics, as shown in Table 6, including AUC, F1 score, precision, and recall. These metrics provide deeper insight into the reliability of the classification beyond accuracy.

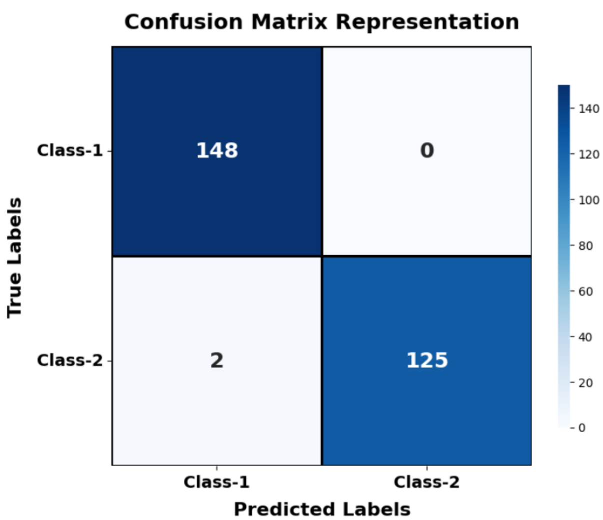
Table 6: Performance Metrics of the Proposed Quantum-KNN (QKNN) Architecture.

Dataset	AUC	F1-Score	Precision	Recall
Breast Cancer	0.9859	0.98	0.98	0.99
Iris	1.0000	1.00	1.00	1.00
Bank Note	0.9921	0.99	0.99	0.99

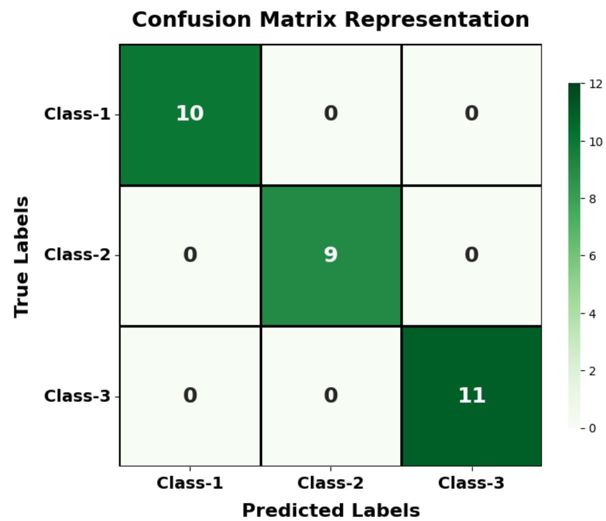
The AUC values for QKNN are consistently high, demonstrating its strong power of classification across three datasets. The high F1-score values indicate a balanced performance between precision and recall, confirming the reliability of QKNN in minimizing both false positives and false negatives. This balance is particularly important in high-stakes applications such as medical diagnosis and fraud detection, where misclassification can have significant consequences.

4.3 Confusion Matrices for Model Performance

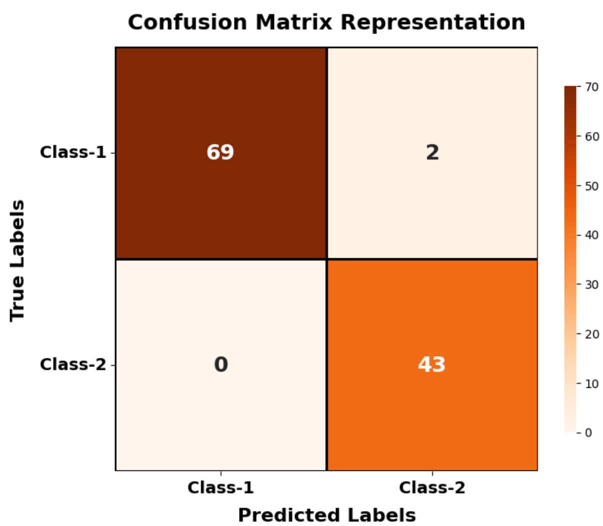
Confusion matrices provide additional insights into the ability of a model to distinguish between true positives, false positives, true negatives, and false negatives. The confusion matrices for each dataset are shown in Figure 7. These matrices offer a deeper understanding of how well the models perform in various classification scenarios.



(a) Confusion Matrix for Banknote Authentication Dataset



(b) Confusion Matrix for Iris Dataset



(c) Confusion Matrix for Breast Cancer Wisconsin Dataset

Figure 7: Confusion Matrices for various datasets.

The comparative performance analysis across QKNN, CKNN, and QNN shows that QKNN consistently delivers superior performance, particularly in complex datasets. The evaluation of additional metrics such as AUC, F1-score, precision, and recall confirms that the proposed QKNN achieves a balanced and reliable classification performance. The results suggest that quantum-enhanced methods like QKNN are more effective in leveraging quantum properties for complex classification tasks compared to classical methods and QNNs.

4.4 Effect of Quantum Noise on QKNN Performance

QC systems are susceptible to noise, which can impact classification performance. Figure 8 illustrates the effect of increasing noise probability on the accuracy of QKNN. The noise probability p is varied from 0 to 0.6 to observe its impact on the performance of the model.

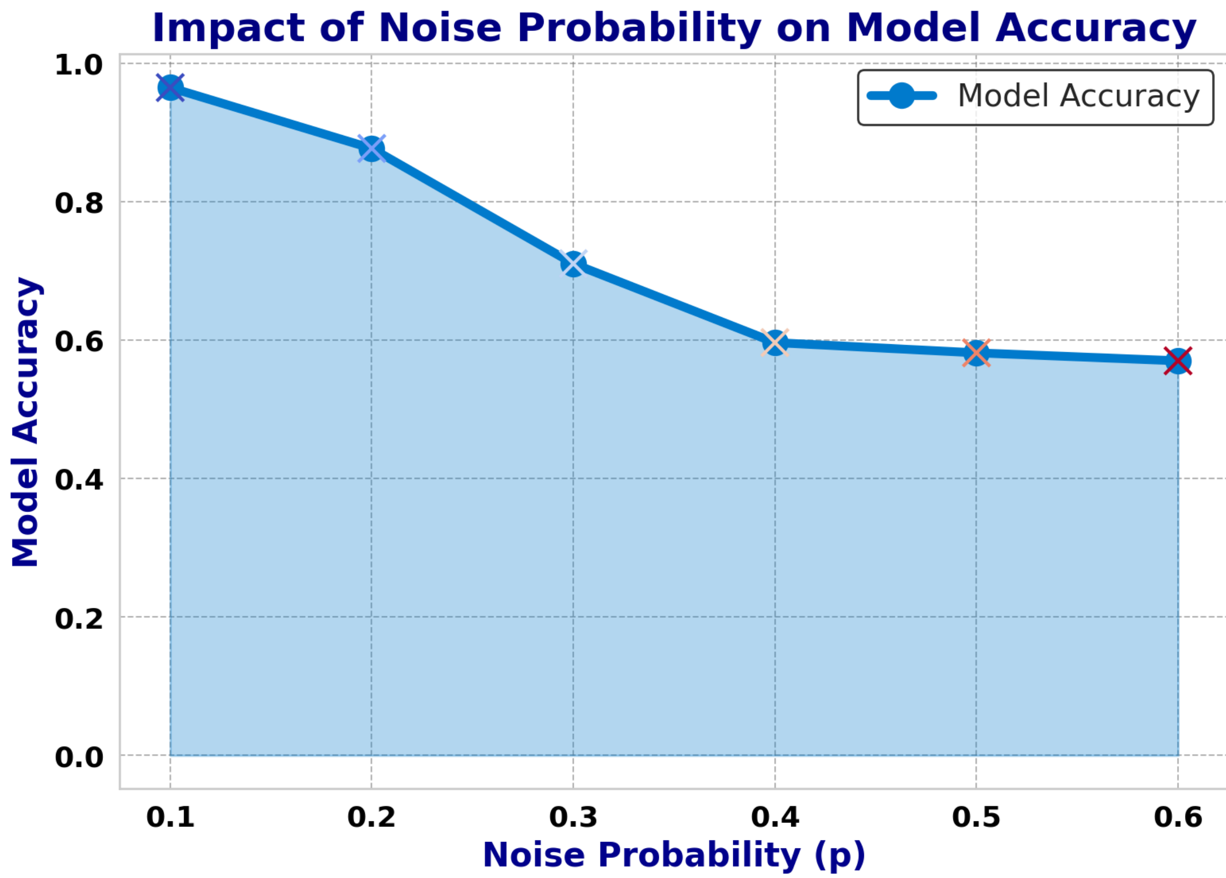


Figure 8: Impact of Noise Probability on Model Accuracy.

The results highlight a clear inverse relationship between noise probability and model accuracy. QKNN achieves an accuracy of 96.49% at a low noise level of $p = 0.1$, demonstrating its robustness under minimal noise. However, as p increases to 0.3, accuracy declines sharply to 71.05%, indicating the sensitivity of the model to quantum decoherence and gate errors. At higher noise levels ($p \geq 0.6$), accuracy stabilizes at approximately 57%, suggesting a noise tolerance threshold beyond which the model performance is severely degraded.

These observations emphasize the importance of implementing quantum error correction techniques such as repetition encoding to mitigate the effects of noise and improve the robustness of classification. The need for error correction becomes more evident as the noise probability increases, underscoring the challenges faced by QML models in noisy quantum environments.

5 Conclusion and Future Work

This study presented the QKNN algorithm as a quantum-enhanced classification method and evaluated its performance against CKNN and QNN across multiple datasets, including Breast Cancer, Iris, and Bank Note Authentication. The experimental results showed that the proposed QKNN consistently outperforms or matches CKNN, particularly excelling in datasets with complex decision boundaries. Specifically, QKNN achieved an accuracy of 98.25% on the breast cancer dataset against that of 92.98% in CKNN, and attained 99.27% accuracy on the bank note dataset, outperforming CKNN at 98.55%. These results demonstrate the ability of QKNN to better handle complex and high-dimensional classification tasks. For the Iris dataset, both QKNN and CKNN achieved perfect classification accuracy of 100%, while QNN lagged behind at 83.33%, suggesting that variational quantum circuits may not be very effective for certain classification tasks without extensive parameter optimization.

A critical aspect of this study was to analyze the consequences of quantum noise on QKNN accuracy. The results revealed a strong inverse correlation between noise probability and model accuracy, with performance sharply degrading beyond $p = 0.3$, stabilizing at approximately 57% accuracy at $p = 0.6$. This highlights the sensitivity of quantum models to noise and the hurdles posed by quantum decoherence and gate errors on near-term quantum devices. These findings emphasize the need for quantum error correction strategies to strengthen model validity in practical quantum environments.

The implications of these results are significant for QML applications. The superior accuracy and robustness of QKNN in complex datasets suggest its potential in critical areas such as medical diagnostics, financial fraud detection, and security. However, to move from theoretical promise to practical deployment, several avenues must be explored in future work. Firstly, optimizing quantum feature encoding techniques to improve scalability on larger datasets is essential. Secondly, developing adaptive quantum circuits that dynamically adjust to noise conditions could further enhance QKNN's performance. Moreover, empirical validation on real quantum hardware will provide valuable insights into quantum computational constraints, guiding future refinements of the model. Lastly, integrating QKNN into the hybrid quantum-classical frameworks will allow the strengths of both paradigms to be leveraged, potentially leading to more powerful and scalable QML solutions. As QC technology matures, continued improvements in QKNN and related quantum classifiers could revolutionize data-driven decision-making in high-impact industrial applications.

References

[1] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, 2017.

[2] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[3] N. Wiebe, A. Kapoor, and K. M. Svore. Quantum algorithms for nearest-neighbor methods for supervised and unsupervised learning. *Quantum Information and Computation*, 15(3-4):316–356, 2015.

[4] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta. Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747):209–212, 2019.

[5] S. Lloyd, M. Mohseni, and P. Rebentrost. Quantum algorithms for supervised and unsupervised machine learning. *arXiv preprint arXiv:1307.0411*, 2013.

[6] J. Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, 2018.

[7] Z. Yang, M. Zolanvari, and R. Jain. A survey of important issues in quantum computing and communications. *IEEE Communications Surveys & Tutorials*, PP(99):1–1, Jan. 2023.

[8] R. D. Ranga, A. Rana, S. Prajapat, P. Kumar, K. Kumar, and A. V. Vasilakos. Quantum machine learning: Exploring the role of data encoding techniques, challenges, and future directions. *Mathematics*, 12(21):3318, 2024.

[9] H. Yano, Y. Suzuki, K. M. Itoh, R. Raymond, and N. Yamamoto. Efficient discrete feature encoding for variational quantum classifier. *arXiv preprint arXiv:2005.14382*, 2020.

[10] X. B. Nguyen, H. Q. Nguyen, H. Churchill, S. U. Khan, and K. Luu. Quantum visual feature encoding revisited. *arXiv preprint arXiv:2405.19725*, 2024.

[11] K. Sakka, K. Mitarai, and K. Fujii. Automating quantum feature map design via large language models. *arXiv preprint arXiv:2504.07396*, 2025.

[12] Z. Jarir and M. Quafafou. Quantum-enhanced k-nearest neighbors for text classification: A hybrid approach. *Advanced Quantum Technologies*, 2024.

[13] E. Zardini, E. Blanzieri, and D. Pastorello. A quantum k-nearest neighbors algorithm based on the euclidean distance estimation. *Quantum Machine Intelligence*, 2024.

[14] Jiaye Li, Jian Zhang, Jilian Zhang, and Shichao Zhang. Quantum knn classification with k value selection and neighbor selection. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 43(5):1332–1345, 2023.

[15] Alessandro Berti, Anna Bernasconi, Gianna M Del Corso, and Riccardo Guidotti. The role of encodings and distance metrics for the quantum nearest neighbor. *Quantum Machine Intelligence*, 6(2):62, 2024.

[16] Y. Wang and R. Wang. Improved handwritten digit recognition using quantum k-nearest neighbor algorithm. *Springer*, 2019.

[17] P. Bhaskaran and S. Prasanna. An accuracy analysis of classical and quantum-enhanced k-nearest neighbor algorithm. *Knowledge and Information Systems*, 2025.

[18] Congcong Feng, Bo Zhao, Xin Zhou, Xiaodong Ding, and Zheng Shan. An enhanced quantum k-nearest neighbor classification algorithm based on polar distance. *Entropy*, 25(1):127, 2023.

[19] Qiuyu Xiang, Dongfen Li, Zhikang Hu, Yuhang Yuan, Yuchen Sun, Yonghao Zhu, You Fu, Yangyang Jiang, and Xiaoyu Hua. Quantum classical hybrid convolutional neural networks for breast cancer diagnosis. *Scientific Reports*, 14(1):24699, 2024.

[20] Rong Zhul, Guangshun Li, Jin-Xing Liu, Ling-Yun Dai, Shasha Yuan, and Ying Guo. A fast quantum clustering approach for cancer gene clustering. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1610–1613. IEEE, 2018.

[21] W. Street, W. Wolberg, and O. Mangasarian. Nuclear feature extraction for breast tumor diagnosis. UCI Machine Learning Repository, 1993.

[22] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188, 1936.

[23] D. Dua and C. Graff. Banknote authentication dataset. UCI Machine Learning Repository, 2019.

[24] G. Forman. An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research*, 3:1289–1305, 2003.

[25] Jie Cai, Jiawei Luo, Shulin Wang, and Sheng Yang. Feature selection in machine learning: A new perspective. *Neurocomputing*, 300:70–79, 2018.

[26] Maria Schuld, Alex Bocharov, Krysta M. Svore, and Nathan Wiebe. Circuit-centric quantum classifiers. *Physical Review A*, 101(3), March 2020.

[27] Pierre-Luc Dallaire-Demers and Nathan Killoran. Quantum generative adversarial networks. *Physical Review A*, 98(1):012324, 2018.

[28] S. Aaronson. *Read the fine print*, volume 11. 2015.

[29] Thomas Hofmann, Bernhard Schölkopf, and Alexander J Smola. Kernel methods in machine learning. 2008.

[30] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

[31] D. Deutsch. Quantum theory, the church-turing principle, and the universal quantum computer. *Proceedings of the Royal Society of London A*, 400(1818):97–117, 1989.

[32] R. P. Feynman. Quantum mechanical computers. *Foundations of Physics*, 16(6):507–531, 1986.

[33] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997.

[34] L. K. Grover. A fast quantum mechanical algorithm for database search. *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, pages 212–219, 1996.

[35] Li-Zhen Gao, Chun-Yue Lu, Gong-De Guo, Xin Zhang, and Song Lin. Quantum k-nearest neighbors classification algorithm based on mahalanobis distance. *Frontiers in Physics*, 10:1047466, 2022.

[36] Jing Li, Song Lin, Kai Yu, and Gongde Guo. Quantum k-nearest neighbor classification algorithm based on hamming distance. *Quantum Information Processing*, 21(18), 2022.

[37] M. Schuld, I. Sinayskiy, and F. Petruccione. The quest for a quantum neural network. *Quantum Information Processing*, 14(7-8):796–817, 2014.

[38] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii. Quantum circuit learning. *Physical Review A*, 98(3):032309, 2018.

[39] Renxin Zhao and Shi Wang. A review of quantum neural networks: methods, models, dilemma. *arXiv preprint arXiv:2109.01840*, 2021.

[40] Maria Schuld and Francesco Petruccione. *Machine learning with quantum computers*, volume 676. Springer, 2021.

[41] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, August 2021.

[42] Simon J Devitt, William J Munro, and Kae Nemoto. Quantum error correction for beginners. *Reports on Progress in Physics*, 76(7):076001, 2013.

[43] Daniel A Lidar and Todd A Brun. *Quantum error correction*. Cambridge university press, 2013.

[44] H.-P. Breuer and F. Petruccione. *The Theory of Open Quantum Systems*. Oxford University Press, 2002.

[45] D. Gottesman. *Stabilizer codes and quantum error correction*. PhD thesis, California Institute of Technology, 1997.