QKNN: A Novel Quantum K-Nearest Neighbor Algorithm for High-Accuracy Classification with Quantum Noise Mitigation

*Asif Akhtab Ronggon, Tuhin Hossain, Tahani Jaser Alahmadi, Mohammad Ali Moni**

A. A. Ronggon
Department of Electrical and Electronic Engineering, Bangladesh University of Engineering and Technology, Dhaka-1000, Bangladesh
Email: asifaftab172@gmail.com
T. Hossain
Department of Computer Science and Engineering, Jahangirnagar University, Savar-1342, Dhaka, Bangladesh
Email: tuhin97.hossain@gmail.com
T. J. Alahmadi
Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, Saudi Arabia
Email: tjalahmadi@pnu.edu.sa
M. A. Moni*
AI & Digital Health Technology, Rural Health Research Institute, Charles Sturt University, Orange, NSW 2800, Australia
School of Health and Rehabilitation Sciences, The University of Queensland, St Lucia, Brisbane, QLD 4072, Australia
Email: m.moni@uq.edu.au*
Corresponding author: Mohammad Ali Moni (e-mail: m.moni@uq.edu.au).

Keywords: *Quantum machine learning, quantum K-nearest neighbors(QKNN), quantum distance computation, quantum feature mapping, quantum noise and error correction, classical KNN, hybrid classical-quantum systems, and quantum neural network(QNN).*

This paper introduces a novel quantum K-nearest neighbors(QKNN) algorithm, which offers improved performance over the classical KNN algorithm by incorporating quantum computing techniques to enhance classification accuracy, scalability, and robustness. Our approach optimizes Hadamard and rotation gates for quantum data encoding and embeds classical data into quantum states efficiently. Entangled gates like IsingXY and CNOT improve feature extraction and classification by allowing complicated feature interactions. A new quantum distance metric based on swap test results is used to calculate similarity measures between quantum states. This offers superior accuracy and computational efficiency compared to traditional Euclidean distance metrics. We evaluated the proposed QKNN algorithm on three benchmark datasets. We received its superior performance relative to classical KNN(CKNN), Quantum Neural Network(QNN), and recent QKNN studies. The proposed QKNN algorithm is found to achieve prediction accuracies of 98.25%, 100%, and 99.27%, respectively, for the three datasets, while the QNN shows prediction accuracies of 97.17%, 83.33%, and 86.18%, respectively. Moreover, quantum noise challenges are addressed by integrating a Shor code-based error mitigation strategy, which ensures stability of the algorithm and resilience to noisy quantum environments. The results indicate that the proposed QKNN algorithm is scalable, more efficient, and robust.

1 Introduction

The major computational challenges in modern research include data classification, feature selection, and model optimization. Quantum machine learning (QML) has emerged as a promising paradigm that leverages the principles of quantum mechanics to improve classical machine learning (ML) algorithms. Recent advances in quantum computing (QC) have demonstrated the potential of quantum algorithms to address these conventional problems in ways that classical approaches cannot, offering unique solutions and substantial improvements in computational efficiency and problem-solving capabilities.

The classical K-nearest neighbors (CKNN) algorithm is a widely used nonparametric machine learning (ML) approach in data mining and pattern recognition. Although this is a simple and efficient algorithm, it encounters significant computational hurdles, particularly for large datasets. This approach is computationally expensive for real-time applications because it relies on brute-force computation to determine closest neighbors for class prediction [1]. Therefore, it was necessary to find other ways to accelerate the classification process, as the performance of the CKNN decreases significantly as the dataset size increases. One possible solution to counteract these challenges is to use QC techniques, which can achieve exponential speedups compared to their classical counterparts.

Rapid advancements in QC have spurred significant interest in developing algorithms that possess the potential to outperform their classical counterparts across various domains, including machine learning (ML). QML integrates quantum computational principles with traditional ML algorithms to improve their computational efficiency and scalability. Considerable attention has been directed towards quantum implementations of well-established classical algorithms, including the quantum K-nearest neighbors (QKNN) algorithm. The versatility and effectiveness with which the QKNN and QNN algorithms categorize large datasets have brought them to the forefront of existing QML algorithms. They have attracted significant attention for their rapid processing power of high-dimensional data and greater computational efficiency than their classical counterparts [2]. This is why a quantum variant known as the quantum KNN or QKNN algorithm has emerged as a substitute for the CKNN algorithm [3]. A QKNN algorithm utilizes quantum features such as superposition, entanglement, and quantum parallelism to improve the effectiveness of searching for neighbors and computing the distance of target data points from the neighbor data points. In particular, quantum algorithms may offer more accurate similarity measurements compared to classical algorithms, and quantum systems can store large amounts of data [4]. For solving high-dimensional classification problems, the QKNN approach is a highly desirable option, as it uses quantum circuits to achieve exponential speedup compared to the traditional CKNN approach in some specific applications [5].

Despite its theoretical advantages, the current status of quantum technology continues to impede the practical implementation of the QKNN approach. The full realization of QML algorithms is not yet fully realizable because of decoherence, noise, and the limited number of qubit counts of currently available quantum hardware. Furthermore, developing an efficient quantum feature mapping technique poses a fundamental challenge. Therefore, research must be carried out to develop faster and more efficient techniques for the mapping process. Before practical implementation of the proposed QKNN algorithm, evaluating its computational complexity, resilience to quantum noise, and classification accuracy is critical in the presence of realistic quantum errors [6].

A few recent studies on developing the QKNN approach need special mention here. They are by Feng et al. [7], Maldonado et al. [8], Li et al. [9], and Bhaskaran et al. [10]. Feng et al. [7] employed quantum amplitude encoding alongside a quantum polar distance metric, integrated with classical feature preprocessing to enhance the performance of their QKNN model. Maldonado et al. [8] also proposed a QKNN methodology using angle encoding combined with Grover's algorithm and the SWAP test for efficient quantum state comparison and neighbor identification. Again, Li et al. [9] presented a QKNN methodology that simultaneously quantumizes neighbor selection and K-value optimization. They used a special quantum circuit based on quantum phase estimation, controlled rotation, and inverse phase estimation techniques. On the other hand, Bhaskaran et al. [10] used a QKNN methodology using the Canberra distance metric and evaluated its classification performance under quantum enhancement. They also performed a comparative performance analysis of their QKNN algorithm with the CKNN algorithm for multiple datasets.

However, there is still ample scope to improve the accuracy of the QKNN algorithm. In this work, we propose a novel QKNN algorithm that introduces several methodological advances not addressed in the studies carried out earlier. The proposed algorithm employs Hadamard and R_z rotation gates for quantum state encoding, followed by a dedicated feature extraction layer using IsingXY and CNOT gates to enhance entanglement and representation learning. To improve neighbor selection, we replace conventional Grover search with a quantum SWAP test-based sorting mechanism that offers a more scalable and structured alternative. Furthermore, the framework incorporates a noise mitigation technique based on Shor code. However, this vital component is missing in all the existing QKNN architectures reported earlier.

The objective of the proposed research was carefully chosen to address key challenges that hinder the practical implementation of QKNN algorithms, such as the effects of quantum noise, limited qubit resources, and hardware constraints. This study aims to bridge the gap between the theoretical promise and practical implementation of the proposed QKNN algorithm on near-term quantum devices using performance analysis, noise resilience assessment, and hardware-aware design. The major contributions of the current research are as follows:

- Propose a novel QKNN algorithm that combines efficient quantum data encoding with entangling gates and a distance metric based on the swap test to improve classification accuracy and efficiency.
- Introduce a conceptual framework that uses quantum superposition and entanglement to obtain richer feature representation and robust similarity measurement, balancing expressivity, noise resilience, and hardware efficiency.
- Develop an integrated noise model with Pauli error channels and suggest a Shor code-based error mitigation to improve the robustness of QKNN on noisy quantum hardware.
- Provide comprehensive benchmarking against CKNN, QNN, and recent QKNN studies across standard datasets, demonstrating superior performance, scalability, and noise tolerance.

In this section, an initial review of the literature on CKNN, QNN, and QKNN algorithms has been carried out. The major contributions of the proposed QKNN algorithm have also been highlighted. The remainder of this paper is organized as follows. Section II comments on the QML capability, its growth and challenges, and the feature mapping technique as its essential component. This section further elaborates on the related work on QKNN that was carried out in the past. Section III describes the three test datasets used to validate the proposed QKNN algorithm's performance, the algorithm's steps, and the experimental setup used. The results of the performance analysis of the proposed QKNN algorithm are presented in Section IV. This section also compares the performance parameters, such as classification accuracy, computational efficiency, and resilience to noise, of the proposed QKNN algorithm with those of the traditional CKNN and QNN approaches. Section V concludes the findings of the results and suggests further research directions in QML to enhance and make the proposed algorithm more robust. Finally, an acknowledgment is included at the end of the paper, followed by a list of references. For clarity of presentation, a summary of the complex symbols and notation used throughout the paper is shown in Table 1.

2 Related Work

In the previous section, we mentioned that quantum algorithms can uniquely solve conventional computational problems, including data classification, feature selection, and model optimization. In this section, we first comment on QML capability, its growth and challenges, and the feature mapping technique as an essential component. We then elaborate on some more research work carried out on the QKNN algorithm, apart from those mentioned in the previous section.

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

In recent years, an upsurge of interest has been observed in integrating QC principles with ML algorithms, especially the QKNN and QNN algorithms. For large, high-dimensional datasets and multiclassification jobs, such algorithms aim to simultaneously improve computational efficiency and classification accuracy. Despite the encouraging findings demonstrated by quantum techniques, several scaling problems and realistic application challenges persist, particularly about quantum noise and decoherence [11].

An essential component of QML is quantum feature encoding, which maps classical input to its equivalent quantum states. Several encoding methods, including amplitude encoding, angle encoding, and quantum random access coding (QRAC), have been examined to represent classical information in quantum systems effectively [12] [13]. Nguyen et al. [14] reviewed quantum visual encoding strategies and

Table 1: Nomenclature

Symbol	Description	Unit / Note
$D(\psi_{\text{train},i}, \psi_{\text{test}})$	Quantum distance metric based on swap test between training and test states	Dimensionless
$\mathcal{E}(\rho)$	Quantum noise channel acting on density matrix ρ	—
H	Hadamard gate operator	Unitary operator
$R_Z(\theta)$	Rotation gate about Pauli-Z axis parameterized by θ	Unitary operator
$R_Y(\theta)$	Rotation gate about Pauli-Y axis parameterized by θ	Unitary operator
S_i	Stabilizer operator for error detection in Shor code	Hermitian operator
U	General unitary operator for combined quantum transformations	Unitary operator
$U_{\text{embed}}(x)$	Quantum embedding operator encoding classical data x	Unitary operator
$U_{\text{entangle}}(\theta)$	Unitary operator implementing parameterized entangling gates	Unitary operator
$U_{\text{IsingXY}}(\theta)$	Two-qubit IsingXY interaction gate parameterized by θ	Unitary operator
X, Y, Z	Pauli operators for bit flip, bit-phase-flip, and phase flip errors	Hermitian operators
ρ	Density matrix representing the quantum state	Operator
$P_{\text{logical error rate}}$	Logical error probability after error correction	Probability
p	Physical qubit error probability	Probability
$\mathbb{I}(\cdot)$	Indicator function counting class occurrences (1 if true, else 0)	—

identified the Quantum Information Gap (QIG), which results in information loss between classical and quantum features. The Quantum Information Preserving (QIP) loss function was proposed to reduce this gap, thus improving the effectiveness of QML algorithms. Sakka et al. [15] presented an agentic framework capable of autonomously generating, evaluating, and refining quantum feature maps using large language models. This method revealed that the algorithm was worthy of identifying feature maps that surpassed current quantum baselines and achieved a comparable accuracy relative to classical kernels in datasets such as MNIST.

Numerous studies have investigated QKNN algorithms, demonstrating potential improvements over CKNN methods. Jarir and Quafafou [16] proposed a hybrid quantum-classical KNN approach for text classification, which utilizes quantum algorithms for feature extraction. However, their study did not compare the accuracy of their algorithms with that of the classical methods. Zardini et al. [17] introduced a QKNN algorithm based on the Euclidean distance estimation, which improves computational efficiency in high-dimensional spaces. However, the study lacked explicit accuracy benchmarks.

In addition to KNN, QNN has become a viable approach to enhance classification tasks. Quantum-enhanced neural networks take advantage of QC to significantly improve speed and accuracy. Berti et al. [18] explored the transformation of CKNN algorithms into their quantum equivalents, focusing on two distinct designs, namely 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. [19] applied quantum KNN to handwritten digit recognition, achieving 98% precision in the MNIST dataset, which outperformed the CKNN algorithms that achieved an accuracy of around 94%.

Further developments in QML have seen significant progress in QNN. Dixit et al. [20] have demonstrated the effectiveness of QNN in cancer detection, using quantum principles such as superposition and entanglement to enhance pattern recognition. These features enable QNNs to capture complex relationships and improve classification accuracy in biomedical datasets. Xiang et al. [21] explored hybrid quantum-classical convolutional neural networks (QCCNNs) for breast cancer diagnosis and demonstrated accuracy and computational efficiency improvements over classical approaches. Zhul et al. [22] proposed a fast clustering approach that integrates graph-regularized non-negative matrix factorization with quantum clustering. The objective was to harness the strengths of both techniques, which enhanced the speed and accuracy of the classifier.

Thus, several studies have focused on improving the computational efficiency and scalability of quantum KNN and QNN algorithms. Some of them also demonstrated notable improvements in classification accuracy. However, the QKNN algorithm we proposed improves the classification accuracy much more than the work done by others in the past. A summary of its contribution is highlighted in the Introduction section. The novelty of the proposed QKNN algorithm is described in more detail in Table 4 of Section III.F, and the proposed algorithm in Algorithm 1 is described in the same section.

3 Methodology

This section outlines the methodology used to evaluate the performance of the proposed QKNN algorithm. First, 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: 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 [23].

Iris Dataset: Introduced by Fisher [24], 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 [25]. The basic attributes of the three datasets are shown in Table 2.

Table 2: The basic attributes of the three datasets.

Dataset	Instances	Features	Classes	Reference
Wisconsin Breast Cancer	569	30	2	[23]
Iris	150	4	3	[24]
Bank Note Authentication	1372	4	2	[25]

3.2 Feature Selection Using the Chi-Square Test

Feature selection reduces dimensionality, improves model efficiency, and prevents 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 statistical independence exists 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 [26]. This method has been successfully used in numerous ML applications to select relevant features and enhance model performance [27].

In this study, the UCI breast cancer dataset, composed initially of multiple features, is reduced to the four most statistically significant using the chi-square test for quantum encoding. The remaining two datasets naturally contain four features each. Standardizing the count of features across datasets ensures uniform dimensions of the quantum circuit, facilitating an equitable comparison of experimental results. Employing chi-square feature selection is critical in QML, where qubit availability is a fundamental limitation. Since each feature maps to one or more qubits, a higher dimensionality directly increases quantum resource demands and circuit complexity. By rigorously selecting only the most relevant features, the chi-square test minimizes qubit usage without compromising predictive accuracy. This dimensionality reduction enhances computational tractability on near-term quantum devices and mitigates the noise and decoherence effects inherent to deeper circuits. Thus, chi-square-based feature selection provides a systematic and practical strategy to reconcile the informativeness of the dataset with the constraints of current quantum hardware.

3.3 Quantum Machine Learning

QML is an emerging paradigm that integrates the principles of QC 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 their classical counterparts [6]. These quantum properties allow 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 the encoding of quantum data, where the classical data is mapped to quantum states. Unlike classical representations, quantum encoding allows data to exist simultaneously in a superposition of multiple states, 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 the data. Hybrid quantum-classical models combine QC with classical optimization techniques, showing promising results in classification, clustering, and generative modeling [28]. Given the limitations of noisy intermediate-scale quantum (NISQ) devices, these hybrid models are beneficial, as they 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]. Furthermore, quantum Boltzmann machines and quantum generative adversarial networks (QGANs) have been explored for generative learning tasks such as image synthesis and anomaly detection [29].

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 research areas. One significant bottleneck is quantum data loading, where encoding large classical datasets into quantum states is a non-trivial task [30].

The integration of QC into ML has the potential for significant advances in both theoretical and practical applications. As quantum hardware improves, QML is expected to lead to drug discovery, materials science, and financial modeling breakthroughs, where complex simulations require computational power beyond classical capabilities [2]. 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 previously considered intractable problems.

3.4 Classical vs. Quantum Feature Spaces

In classical ML, data is represented as feature vectors in an n-dimensional Euclidean space given as follows:

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

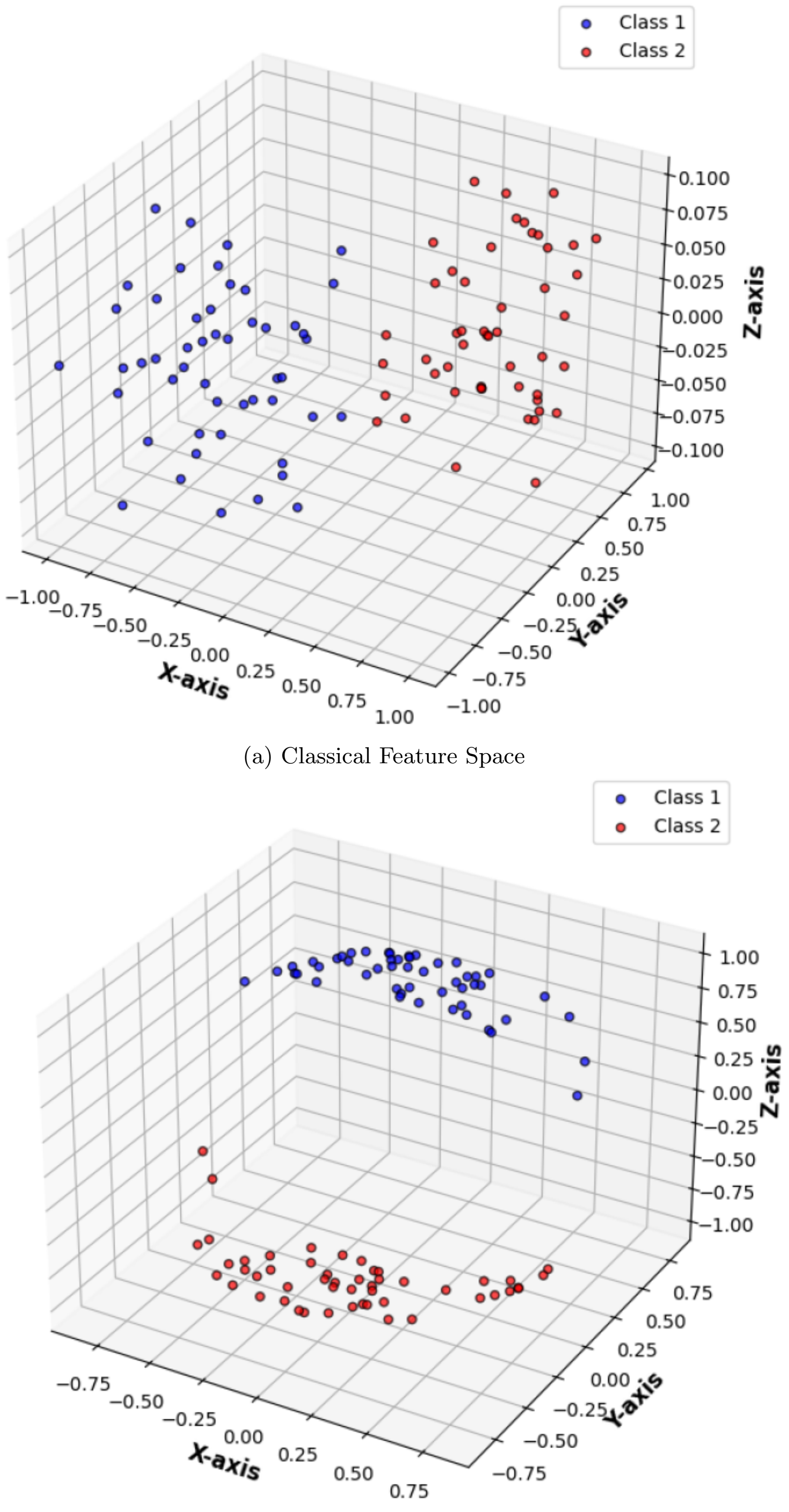
Algorithms, like SVMs, rely on kernel methods to map data into higher-dimensional spaces, allowing for the separation of nonlinearly separable data. However, this mapping increases computational complexity [31].

In contrast, QC encodes data into quantum states via superposition and entanglement. A classical data point $\mathbf{x}$ is assigned to a quantum state represented 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, which makes QML particularly suitable for large, complex datasets. Figure 1 shows classical and quantum feature spaces.



(a) Classical Feature Space

(b) Quantum Feature Space

Figure 1: Comparison between classical and quantum feature spaces.

From Figure 1(a), it is seen that the class 1 and class 2 data points are not separable, while Figure 1(b) shows that they are separable. The quantum encoding technique maps data into an exponentially ample quantum Hilbert space. Thus, it offers a more powerful feature representation technique than the classical methods and leverages exponential dimensionality for better pattern separation.

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, unitary matrices represent quantum gates, ensuring reversibility in quantum operations [32]. These gates manipulate quantum states by leveraging principles such as superposition and entanglement, enabling complex computations that classical computers cannot efficiently simulate.
Table 3 presents an overview of the commonly used quantum gates, their matrix representations, and a brief description of their functionalities.

Table 3: Matrix Representation and Functions of the Common Quantum Gates.

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 [32].
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 [32].
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 [33].
Pauli-Z (Z)	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$	Applies a phase shift of π to the $\|1\rangle$ state [34].
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 $\|1\rangle$ [32].
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 [35].
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 [32].
Phase (S)	$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$	Applies a phase shift of $\pi/2$ to the $\|1\rangle$ state [34].
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 [32].

In constructing quantum algorithms such as Grover's algorithm for unstructured search and Shor's algorithm for integer factorization, quantum gates are fundamental [35] [36]. One everyday use of the Hadamard gate is the construction of superpositions, which enable quantum parallelism. An essential part of quantum processing, the entanglement of qubits, requires the CNOT gate [33]. The Toffoli gate is a key component of fault-tolerant QC design, which is classically applicable to all reversible computations.
In quantum circuits, 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 depends on the effective deployment and enhancement of quantum gates as

the hardware evolves.

3.6 The proposed QKNN algorithm

The proposed QKNN algorithm introduces a novel approach to classification by integrating quantum computational principles into the conventional CKNN framework. This quantum-enhanced method capitalizes on 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 classical methods, particularly for high-dimensional and intricate feature spaces. Figure 2 represents the flow diagram of the proposed QKNN algorithm.

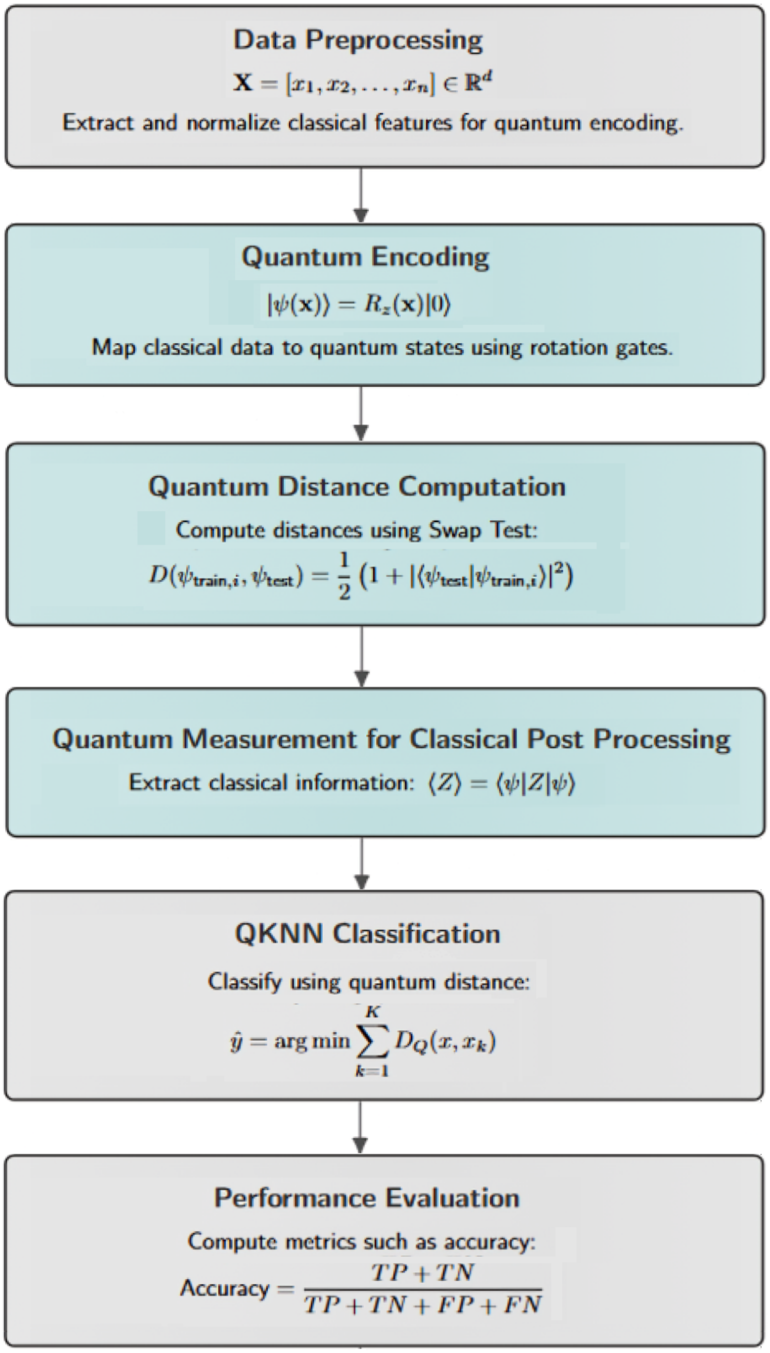

Data Preprocessing
$\mathbf{X} = [x_1, x_2, \ldots, x_n] \in \mathbb{R}^d$
Extract and normalize classical features for quantum encoding.
Quantum Encoding
$|\psi(\mathbf{x})\rangle = R_z(\mathbf{x})|0\rangle$
Map classical data to quantum states using rotation gates.
Quantum Distance Computation
Compute distances using Swap Test:
$D(\psi_{\text{train},i}, \psi_{\text{test}}) = \frac{1}{2}\left(1 + |\langle\psi_{\text{test}}|\psi_{\text{train},i}\rangle|^2\right)$
Quantum Measurement for Classical Post Processing
Extract classical information: $\langle Z\rangle = \langle\psi|Z|\psi\rangle$
QKNN Classification
Classify using quantum distance:
$\hat{y} = \arg\min \sum_{k=1}^{K} D_Q(x, x_k)$
Performance Evaluation
Compute metrics such as accuracy:
$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$

Figure 2: Flow diagram of the Proposed QKNN Algorithm

3.6.1 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 assigned to 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 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:

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

where d is the dimension of the feature vector.

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

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

where $\theta = 2\pi x_i$ and x_i is the normalized i-th feature value in the input vector. As a result, each classical data point is transformed into the quantum representation:

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

Where $\bigotimes$ denotes the tensor product across qubits. This encoding capability enables the quantum system to process multiple data points, improving computational efficiency. Furthermore, the proposed quantum data encoding scheme is a fundamental conceptual advancement by embedding classical data into a high-dimensional quantum feature space that exploits quantum superposition and interference for enhanced expressivity.

3.6.2 Quantum Feature Extraction

Quantum transformations enhance data separability by introducing higher-order correlations and structural modifications within the encoded quantum states. In our QKNN framework, the IsingXY gate establishes entanglement between qubits, enabling richer feature interactions. Formally, the IsingXY gate is defined as:

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

where $\sigma_{\mathbf{x}}^{(\mathbf{k})}$ and $\sigma_{\mathbf{y}}^{(\mathbf{k})}$ denote the Pauli X and Y operators acting on qubit k, respectively.

This gate facilitates the generation of multipartite entanglement, which captures complex and nonlinear correlations among input features. This enhanced expressivity in the quantum feature space supports improved class separability, which is difficult to achieve with classical transformations.

Additionally, Controlled-NOT (CNOT) gates are incorporated to further reduce class overlap by creating controlled entanglement patterns that promote feature space individuality among data points. The overall quantum state transformation can be expressed as:

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

where U denotes the combined unitary operation of the IsingXY and CNOT gates.

The deliberate selection of IsingXY and CNOT gates reflects a theoretical design choice to balance circuit expressivity and noise robustness. This strategic entanglement structure enables our QKNN algorithm to surpass CKNN by creating more discriminative representations of quantum features, thereby significantly enhancing classification performance.

3.6.3 Quantum Distance Calculation and Neighbor Selection

The similarity between a test quantum state $|\psi_{\text{test}}\rangle$ and each training quantum state $|\psi_{\text{train},i}\rangle$ is evaluated using a quantum distance metric based on the well-established swap test, defined as:

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

Although the swap test is a well-established quantum algorithmic technique for estimating the overlap between quantum states, our work introduces a novel integration of the swap test–based distance metric within a comprehensive and scalable QKNN framework.

This approach exploits the ability of the swap test to quantify subtle quantum features, including coherence and entanglement, within the exponentially ample Hilbert space, thereby providing a similarity measure that transcends the limitations of classical distance metrics. Although the swap test itself is not new, our contribution lies in its rigorous theoretical justification and effective incorporation as a fundamental component of a noise-resilient quantum KNN algorithm. This integration bridges the core principles of quantum computing with practical QML implementations optimized for near-term quantum hardware. The nearest neighbors for the test point are selected based on the smallest quantum distances, ensuring optimal classification.

3.6.4 Classification via Quantum Nearest Neighbors

After identifying the nearest neighbors, the 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 the occurrences of each class among the selected neighbors. The class with the highest frequency is assigned to the test instance, ensuring robust classification. Figure 3 shows the proposed QKNN architecture. This architecture provides a novel approach to improve the efficiency of KNN classification by incorporating QC principles.

3.6.5 Rationale and Impact of Gate Selection

The selection of IsingXY, CNOT, and SWAP gates in our QKNN algorithm reflects their complementary roles in improving algorithmic efficiency and hardware compatibility. IsingXY gates implement native two-qubit interactions from the XY-model Hamiltonian, common in superconducting qubits and trapped ions. Their use enables hardware-efficient ansatzes that reduce circuit depth and gate overhead compared to CNOT-only designs, improving expressivity and noise resilience.

Although both CNOT and SWAP gates are multiqubit operations that increase the depth and circuit complexity, they perform distinct and essential functions. CNOT gates enable feature extraction based on entanglement within the circuit, encoding complex data correlations critical to accurate classification. In contrast, SWAP gates facilitate the swap test, a fundamental procedure for assessing quantum state similarity integral to distance estimation and classification in the KNN framework. Additionally, SWAP gates support qubit routing to address hardware connectivity limitations. Despite increasing circuit depth and compilation overhead, the complementary functions of the CNOT gates in feature processing and the SWAP gates in similarity evaluation justify their combined use.

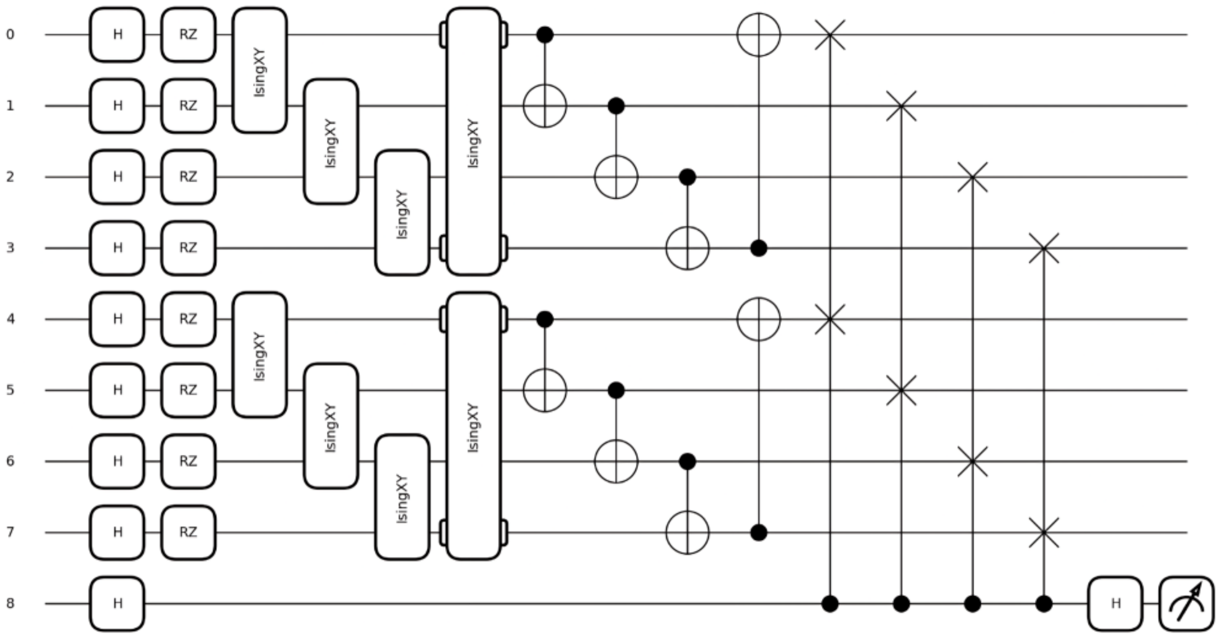


Figure 3: The proposed QKNN architecture.

Together, this set of gates balances expressivity, noise robustness, and hardware adaptability, ensuring effective implementation of the QKNN algorithm in near-term quantum devices with diverse native gates and limited connectivity.

3.6.6 Description of the Proposed QKNN Algorithm

The steps performed in the proposed QKNN algorithm are included in Algorithm 1. To identify the nearest neighbors for classification, the proposed algorithm first takes quantum transformations as input for feature extraction and utilizes quantum distances determined from the swap test.

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

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

Table 4 presents a comparative analysis of various existing QKNN approaches with the proposed QKNN, focusing on key factors such as encoding methods, feature extraction techniques, distance metrics, neighbor selection, and noise mitigation strategies. Our proposed QKNN algorithm introduces significant performance improvements with a simplified quantum circuit and smaller circuit depth compared to existing methods. Our method aims to optimize the performance and scalability of QKNN algorithms, making them more feasible for real-world applications where computational resources and quantum hardware capabilities are constrained.

By leveraging quantum-enhanced distance evaluation and feature transformations, the QKNN algorithm improves classification tasks' efficiency and performance. 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.

Table 4: Comparison between Various Existing QKNN Approaches with the Proposed QKNN

Study	Encoding Method	Feature Extraction	Distance Metric	Neighbor Selection	Noise Mitigation
Gao et al. [37]	Amplitude Encoding with Quantum Random Access Memory (QRAM)	Classical	Quantum Mahalanobis Distance	Grover search based on quantum distance	Not addressed
Zardini et al. [17]	Amplitude Encoding	Classical	Quantum Euclidean Distance	Quantum distance based sorting	Not addressed
Jing Li et al. [38]	Binary Encoding Method	Not addressed	Quantum Hamming Distance	Quantum distance based sorting	Not addressed
Feng et al. [7]	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	Shor Code

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 [39] [40]. Unlike classical neural networks that rely on weighted connections between neurons, QNNs utilize parameterized quantum circuits (PQCs) to process and classify data [2] [41].

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 parameterized unitary operator defined by the quantum circuit with trainable parameters θ [42]. The trainable parameters θ are optimized using classical gradient-based methods, often through a quantum-classical hybrid approach involving variational quantum eigensolver (VQE) or quantum natural gradient methods [43].

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, 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.

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 the preparation of the quantum state. Classical data is embedded

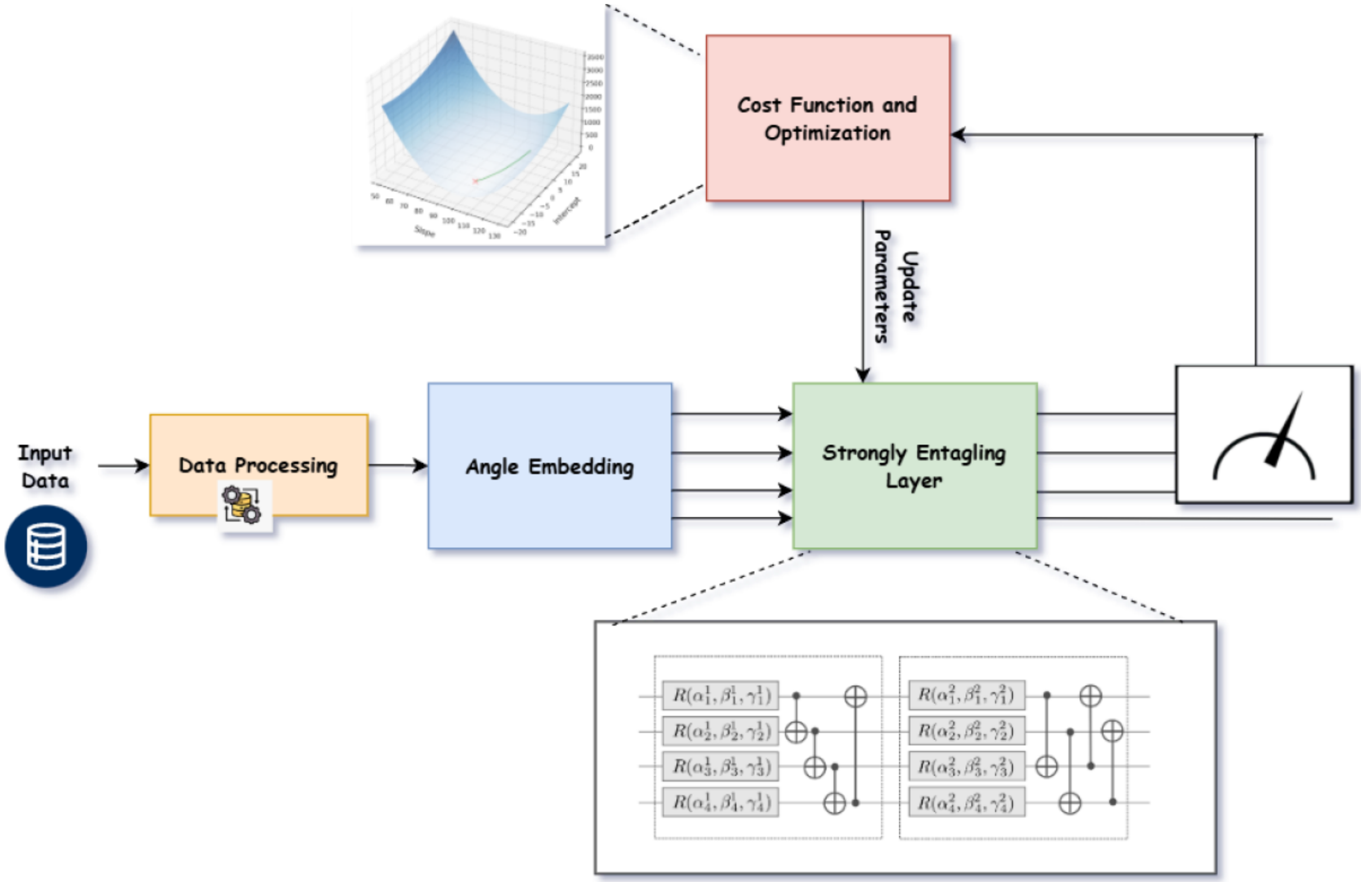


Figure 4: The QNN Architecture with Parameterized Quantum Circuits.

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 $R_Y(x_i) = e^{-ix_iY/2}$ denotes a rotation by angle x_i about the Pauli-Y axis acting on the i-th qubit, and $\mathbf{x} = [x_1, x_2, \ldots, x_d]^\top$ is the classical input feature vector [32]. Importantly, these local rotations act independently on each qubit and do not generate entanglement.

To introduce nonlinearity and entanglement, we employ PennyLane's strongly entangling layer, which consists of parameterized single-qubit R_Z rotations followed by CNOT gates arranged in a nearest-neighbor ring topology. The CNOT gates establish pairwise controlled entanglement between adjacent qubits, which propagates to create multipartite entanglement across the qubit register. This layered entanglement structure enables the quantum neural network to capture complex correlations between input features effectively. The strongly entangling layer is expressed as:

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

where $R_Z(\theta_{l,i}) = e^{-i\theta_{l,i}Z/2}$ is a rotation about the Pauli-Z axis on the qubit i in layer l, being a trainable parameter optimized during training, and $\text{CNOT}_{i,j}$ denotes a controlled NOT gate with the control qubit i and the target qubit j. The modulo operation ensures a ring topology for entanglement across all d qubits.

3.7.1 Quantum Measurement and Output

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

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

where Z_i is the Pauli-Z operator acting on the i-th qubit and C corresponds to the number of classes in the classification task. The measurement results form a vector of expectation values used as features for classification.

3.7.2 Cost Functions for Classification

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

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

where N = total number of samples, $y_i \in \{0,1\}$ is the true label of sample i, $\hat{y}_i \in [0,1]$ is the predicted probability for the positive class.
For multiclass classification, we utilize the categorical cross-entropy (CCE), which 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, $y_{i,j} \in \{0,1\}$ is the true class j of sample i, and $\hat{y}_{i,j} \in [0,1]$ is the predicted probability for class j of sample i. This $\hat{y}_j$ 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 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 affecting the reliability of QML algorithms such as the QKNN algorithm [6] [32]. This section presents a noise model based on Pauli error channels and introduces an error mitigation strategy using Shor code to improve computational stability [44] [45].

3.8.1 Quantum Noise Channels

Quantum noise channels describe the probabilistic evolution of the quantum state ρ under noisy conditions, which affects the precision of quantum computations [32] [46]. The primary sources of noise considered in this study include:

- **Bit Flip Channel (Pauli-X Error):** This channel applies the Pauli-X operator with probability p to the density matrix ρ, modeling bit flip errors as [6]:

$$\mathcal{E}_X(\rho) = (1-p)\rho + pX\rho X^\dagger \tag{19}$$

- **Phase Flip Channel (Pauli-Z Error):** This channel induces phase errors, applying the Pauli-Z operator with probability p [46]:

$$\mathcal{E}_Z(\rho) = (1-p)\rho + pZ\rho Z^\dagger \tag{20}$$

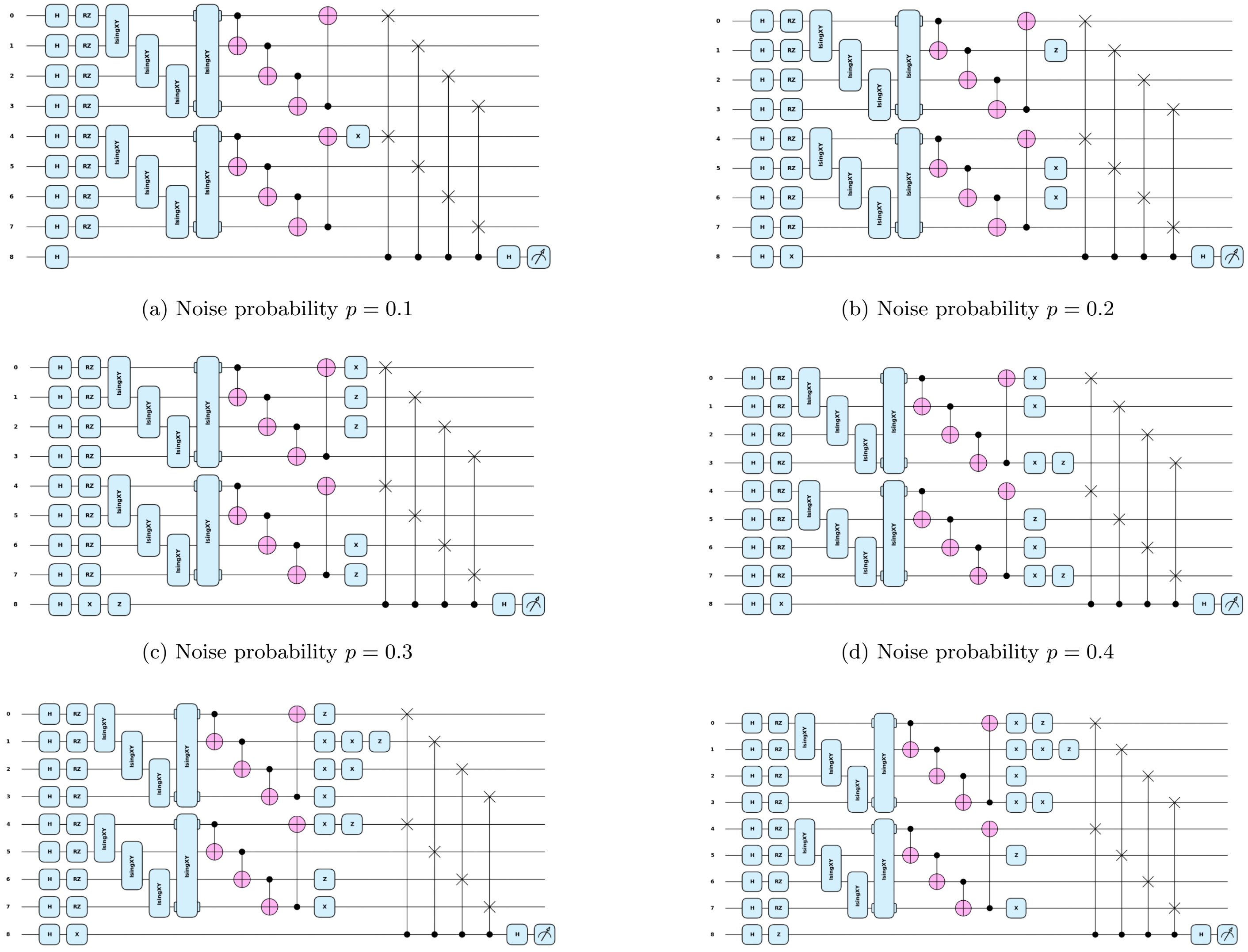
- **Bit-Phase Flip Channel (Pauli-Y Error):** This channel simultaneously introduces bit flip and phase flip errors using the Pauli-Y gate [32]:

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

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] [44]. 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 for various noise models with increasing probability, 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: The quantum circuits for various noise models with increasing probability.

3.8.2 Error Mitigation via Shor Code

To address the challenges of quantum noise, we employ the Shor code, a quantum error correction (QEC) scheme that encodes logical qubits into a larger number of physical qubits, protecting against bit-flip and phase-flip errors [47] [32]. Unlike simpler codes such as the n-qubit repetition code, the Shor code simultaneously protects against both error types by combining bit-flip and phase-flip codes.

Encoding and Error Detection: A logical qubit is encoded using nine physical qubits in Shor code, arranged as follows:

$$|0\rangle_L = |000\rangle \otimes |000\rangle \otimes |000\rangle \tag{23}$$

$$|1\rangle_L = |111\rangle \otimes |111\rangle \otimes |111\rangle \tag{24}$$

where $|0\rangle_L$ and $|0\rangle_L$denote the logical qubit states. This encoding involves first applying a bit-flip code (repetition code) to encode the logical qubit into three qubits, followed by a phase-flip code for each of these three qubits, thus utilizing nine physical qubits to encode a single logical qubit.
Errors are detected and corrected using stabilizer measurements, which involve applying bit-flip and phase-flip checks. The stabilizer measurements for the Shor code are represented as:

$$S_1 = X_1X_2, \quad S_2 = X_2X_3, \quad S_3 = X_4X_5, \quad S_4 = X_5X_6$$

$$S_5 = X_7X_8, \quad S_6 = X_8X_9 \tag{25}$$

$$S_7 = Z_1Z_2Z_3Z_4Z_5Z_6, \quad S_8 = Z_4Z_5Z_6Z_7Z_8Z_9 \tag{26}$$

Here, X_i and Z_i are Pauli-X and Pauli-Z operators on qubit i. Stabilizers S_1 to S_6 detect bit-flip errors (X_i) within each three-qubit block (qubits 1–3, 4–6, 7–9), while S_7 and S_8 detect phase-flip errors (Z_i) across blocks. The parity check matrices are:

$$H_X = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \tag{27}$$

$$H_Z = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{28}$$

where rows of H_X correspond to S_1 to S_6, checking Z-type errors (bit-flips), and rows of H_Z correspond to S_7, S_8, checking X-type errors (phase-flips). The code is a $[9, 1, 3]$ CSS code, encoding 1 logical qubit with distance $d = 3$, as determined by the minimum weight of non-trivial operators in the normalizer of the stabilizer group [48].

Error Correction: Error correction in Shor's code leverages syndrome measurements from the eight stabilizers S_1 to S_8 to identify and correct single-qubit bit-flip (X) and phase-flip (Z) errors, ensuring robust operation of the QKNN algorithm on noisy quantum hardware. The stabilizers generate an 8-bit syndrome vector $(s_1, s_2, \ldots, s_8)$, where $s_i = +1$ or -1 corresponds to the eigenvalue of S_i. The syndrome is computed using ancillary qubits and controlled operations, preserving the logical state of the $[9, 1, 3]$ CSS code. The syndromes for single-qubit errors are unique, enabling precise identification of error type and location, as shown in Table 5.
Errors are corrected through syndrome measurements based on the stabilizer values. Using these measurements, the error syndrome is analyzed and the appropriate correction is applied:

- **Bit-flip errors**: Stabilizers S_1 to S_6 detect bit-flip errors within each three-qubit block (qubits 1-3, 4-6, 7-9). For example, a bit-flip error X_1 on qubit 1 produces the syndrome $(s_1, s_2, \ldots, s_6, s_7, s_8) = (-1, +1, +1, +1, +1, +1, +1, +1)$, as X_1 anticommutes with $S_1 = X_1X_2$ but commutes with all others. The syndrome corresponds to the first column of H_X, uniquely identifying qubit 1. The correction applies the Pauli operator X_1. Similarly, the syndromes for other qubits (e.g. $(-1, -1, +1, +1, +1, +1,$ for X_2) are decoded using the block stabilizer pair.

Table 5: Syndromes for Single-Qubit Errors in Shor's Code

Error	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8
X_1	-1	+1	+1	+1	+1	+1	+1	+1
X_2	-1	-1	+1	+1	+1	+1	+1	+1
X_3	+1	-1	+1	+1	+1	+1	+1	+1
X_4	+1	+1	-1	+1	+1	+1	+1	+1
X_5	+1	+1	-1	-1	+1	+1	+1	+1
X_6	+1	+1	+1	-1	+1	+1	+1	+1
X_7	+1	+1	+1	+1	-1	+1	+1	+1
X_8	+1	+1	+1	+1	-1	-1	+1	+1
X_9	+1	+1	+1	+1	+1	-1	+1	+1
Z_1, Z_2, Z_3	+1	+1	+1	+1	+1	+1	-1	+1
Z_4, Z_5, Z_6	+1	+1	+1	+1	+1	+1	-1	-1
Z_7, Z_8, Z_9	+1	+1	+1	+1	+1	+1	+1	-1

- **Phase-flip errors**: Syndromes from S_7, S_8 identify the block with a phase flip. For example, $S_7 = -1, S_8 = +1$ indicates a phase flip in the first block, corrected by applying Z_i on any qubit.
- **No error**: If all syndromes are $s_i = +1$, the state is in the codespace, and no correction is applied.

The Shor code approach provides error detection and correction, significantly reducing the logical error rate compared to the physical qubit error rate [44]. Although the Shor code corrects both bit-flip and phase-flip errors, it introduces overhead due to the increased circuit depth and gate operations, potentially leading to additional errors from the larger number of gates [49]. Despite this, Shor's code remains a powerful tool for quantum error correction, especially for quantum machine learning (QML) algorithms, ensuring error-resilient quantum state preparation and measurement in noisy environments. Its integration into the QKNN framework enhances the reliability of quantum state classification on NISQ devices. Future research may explore more efficient error correction strategies, such as surface codes, which offer better scalability for larger quantum systems [50].

4 Result and Discussion

In this part, we thoroughly evaluate the classification capabilities of QKNN, CKNN, and QNN on the three standard datasets: Breast Cancer, Iris, and Banknote Authentication. The review provides a comprehensive picture of the success of each model by including essential performance parameters, such as accuracy, precision, recall, F1 score, and AUC. Furthermore, to evaluate the model's stability, we investigate how quantum noise affects the performance of QKNN.

4.1 Comparative Performance Analysis Across Models

A comparative performance of the proposed QKNN, QNN, and CKNN classification accuracies is presented in Table 6 for the three test datasets. The proposed QKNN consistently outperforms or matches CKNN, showcasing its superior classification capabilities in diverse datasets. Although the QNN model exhibits competitive performance in several datasets, its overall accuracy is somewhat limited compared to that of QKNN. This discrepancy arises primarily from the inherent challenges in optimizing variational quantum circuits, which form the core of QNN architectures. Such optimization difficulties are especially pronounced in scenarios with limited training data, where the model struggles to converge to optimal parameter settings. Consequently, QNN may not fully realize its potential in these contexts, resulting in comparatively lower classification accuracy.

Table 6: Classification Accuracies of the Proposed QKNN, QNN, and CKNN Algorithms for the Three Test 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 results demonstrate that the proposed QKNN algorithm consistently outperforms CKNN, particularly in the Wisconsin Breast Cancer dataset. Specifically, QKNN achieves an accuracy of 98.25%, surpassing both CKNN (92.98%) and QNN (97.17%). This superior performance is attributed to using entangling gates such as IsingXY and CNOT, which facilitate richer quantum state representations and enhance the model's capacity to discriminate complex and overlapping class boundaries. In particular, QNN also outperforms CKNN in this dataset, suggesting that variational quantum circuits can effectively capture the data's salient features despite optimization's inherent challenges.
For the Iris dataset, both QKNN and CKNN achieve a perfect classification accuracy of 100%, indicating the inherent linear separability of the dataset. In contrast, QNN performs significantly worse, attaining an accuracy of only 83.33%. This disparity suggests that QNN struggles to capture the underlying feature space effectively, likely because of the limitations of quantum variational circuits when applied to 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. For better visualization, the comparative performance analysis included in Table 6 is also shown in the bar diagram of Figure 6.

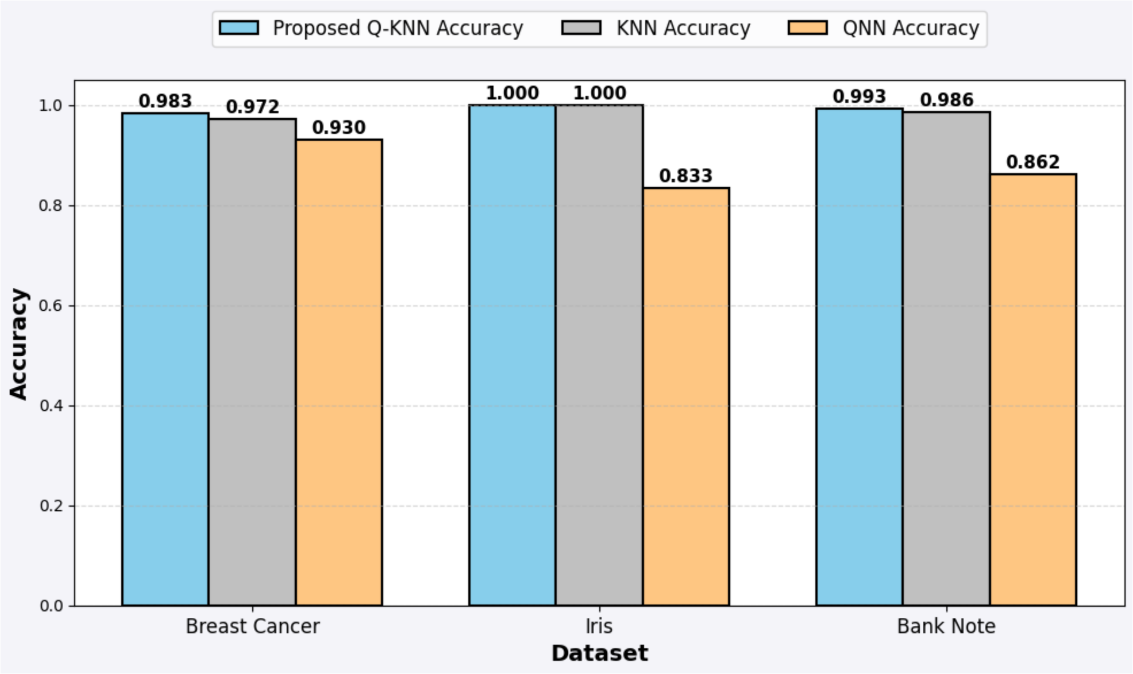


Figure 6: Comparative Performance of QKNN, CKNN, and QNN across the three datasets.

4.2 Key Performance Metrics and Their Implications

To further assess the efficiency of the proposed QKNN algorithm, we analyze additional performance metrics, such as AUC, F1 score, precision, and recall, which are shown in Table 7. These metrics provide deeper insight into the reliability of the classification beyond accuracy.

Table 7: Additional Performance Metrics of the Proposed QKNN Algorithm.

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 consistently high AUC values and balanced F1 scores demonstrate that QKNN possesses strong discriminative power and effectively balances precision and recall in all datasets, minimizing false positives and false negatives. This balance is critical in high-stakes applications such as medical diagnosis and fraud detection, where misclassification entails severe consequences. These superior results stem from the quantum-enhanced feature encoding and distance computation strategies employed by QKNN, which yield richer data representations and more precise similarity assessments than classical approaches. In particular, entangling gates improve class separability within the quantum feature space. Together, these performance metrics affirm the robustness and reliability of QKNN in noisy and complex data environments.

4.3 Confusion Matrices for Model Performance

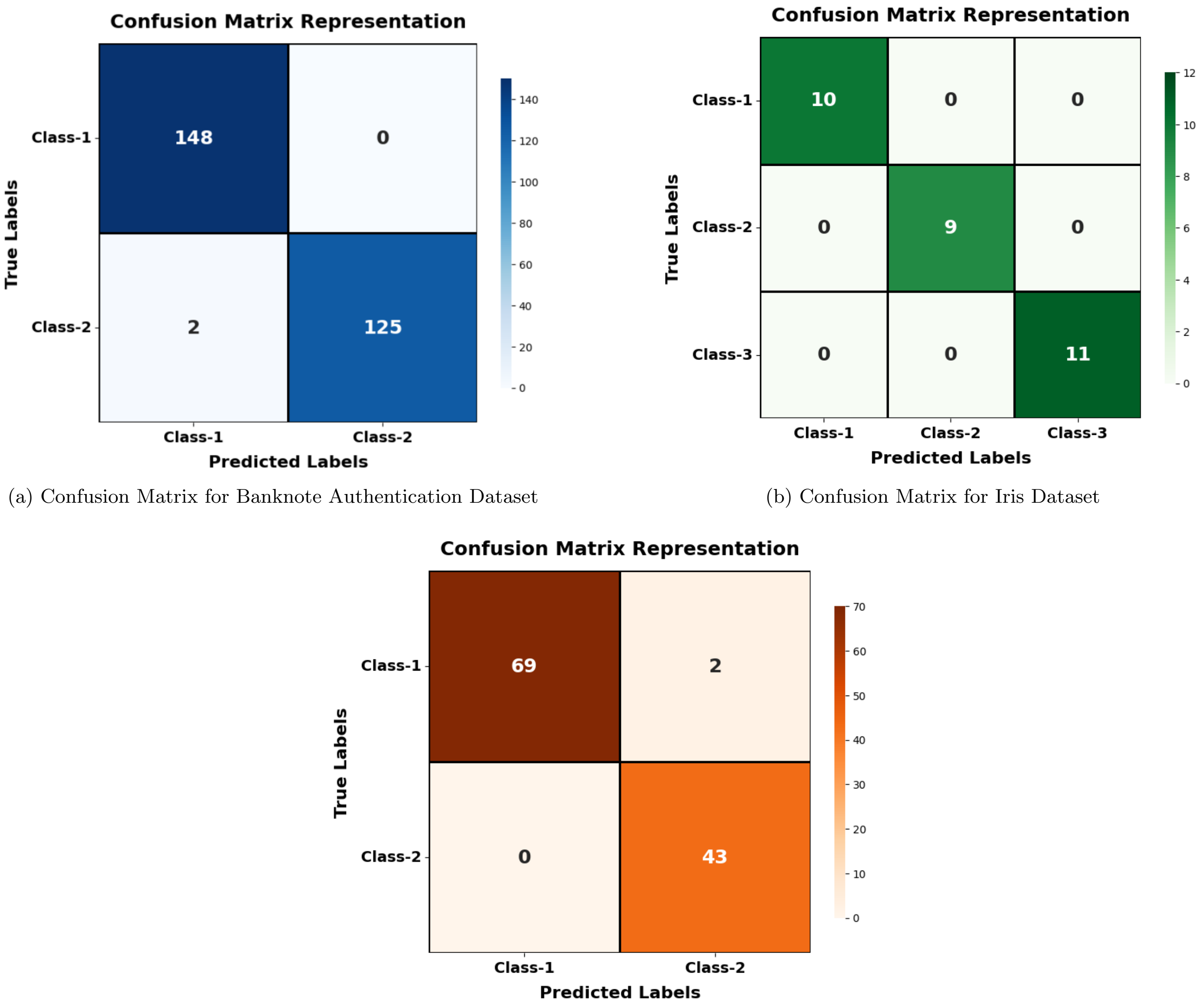
Confusion matrices provide additional insight into the ability of a model to distinguish between true positives, false positives, true negatives, and false negatives. The confusion matrices for the three datasets are shown in Figure 7. The visualizations enable a granular understanding of the model's performance beyond the aggregate metrics.

The confusion matrices show that QKNN consistently achieves high true positive and true negative rates while minimizing false positives and false negatives, particularly on complex datasets such as Breast Cancer and Banknote Authentication. This confirms the strong discriminative capacity of the model and its reliable and balanced classification performance, highlighting its potential for practical high-stakes quantum machine learning applications.

4.4 Benchmarking Against Existing Quantum KNN Approaches

We benchmark the proposed Quantum k-Nearest Neighbors (QKNN) algorithm against recent state-of-the-art quantum KNN methods using three widely recognized datasets: Wisconsin Breast Cancer, Iris, and Bank Note Authentication. However, we found a limited availability of comparable studies on these specific datasets during benchmarking, which constrained direct performance comparisons with existing literature.

The proposed QKNN algorithm achieves classification accuracies of 98.25%, 100.00%, and 99.27% for the Wisconsin Breast Cancer, Iris, and Bank Note Authentication datasets, respectively. The high performance has been achieved by enhancing the feature encoding using the superposition and entanglement phenomena and exploiting the distance computation. For performance comparison, Table 8 shows the classification accuracies of the proposed QKNN algorithm along with those reported in recent QKNN studies by Feng et al. [7], Maldonado et al. [8], Li et al. [9], and Bhaskaran et al. [10].



(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 the Three Datasets.

Table 8: Performance Comparison of the Proposed QKNN Algorithm with Existing Studies

Reference	Wisconsin Breast Cancer	Iris	Bank Note Authentication
Feng et al. [7]	–	95.82%	–
Meldonado et al. [8]	–	94%	87.50%
Li et al. [9]	–	–	–
Bhaskaran et al. [10]	93.85%	93.33%	–
Proposed QKNN	**98.25%**	**100%**	**99.27%**

The performance of the proposed QKNN algorithm exceeds that of the existing QKNN algorithms for the three test datasets. The former improves the accuracy by 4.40% over the QKNN algorithm by Bhaskaran et al. for the Wisconsin Breast Cancer dataset. It achieves the perfect classification accuracy for the Iris dataset, outperforming all previously reported quantum benchmarks. Furthermore, in the Bank Note Authentication dataset, our approach establishes a new state-of-the-art quantum performance, improving the performance of Maldonado et al. by 11.77%. These results highlight the effectiveness of our quantum-enhanced feature encoding and noise-resilient distance metric within the quantum computational framework. This comprehensive evaluation demonstrates the significant advancement that the proposed QKNN algorithm offers in QML classification.

4.5 Effect of Quantum Noise on QKNN Performance

QC systems are susceptible to noise, which can hurt classification performance. Figure 8 illustrates the effect of increasing the noise probability on the accuracy of the proposed QKNN algorithm.

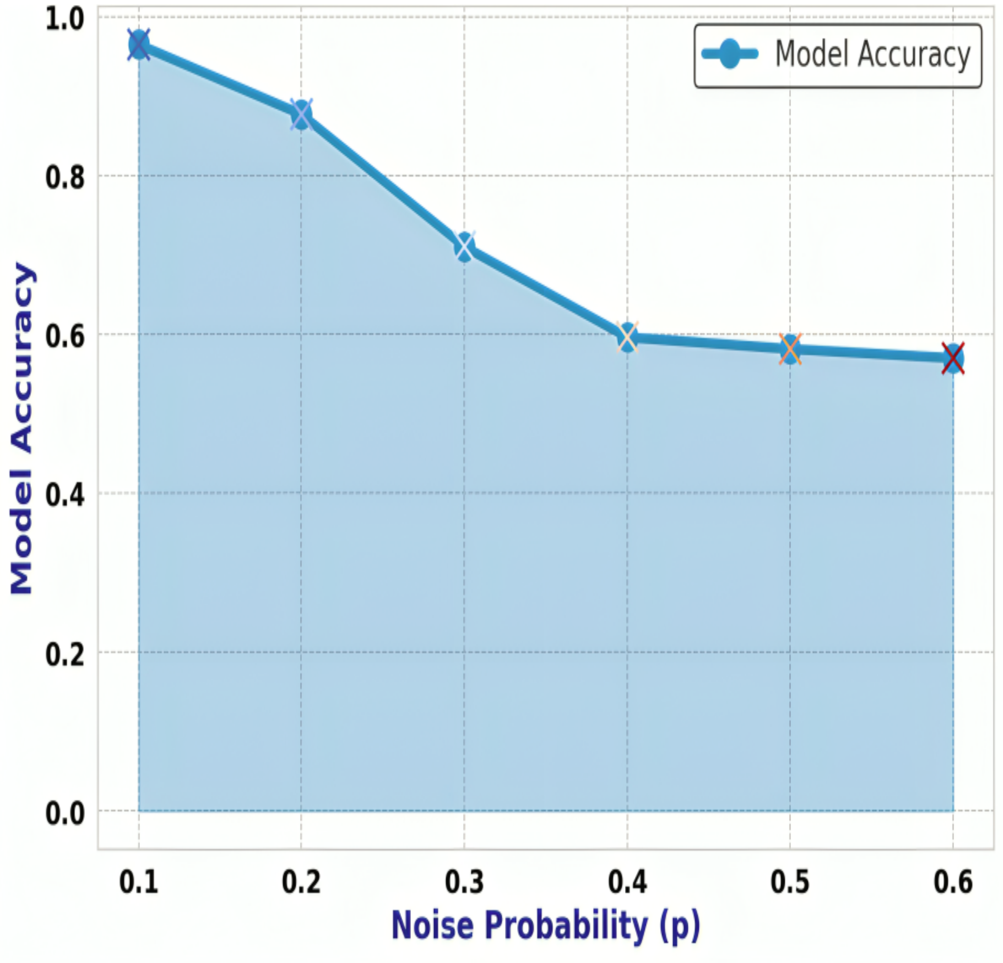


Figure 8: Impact of Noise Probability on Model Accuracy.

The noise probability p is varied from 0 to 0.6 to observe its impact on the model's performance. 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 model's sensitivity 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 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 QML models' challenges in noisy quantum environments.

5 Conclusion

This study presents a novel QKNN algorithm as a quantum-enhanced classification method. It evaluates its performance against CKNN and QNN in three test datasets: Wisconsin Breast Cancer, Iris, and Bank Note Authentication. The experimental results show that the proposed QKNN algorithm outperforms or matches the CKNN, particularly excelling in datasets with complex decision boundaries. Specifically, QKNN achieves an accuracy of 98.25% in the breast cancer dataset compared to 92.98% in CKNN and achieves 99.27% accuracy in the banknote dataset, outperforming CKNN at 98.55%. The results demonstrate the ability of QKNN to handle complex and high-dimensional classification tasks better. Again, for the Iris dataset, both QKNN and CKNN achieved a perfect classification accuracy of 100%. At the same time, QNN lagged behind at 83.33%, suggesting that variational quantum circuits may not be very effective for specific classification tasks without extensive parameter optimization. Furthermore, benchmarking against recent QKNN studies shows that our approach improves accuracy by up to 11.77% over the former. Thus, the proposed QKNN may be a new state-of-the-art technique for the standard datasets' quantum K-nearest neighbors classification method.

Another critical aspect of this study was to analyze the consequences of quantum noise on QKNN accuracy. It should be noted here that the results reveal a strong inverse correlation between noise probability and model accuracy, with performance sharply declining 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 further emphasize the need for incorporating quantum error correction strategies to improve the model's performance 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, several avenues need to be explored to move from theoretical promise to practical deployment. 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 enhance QKNN performance. Additionally, empirical validation on real quantum hardware will provide valuable insights into understanding quantum computational constraints, guiding future model refinements. Finally, integrating QKNN into 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 algorithms may revolutionize data-driven decision-making in high-impact industrial applications.

Acknowledgements
This research has been carried out in joint collaboration with the Department of Electrical and Electronic Engineering of the Bangladesh University of Engineering and Technology and the Department of Computer Science and Engineering of Daffodil International University. The authors thankfully acknowledge the support of the personnel of the Photonics Laboratory for letting them use the computing facility.

References

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

[2] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, S. Lloyd, *Nature* **2017**, *549*, 7671 195.

[3] N. Wiebe, A. Kapoor, K. M. Svore, *Quantum Information and Computation* **2015**, *15*, 3-4 316.

[4] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, J. M. Gambetta, *Nature* **2019**, *567*, 7747 209.

[5] S. Lloyd, M. Mohseni, P. Rebentrost, *arXiv preprint arXiv:1307.0411* **2013**.

[6] J. Preskill, *Quantum* **2018**, *2* 79.

[7] C. Feng, B. Zhao, X. Zhou, X. Ding, Z. Shan, *Entropy* **2023**, *25*, 1 127.

[8] A. Maldonado-Romo, J. Y. Montiel-Pérez, V. Onofre, J. Maldonado-Romo, J. H. Sossa-Azuela, *Mathematics* **2024**, *12*, 12 1872.

[9] J. Li, J. Zhang, J. Zhang, S. Zhang, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **2023**, *43*, 5 1332.

[10] P. Bhaskaran, S. Prasanna, *Knowledge and Information Systems* **2025**, *67*, 1 767.

[11] Z. Yang, M. Zolanvari, R. Jain, *IEEE Communications Surveys & Tutorials* **2023**, *PP*, 99 1.

[12] R. D. Ranga, A. Rana, S. Prajapat, P. Kumar, K. Kumar, A. V. Vasilakos, *Mathematics* **2024**, *12*, 21 3318.

[13] H. Yano, Y. Suzuki, K. M. Itoh, R. Raymond, N. Yamamoto, *arXiv preprint arXiv:2005.14382* **2020**.

[14] X. B. Nguyen, H. Q. Nguyen, H. Churchill, S. U. Khan, K. Luu, *arXiv preprint arXiv:2405.19725* **2024**.

[15] K. Sakka, K. Mitarai, K. Fujii, *arXiv preprint arXiv:2504.07396* **2025**.

[16] Z. Jarir, M. Quafafou, *Advanced Quantum Technologies 2024*.

[17] E. Zardini, E. Blanzieri, D. Pastorello, *Quantum Machine Intelligence 2024*.

[18] A. Berti, A. Bernasconi, G. M. Del Corso, R. Guidotti, *Quantum Machine Intelligence* **2024**, *6*, 2 62.

[19] Y. Wang, R. Wang, *Springer* **2019**.

[20] V. Dixit, K. Rajkumar, *The Journal of Supercomputing* **2025**, *81*, 7 822.

[21] Q. Xiang, D. Li, Z. Hu, Y. Yuan, Y. Sun, Y. Zhu, Y. Fu, Y. Jiang, X. Hua, *Scientific Reports* **2024**, *14*, 1 24699.

[22] R. Zhul, G. Li, J.-X. Liu, L.-Y. Dai, S. Yuan, Y. Guo, In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, **2018** 1610–1613.

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

[24] R. A. Fisher, *Annals of Eugenics* **1936**, *7*, 2 179.

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

[26] G. Forman, *Journal of Machine Learning Research* **2003**, *3* 1289.

[27] J. Cai, J. Luo, S. Wang, S. Yang, *Neurocomputing* **2018**, *300* 70.

[28] M. Schuld, A. Bocharov, K. M. Svore, N. Wiebe, *Physical Review A* **2020**, *101*, 3.

[29] P.-L. Dallaire-Demers, N. Killoran, *Physical Review A* **2018**, *98*, 1 012324.

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

[31] T. Hofmann, B. Schölkopf, A. J. Smola **2008**.

[32] M. A. Nielsen, I. L. Chuang, *Quantum computation and quantum information*, Cambridge university press, **2010**.

[33] D. Deutsch, *Proceedings of the Royal Society of London A* **1989**, *400*, 1818 97.

[34] R. P. Feynman, *Foundations of Physics* **1986**, *16*, 6 507.

[35] P. W. Shor, *SIAM Journal on Computing* **1997**, *26*, 5 1484.

[36] L. K. Grover, *Proceedings of the 28th Annual ACM Symposium on Theory of Computing* **1996**, 212–219.

[37] L.-Z. Gao, C.-Y. Lu, G.-D. Guo, X. Zhang, S. Lin, *Frontiers in Physics* **2022**, *10* 1047466.

[38] J. Li, S. Lin, K. Yu, G. Guo, *Quantum Information Processing* **2022**, *21*, 18.

[39] M. Schuld, I. Sinayskiy, F. Petruccione, *Quantum Information Processing* **2014**, *14*, 7-8 796.

[40] K. Mitarai, M. Negoro, M. Kitagawa, K. Fujii, *Physical Review A* **2018**, *98*, 3 032309.

[41] R. Zhao, S. Wang, *arXiv preprint arXiv:2109.01840* **2021**.

[42] M. Schuld, F. Petruccione, *Machine learning with quantum computers*, volume 676, Springer, **2021**.

[43] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, P. J. Coles, *Nature Reviews Physics* **2021**, *3*, 9 625–644.

[44] S. J. Devitt, W. J. Munro, K. Nemoto, *Reports on Progress in Physics* **2013**, *76*, 7 076001.

[45] D. A. Lidar, T. A. Brun, *Quantum error correction*, Cambridge university press, **2013**.

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

[47] P. W. Shor, *Physical Review A* **1995**, *52*, 4 R2493.

[48] D. Gottesman, *Physical Review A* **1996**, *54*, 3 1862.

[49] B. M. Terhal, *Reviews of Modern Physics* **2015**, *87*, 2 307.

[50] D. Gottesman, Ph.D. thesis, California Institute of Technology, **1997**.