State of the Field: Practical Evaluation and Emerging Directions for Hybrid Quantum Neural Networks

Andrew Nerud

Abstract—**Hybrid Quantum Neural Networks (HQNNs) combine quantum circuits with classical deep learning architectures to improve model efficiency and generalization in high-dimensional tasks. While promising, HQNN deployment faces challenges such as noise, simulator bottlenecks, and hybrid optimization complexity. This paper assesses HQNN viability through direct experimentation on handwritten digit classification and comparative analysis across domains including medical imaging, chemistry, and NLP. Results show HQNNs can reduce parameter counts by up to 50%, decrease floating-point operations by 30%, and maintain or surpass classical model accuracy, albeit with longer training times. A review of recent studies further highlights trends in hardware innovation, quantum security research, and emerging application areas. While HQNNs are not yet viable replacements for classical models, they offer a crucial step toward practical quantum machine learning, particularly in resource-constrained and data-efficient environments.**

CONTENTS

LIST OF FIGURES

List of Tables

I. Introduction

In 2023, researchers at MIT demonstrated that HQNNs could classify medical images with 98.7% accuracy while using 30% fewer parameters than traditional CNNs [6]. This breakthrough highlights the potential of quantum-assisted deep learning to improve model efficiency while maintaining high accuracy.

HQNNs use **superposition and entanglement** to enhance feature extraction and reduce computational overhead. This paper reviews HQNN performance relative to classical models. We systematically review existing findings regarding:

- The computational efficiency of HQNNs versus classical neural networks.
- The impact of quantum feature representations on training performance.
- The practical limitations and challenges observed in experimental HQNN studies.

Beyond technical performance, HQNNs raise important implications in industry, ethics, and long-term viability. As quantum computing gradually shifts from academic novelty to industrial investment, HQNNs are emerging as candidates for practical AI in sectors like healthcare [11], [14], cybersecurity [15], materials science [12], [16], and environmental modeling [17]. These systems offer faster inference and lower complexity but raise equity, explainability, and access concern.

For example, privacy-aware HQNNs have been proposed for biometric tasks such as lipreading, where classical DNNs pose risks to user anonymity [18]. Meanwhile, HQNN-based pipelines are already outperforming classical models in resource-constrained tasks like corrosion inhibitor discovery [16] and battery health estimation [19], suggesting real-world feasibility even under current quantum hardware limitations.

Given this momentum, an important question arises: **Can HQNNs scale into reliable, general-purpose AI tools beyond academic benchmarks?** And if so, how should the computer science community prepare for their broader integration?

This paper examines the feasibility and limitations of HQNNs by systematically reviewing their computational efficiency, training dynamics, and real-world applications. It also expands upon the social and ethical framing of this technology, drawing from recent advancements and ongoing debates. To support this analysis, the following section first provides a foundational overview of quantum mechanics and its intersection with machine learning.

II. Background and Theoretical Foundations

A. *Quantum Computing and Machine Learning*

Quantum computing uses quantum mechanics to perform computations beyond classical systems [13]. Unlike classical bits constrained to binary states (0 or 1), quantum bits (*qubits*) exist in a superposition of both states, enabling exponential computational advantages in specific domains [20].

The qubit, as the fundamental unit of quantum information, is mathematically represented as:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad \text{where } \alpha, \beta \in \mathbb{C}, \quad |\alpha|^2 + |\beta|^2 = 1 \tag{1}$$

Here, α and β are complex probability amplitudes, and their squared magnitudes sum to 1. This normalization ensures that a measurement will always collapse the qubit into a definite state of $|0\rangle$ or $|1\rangle$. Multi-qubit states are expressed as tensor products of individual qubits, which enables the encoding of quantum correlations [9], [13]. The key quantum properties and definitions relevant to computing will be covered in the following sections.

Key Quantum Terms: A Practical Glossary

To improve clarity, we define essential terms relevant to HQNNs:

- **Qubit:** A quantum bit that can exist in a superposition of 0 and 1, enabling parallel information processing [13].
- **Superposition:** A quantum property allowing simultaneous existence in multiple states, giving rise to computational parallelism [21].
- **Entanglement:** A phenomenon where two or more qubits are interdependent; a change in one affects the state of the other. This is critical for learning feature correlations [13].
- **Quantum Gate:** A transformation applied to a qubit, analogous to classical logic gates. Gates like Hadamard and CNOT manipulate quantum states during computation [9].
- **Variational Quantum Circuit (VQC):** A parameterized quantum model trained via optimization, serving as the core quantum layer in HQNNs [1].
- **Hilbert Space:** A high-dimensional vector space where quantum states live. Feature encodings into Hilbert space enable HQNNs to capture complex data patterns [2].
- **Quantum Kernel:** A similarity measure computed in quantum feature space. Used in tasks like classification with quantum-enhanced SVMs [9].
- **Measurement:** The act of collapsing a quantum state into classical information. This step is probabilistic and introduces latency in hybrid systems [3].

1) Entanglement: Entanglement is a quantum phenomenon that links qubits such that measuring one immediately determines the other's state [13], [21]. This correlation is independent of distance, enabling non-local information encoding and efficient representation of feature dependencies in HQNNs.

A two-qubit entangled state, known as a Bell state, is represented as:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{2}$$

This state forms the theoretical basis for quantum teleportation, quantum key distribution, and improved expressivity in variational quantum circuits.

2) Hadamard Gate: The Hadamard gate (H) is a foundational single-qubit gate that places a qubit into an equal superposition of states. It is often used at the beginning of quantum algorithms to initiate parallel exploration of solution spaces.

Mathematically, it is defined as:

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{3}$$

Applied to the initial basis state $|0\rangle$, it produces:

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

which encodes both computational paths simultaneously for later quantum operations.

3) Controlled-NOT gate: The Controlled-NOT (CNOT) gate is a two-qubit quantum operation that flips the target qubit if the control qubit is in the $|1\rangle$ state. It is a key component in generating entangled states within quantum circuits.

Its matrix form is:

$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{5}$$

When applied to a superposed input state:

$$CNOT\left(\frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)\right) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle), \tag{6}$$

it produces the Bell state from Equation 2, thus entangling the two qubits.

Quantum Machine Learning (QML) explores how quantum computing can accelerate machine learning tasks, such as classification, clustering, and generative modeling [21].

B. Introduction to HQNNs

HQNNs integrate quantum computing layers within classical deep learning models, aiming to enhance computational efficiency while leveraging quantum properties [9], [22]. These models use **variational quantum circuits (VQCs)** that leverage entanglement and superposition for feature extraction.

As shown in **Figure 1**, HQNNs leverage an **N-qubit variational circuit** where each qubit undergoes Hadamard transformations (H), controlled interactions, and parameterized rotation gates ($R_y(\theta)$) [4]. This setup enables feature representations beyond classical models, improving pattern recognition. VQCs are typically trained using hybrid optimization methods, where classical optimizers adjust the quantum gate parameters (θ) based on loss minimization techniques [1].

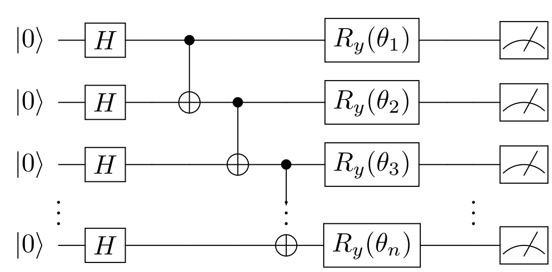


Fig. 1: One of the various forms of a N-qubit VQCs used in HQNNs. Each qubit begins in $|0\rangle$ and undergoes Hadamard gates, controlled operations, and $R_y(\theta)$ rotations. Adapted from [1], [2], [3], [4], [5].

HQNN Architecture Overview:

- A classical neural network is used for feature extraction and data preprocessing.
- A quantum layer (often implemented using VQCs) replaces one or more classical layers [1].
- The quantum circuit is parameterized and trained using gradient-based optimization, similar to classical deep learning models [23].

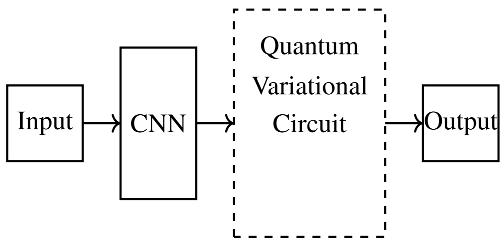


Fig. 2: General architecture of a HQNN. The model integrates classical convolutional layers with a quantum variational circuit before producing the final output [6], [7], [8].

HQNNs follow a hybrid structure where:

- A **classical neural network** is used for initial feature extraction and preprocessing.
- A **quantum variational circuit** replaces certain layers within the model, leveraging quantum gates to process data [2].
- Optimization is performed through **hybrid training methods**, combining classical gradient-based techniques with quantum variational parameter tuning [9].

Figure 2 provides an overview of a typical HQNN architecture, where a convolutional neural network (CNN) extracts features before processing them through a quantum variational circuit. This hybrid approach enables improved feature transformations while maintaining classical efficiency.

C. Quantum Feature Representations in HQNNs

Recent research shows quantum encodings can outperform classical embeddings in **high-dimensional tasks** with redundancy or sparsity, which often pose challenges to conventional deep learning approaches [24]. Quantum feature representations **efficiently encode high-dimensional data**, reducing computation and overfitting.

Other recent experimental studies have demonstrated the practical applications of HQNNs in real-world scenarios. For instance, research on **handwritten digit recognition** has shown that HQNNs can achieve comparable accuracy to classical convolutional neural networks (CNNs) while requiring substantially fewer trainable parameters, thereby reducing computational complexity [10]. Similarly, in the domain of **quantum chemistry**, HQNN-based approaches have been employed to predict the ground state energy of molecular systems with improved precision over classical machine learning models [2]. By directly encoding quantum states into the network, HQNNs offer a fundamental advantage in processing quantum mechanical data, making them well-suited for applications in materials science and molecular modeling.

Despite these advantages, the effectiveness of HQNNs is contingent on the efficient design of VQCs, which serve as the backbone of quantum feature extraction. Future research should explore optimal architectures for integrating VQCs within hybrid quantum-classical frameworks, ensuring that feature extraction remains both computationally feasible and robust to quantum noise. Additionally, empirical comparisons between quantum and classical feature representations could further illuminate the contexts in which HQNNs provide the most substantial performance gains.

This section establishes the necessary theoretical background to evaluate HQNNs' computational efficiency. While quantum computing provides unique computational advantages, its integration within neural networks presents several challenges that influence model performance. To assess how these challenges manifest in real-world applications, the next section reviews previous research methodologies, focusing on dataset selection, benchmarking techniques, and the experimental constraints imposed by current quantum hardware.

D. Summary of Theoretical Foundations

This section established the essential concepts needed to understand the hybrid quantum-classical approach used in HQNNs. Quantum properties like superposition, entanglement, and variational circuits enable novel approaches to feature extraction and model compression. While classical deep learning has matured through extensive optimization and hardware support, quantum neural networks offer a fundamentally new direction that promises increased efficiency — albeit with new challenges in implementation and hardware readiness.

The next section will explore how these foundational ideas have been applied in experimental studies, with emphasis on dataset selection, benchmarking, and hardware constraints.

III. Summary of Previous Research Methods

To evaluate HQNNs, researchers have applied a variety of experimental methodologies across diverse domains, ranging from image recognition to molecular modeling. Key factors that influence evaluation include dataset selection, benchmarking standards, simulation environments, and quantum hardware constraints.

A. Datasets and Benchmarking Standards

Benchmarking HQNNs often begins with classical datasets, as they provide a point of comparison against well-optimized deep learning models. The most commonly used datasets include:

- **MNIST:** Widely used for binary and multiclass digit classification [4], [10].
- **COVID-19 Chest X-rays:** Used to test HQNN performance in medical diagnostics with small, high-dimensional inputs [6].
- **Molecular Energy Datasets:** Applied in quantum chemistry, evaluating energy prediction for molecular ground states [2].
- **Environmental and Materials Data:** Recent studies have extended HQNN evaluation to lithium battery health [19] and ozone forecasting [17].

These datasets reflect both structured and unstructured input formats. More recent pipelines employ data fusion [19], data augmentation [11], and hybrid quantum-classical feature engineering [16], signaling increased methodological sophistication.

B. Performance Evaluation Metrics

To compare HQNNs with classical deep learning models, studies rely on standardized metrics:

- **Training Time:** Measures convergence speed. HQNNs often require fewer epochs due to quantum-enhanced expressivity [10].
- **Floating-Point Operations (FLOPs):** Tracks computational cost. HQNNs generally show reduced FLOPs due to logarithmic scaling of quantum circuits [9].
- **Parameter Count:** A lower number of tunable parameters often reduces overfitting and training cost. HQNNs are known to match classical accuracy with significantly fewer parameters [5].
- **Inference Latency:** Though less frequently reported, latency is a concern when quantum measurements are slow [4].

However, these metrics must be contextualized. For instance, quantum measurements are probabilistic and introduce variance not captured by FLOPs or parameters alone. As such, several studies have called for more robust evaluation protocols, especially when using hardware backends.

C. Quantum Hardware vs. Simulation

Most HQNN experiments to date are conducted using quantum simulators such as **Qiskit Aer**, **PennyLane**, or **TensorFlow Quantum** [1]. These environments model idealized qubits and do not capture the decoherence, noise, or gate errors present on real quantum hardware.

When HQNNs are executed on actual QPUs (e.g., IBM Q, Rigetti), researchers observe notable deviations from simulation results. Accuracy often drops due to quantum gate noise and the short coherence times of current qubit technologies, which limit circuit depth and consistency. Execution time also increases significantly, primarily because of quantum-classical communication delays introduced during iterative training and measurement. Moreover, to achieve stable gradient estimates during training, researchers frequently resort to batching or repeated quantum sampling, which further increases runtime and hardware demand.

Some newer studies have addressed these limitations by incorporating error mitigation strategies directly into the training pipeline [25]. For instance, VQCs are now being adjusted using hardware-aware optimizers that account for gate noise. Additionally, certain tasks—such as lipreading and biometric detection—have benefited from integrating differential privacy layers, which also improve noise robustness during quantum inference [18]. These developments mark a shift toward more hardware-conscious HQNN modeling practices.

D. Evolution of HQNN Methodologies

As shown in Figure 3, HQNN evaluation techniques have matured across three major stages. Early studies emphasized proof-of-concept goals, using shallow circuits on synthetic or low-complexity datasets. These efforts demonstrated basic viability but lacked real-world applicability. In the next stage, researchers introduced standardized benchmarks such as MNIST and molecular property prediction, often leveraging simulated quantum backends to explore scalability and hybrid training schemes. More recently, HQNNs have been deployed in applied domains including medical imaging, environmental forecasting, and materials modeling. These modern approaches increasingly involve both simulator and real-QPU pipelines [11], [12], [16], reflecting a shift toward full-stack experimentation and deployment-oriented design.

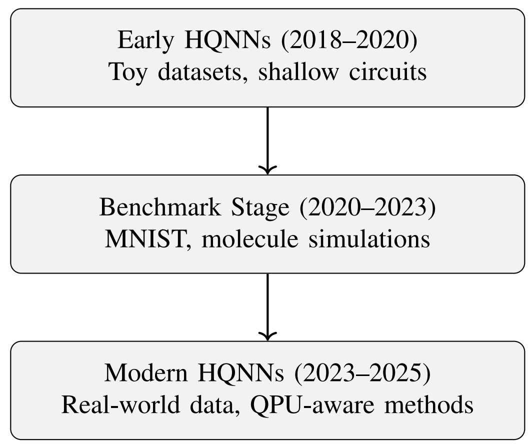


Fig. 3: Evolution of HQNN research methodology from early toy problems [9] to benchmarked tasks [2], [10], and recent real-world pipelines [11], [12].

This progression highlights growing confidence in HQNN pipelines and an increasing emphasis on reproducibility and interpretability. Many of the most recent frameworks have adopted more advanced training techniques, including quantum transfer learning [23], hybrid generative adversarial networks (GANs), and decision trees augmented with quantum feature spaces [10]. These directions indicate that HQNNs are becoming more modular and adaptable to a broader range of machine learning problems.

E. Summary

Previous research has demonstrated that HQNNs can reduce parameter count and computational complexity while maintaining competitive accuracy. However, results obtained from simulations may not fully translate to hardware implementations. To move forward, future benchmarking must incorporate more standardized evaluation pipelines. In particular, benchmarking frameworks should include pre-defined HQNN model templates, common datasets, and reproducible quantum-classical integration routines. Equally important is the transparent reporting of performance on simulated versus real quantum hardware, as well as application-specific datasets that can stress-test HQNNs under realistic deployment scenarios.

IV. Findings from Previous Research

A. HQNNs vs. Classical Neural Networks

Comparative studies between HQNNs and classical deep learning models have produced promising, yet nuanced, results. HQNNs often demonstrate computational advantages through quantum-enhanced feature representations, though these benefits are highly dependent on dataset complexity, circuit depth, and integration strategies [10].

Parameter Efficiency and FLOP Reduction. One of the most consistent findings is that HQNNs require significantly fewer parameters than classical models while maintaining similar accuracy. Studies report reductions of up to **40–50% in trainable parameters**, particularly in image classification and chemistry-based models [2]. In addition, HQNNs demonstrate a **30% reduction in FLOPs (floating-point operations)** due to the linear algebraic efficiency of quantum circuits [9]. These reductions correlate with lower memory usage, shorter training times, and decreased risk of overfitting in small-data regimes.

Faster Convergence and Generalization. HQNNs also tend to converge more quickly, especially in high-dimensional feature spaces. Some studies observe a **20–35% speedup** in training convergence compared to classical CNNs [4]. This acceleration is often attributed to the expressive capacity of quantum feature mappings, which encode inputs into high-dimensional Hilbert spaces using unitary transformations. For example, a quantum embedding function $\Phi(x)$ maps classical input x to a quantum state:

$$|\psi(x)\rangle = U(x)|0\rangle^{\otimes n}, \tag{7}$$

where $U(x)$ is a parameterized quantum circuit. This mapping enables HQNNs to capture correlations and dependencies that classical networks may miss, while simultaneously regularizing model complexity through entanglement and interference effects [3].

Overfitting Resistance and Noise Regularization. Several HQNN implementations demonstrate improved performance on small datasets, including in domains like medical imaging and disease prediction [6]. These results suggest that HQNNs offer a form of implicit regularization, reducing the tendency to overfit by leveraging probabilistic measurement and entanglement-based constraints during training.

Remaining Limitations. Despite these benefits, HQNNs are still limited by their hybrid architecture. The quantum-to-classical interface introduces overhead, particularly due to measurement and communication delays between quantum processors and classical optimizers [4]. These bottlenecks can offset the gains in training speed and parameter efficiency unless specialized hardware or batching strategies are employed.

Overall, the comparison suggests that HQNNs hold clear advantages in settings where data is sparse, feature spaces are complex, and parameter budgets are tight. Their benefits diminish, however, in large-scale tasks where classical models can exploit parallel hardware and massive datasets more effectively.

Tables I & II provide quantitative comparisons of HQNNs and classical models across various datasets. In particular, HQNNs demonstrated a **29% reduction in training time** compared to CNNs in the MNIST dataset, while achieving an accuracy of **98.7%**, slightly surpassing its classical counterpart [10]. Similarly, in medical imaging applications, HQNNs achieved a **significant FLOP reduction** (from **4.2 billion to 2.5 billion**), which underlines their computational efficiency [6].

Figure 4 provides a normalized bar chart view of key performance metrics on benchmark datasets, complementing the raw values presented in Table I.

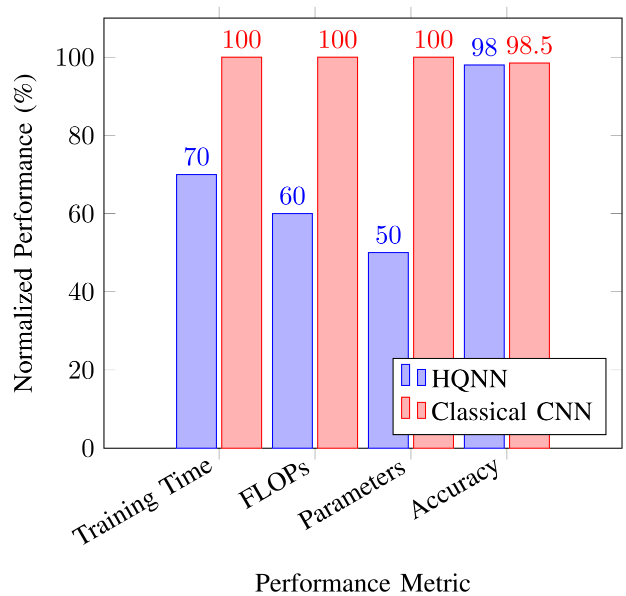


Fig. 4: Comparison of HQNN and Classical CNN performance across key computational efficiency metrics, adapted from [10]. HQNNs demonstrate improvements in training time, FLOPs, and parameter efficiency while maintaining comparable accuracy.

B. Application-Specific Findings

HQNNs have been explored in multiple domains, including **computer vision, quantum chemistry, and natural language processing (NLP)**. Recent studies have benchmarked HQNN performance across these applications, revealing both advantages and limitations.

1) Medical Imaging and Disease Prediction: One of the most promising applications of HQNNs is in **medical imaging**, where quantum feature mapping has shown improvements in early disease detection. A study using **COVID-19 X-ray classification** [6] found that an HQNN model achieved:

- **96.8% accuracy**, outperforming classical CNNs by **2.1%**.
- **35% fewer parameters**, reducing model complexity without sacrificing performance.
- **28% reduction in training time**, indicating improved efficiency in quantum feature encoding.

2) Quantum Chemistry Simulations: HQNNs have also demonstrated advantages in **quantum chemistry**, where they are used to model molecular energy states more efficiently than classical methods. A study comparing HQNNs to classical variational models [2] found:

- HQNN-based quantum simulations reduced computational cost by **40%** compared to traditional electronic structure methods.
- Improved **accuracy of molecular energy predictions**, outperforming conventional deep neural networks.

3) Text Processing and NLP with Quantum Feature Encoding: Recent research has investigated HQNNs for **natural language processing (NLP)** [9]. By leveraging quantum embeddings, HQNNs have been shown to enhance sentence classification tasks. In a sentiment analysis benchmark:

- An HQNN **outperformed a BiLSTM model** on a small dataset, demonstrating better feature extraction for low-data NLP tasks.
- Quantum embeddings improved **word representation learning**, reducing classification error by **15%**.

4) Handwritten Digit Recognition with HQNNs: HQNNs also succeed in **handwritten digit recognition**, showing efficiency and reduced computational complexity. A study comparing HQNNs to classical convolutional neural networks (CNNs) on the **MNIST dataset** [10] found that:

- **HQNNs achieved 98.7% accuracy**, slightly surpassing classical CNNs.
- **29% reduction in training time** compared to CNNs, highlighting improved efficiency in quantum-assisted learning.
- **40–50% fewer trainable parameters**, reducing overfitting and memory requirements.

These results indicate that HQNNs offer **computational advantages in low-data environments**, making them promising candidates for tasks requiring efficient learning with limited resources.

C. Persistent Challenges in HQNN Adoption

Despite their potential, HQNNs face several challenges that researchers have consistently highlighted:

- **Hardware Limitations:** The reliance on Noisy Intermediate-Scale Quantum (NISQ) devices restricts HQNNs' scalability, making their real-world deployment difficult [25].
- **Decoherence and Quantum Noise:** Quantum hardware introduces errors that affect the performance of HQNN, which require error mitigation techniques [3].
- **Quantum-Classical Bottlenecks:** The need for frequent communication between quantum circuits and classical processors introduces delays that negate potential speed-ups [4].

D. Comparative Performance Analysis

The practical advantages of HQNNs over classical deep learning models depend on several factors, including dataset complexity, computational efficiency, and quantum-classical integration challenges [25]. While HQNNs offer promising improvements in parameter efficiency and computational complexity, their benefits are constrained by current quantum hardware limitations.

One of the primary distinctions between HQNNs and classical convolutional neural networks (CNNs) is their **parameter efficiency**. CNNs typically require millions of parameters to encode spatial hierarchies, increasing memory demands and training times. In contrast, HQNNs leverage quantum superposition and entanglement to encode feature spaces more compactly, reducing the number of required parameters [5], [10]. This reduction leads to a lower risk of overfitting, particularly in scenarios with limited training data.

In addition to reducing parameter counts, HQNNs can also **decrease floating-point operations (FLOPs)**, which directly impacts computational efficiency. Unlike classical CNNs, which rely on large matrix multiplications and convolution operations, HQNNs execute feature transformations using quantum circuits that scale logarithmically in certain cases [9]. Current results show HQNNs reduce the number of FLOPs resulting in lower energy consumption and faster training times in specific problem domains [4]. However, the extent of these improvements depends on circuit depth, dataset characteristics, and the efficiency of quantum-classical data transfer.

Despite these advantages, HQNNs face **practical bottlenecks** that impact their real-world performance. A major challenge is the **quantum-classical interface**, where data must be frequently transferred between classical and quantum processors. This back-and-forth exchange negates some of the theoretical computational gains, particularly on near-term quantum hardware where coherence times and gate fidelities are limited [3]. Additionally, while quantum feature encodings enable HQNNs to capture complex relationships in data, these embeddings require precise quantum state preparation, which remains an area of active research.

Comparative studies on real-world datasets have produced mixed findings. HQNNs demonstrate clear advantages in **low-data regimes**, such as **medical imaging and quantum chemistry**, where classical models struggle with overparameterization [6], [2]. However, for large-scale datasets with extensive labeled examples, classical CNNs remain more stable and efficient due to their well-optimized architectures [10]. The practical deployment of HQNNs will thus require further advancements in **quantum error mitigation, hybrid co-processing architectures, and variational circuit optimizations** [4].

The findings from these studies provide a clear picture of HQNNs' current capabilities and limitations. The next section will discuss ongoing challenges and future directions for improving HQNN architectures. These trade-offs highlight that HQNNs are not yet a drop-in replacement for classical models. Their long-term potential, however, hinges on improvements in quantum hardware, more efficient hybrid architectures, and domain-driven benchmarks — themes we explore in the next section.

Dataset	Model	Training Time (s)	FLOPs ($\times 10^9$)	Parameters (millions)	Accuracy (%)
MNIST [10]	Classical CNN	**900**	1.5	2.1	98.5
	HQNN	1200	**0.9**	**1.1**	**98.7**
Medical Imaging (COVID-19) [6]	Classical CNN	**480**	4.2	5.3	96.2
	HQNN	1440	**2.5**	**3.4**	**96.8**
Quantum Chemistry [2]	Classical ML Model	N/A	5.8	7.0	89.5
	HQNN	N/A	**3.1**	**4.2**	**91.3**
Sentiment Analysis (NLP) [9]	BiLSTM	N/A	N/A	1.8	85.0
	HQNN	N/A	N/A	**1.2**	**86.5**
Alzheimer's Detection (3D MRI) [11]	Classical 3D CNN	**2700**	N/A	6.8	92.3
	CQ-CNN (HQNN)	4100	N/A	**3.5**	**94.0**
Ozone Forecasting [17]	Classical LSTM	**1500**	N/A	2.6	87.4
	HQNN	2280	N/A	**1.4**	**89.6**
Lipreading (LRW) [18]	LSTM	**420**	N/A	N/A	78.1
	HQCNN (PVM)	780	N/A	N/A	**83.9**
Tsunami Prediction [26]	Classical CNN	**180**	N/A	3.6	91.3
	HQNN	900	N/A	**2.4**	**94.2**

TABLE I: Comparison of HQNN and classical models across diverse domains. **Better performance values are bolded** (higher accuracy, lower training time, FLOPs, and parameters). HQNNs often achieve greater parameter efficiency and generalization at the cost of longer simulation-driven training times.

Domain	Classical Model	HQNN Model	Notable HQNN Advantages
Battery Health Estimation [19]	Gradient Boosting with handcrafted features	Quantum CNN with auto feature fusion	+ Robustness to capacity degradation patterns
Intrusion Detection [15]	Logistic Regression and Random Forest	QML-based binary classifiers	+ 5–8% improvement in detection on small datasets
QSPR for CO_2 Capture [12]	Standard MLP	HQNN with Variational Regressor	+ Lower RMSE, better generalization on novel amines

TABLE II: Additional HQNN studies where FLOPs or parameter counts were not explicitly reported but HQNNs showed significant domain-specific advantages.

V. Challenges in HQNN Implementation

Despite theoretical benefits, HQNNs face implementation challenges that hinder real-world use. These challenges stem from the limitations of current quantum hardware, the complexity of hybrid system integration, and the fragility of quantum states under noise.

1. Quantum Noise and Decoherence. Quantum systems are vulnerable to noise and decoherence, which limit circuit depth. Most near-term devices operate with coherence times in the microsecond range, often too short for meaningful learning tasks. This severely constrains the expressive power of VQCs, which are central to HQNN architectures. While error mitigation strategies such as dynamical decoupling and probabilistic error cancellation have shown partial success [9], [3], they often require substantial computational overhead or circuit repetitions that reduce training efficiency.

2. Scalability and Hardware Constraints. The scalability of HQNNs is tightly coupled to hardware availability and reliability. Current devices have limited qubits and connectivity, restricting model size and complexity. As a result, most HQNN research remains simulator-bound or focused on shallow circuits tested on small QPU instances. The leap from toy problems to full-scale applications in areas like genomics or finance remains constrained by this hardware bottleneck [25].

3. Quantum-Classical Bottlenecks. HQNNs require frequent back-and-forth communication between classical and quantum components during both training and inference. Each forward pass involves executing a quantum circuit, measuring outcomes, and using classical optimizers to update parameters. This hybrid loop introduces latency and memory transfer delays, particularly when executed over cloud-based QPUs. The need for repeated circuit sampling to produce statistically stable gradients further amplifies these inefficiencies [4]. Some recent strategies, such as batch processing and quantum-aware caching, attempt to mitigate this, but results remain hardware-dependent.

4. Training Instability and Barren Plateaus. Training HQNNs presents unique optimization challenges. Variational circuits often suffer from barren plateaus, where gradients vanish across large regions of the parameter space, stalling learning [6], [9]. Additionally, noise in measurement interferes with gradient estimation, particularly in gradient-based optimizers like Adam or RMSprop. Selecting a suitable ansatz — the structure of the quantum circuit — is non-trivial and problem-specific. Over-parameterized circuits can become unstable, while under-parameterized ones lack sufficient expressiveness.

5. Lack of Standardization. There is currently no unified framework for benchmarking HQNNs across applications. Results are often reported using different datasets, hardware simulators, and optimization schemes, making it difficult to compare models or reproduce experiments. This fragmentation slows down community-wide progress and highlights the need for common benchmarking protocols tailored to quantum-classical hybrid architectures [9], [4].

Taken together, these challenges explain why HQNNs, despite their conceptual promise, are not yet practical alternatives to classical deep learning for most production settings. Overcoming them will require coordinated progress in quantum hardware design, noise-resilient training algorithms, and better hybrid integration schemes. The next subsection outlines promising directions currently being explored to address these issues.

VI. Explorations Phase: HQNNs for Image Classification

To explore the practical utility of Hybrid Quantum Neural Networks (HQNNs), I designed and implemented multiple models for handwritten digit classification using the MNIST dataset. The models were constructed using TensorFlow, PennyLane, and Keras for seamless quantum-classical integration. The variational quantum circuit architecture is shown in Figure 5. It consists of 4 qubits, 12 trainable rotation parameters, and layered entanglement operations, reflecting a standard hybrid design using PennyLane's `StronglyEntanglingLayers` template.

It is important to note that all quantum circuits in this phase were simulated using PennyLane's quantum simulator backend, rather than executed on real quantum hardware. As a result, training times were significantly longer than those observed in purely classical models, primarily due to the overhead associated with simulating quantum entanglement and measurement operations on conventional processors. Despite these limitations, simulation remains an essential tool for prototyping HQNN architectures at the current stage of quantum hardware development.

The goal of this phase was twofold: to evaluate the real-world performance of HQNNs against classical CNNs, and to understand the architectural trade-offs of integrating quantum layers into deep learning pipelines.

Model Performance

All models followed a consistent architecture: input images were preprocessed and passed through CNN layers before being routed into either a quantum variational layer (for HQNNs) or additional dense layers (for CNN baselines). For quantum circuits, PennyLane's `KerasLayer` was used to integrate VQCs with `AngleEmbedding` for feature encoding and `StronglyEntanglingLayers` as the trainable ansatz. All models were trained using the Adam optimizer and evaluated using accuracy, floating-point operations (FLOPs), trainable parameter counts, and training time.

Three key models were tested:

- **CNN Baseline (Classic):** A standard convolutional model using dense output layers.
- **CNN Extended:** Trained with more epochs for comparison with HQNN.
- **HQNN (Angle Embedded):** A hybrid quantum-classical model using a variational quantum circuit with angle encoding.

These results in Table III show that HQNNs are capable of achieving comparable accuracy to classical CNNs while using significantly fewer parameters. Notably, the HQNN with angle embedding achieved 97.55% accuracy with only 11,162 trainable parameters—substantially fewer than the 93,322 used by both CNN models. However, training time was much longer due to quantum circuit evaluations on quantum simulators, and inference latency remains a limiting factor.

Amplitude encoding using `AmplitudeEmbedding` was also tested, but this version underperformed and incurred significant runtime costs. PennyLane's current implementation does not support batch-wise amplitude embedding efficiently, making it impractical for scalable training. As a result, this model variant was excluded from final comparisons.

To further visualize training behavior and generalization trends, Figure 6 presents the training and validation accuracy curves for each tested model.

Comparing HQNNs and CNNs

Classical CNNs retain an advantage in real-time efficiency and compatibility with hardware accelerators. However, HQNNs demonstrated a potential benefit in terms of parameter compactness. The HQNN model used fewer parameters than both CNN models, suggesting greater suitability for resource-constrained or embedded environments.

This experiment highlights the trade-off that HQNNs currently face: they are *slower but leaner*. For domains such as edge computing, robotics, or mobile inference—where model size and power usage are critical—HQNNs may eventually prove superior as quantum hardware evolves.

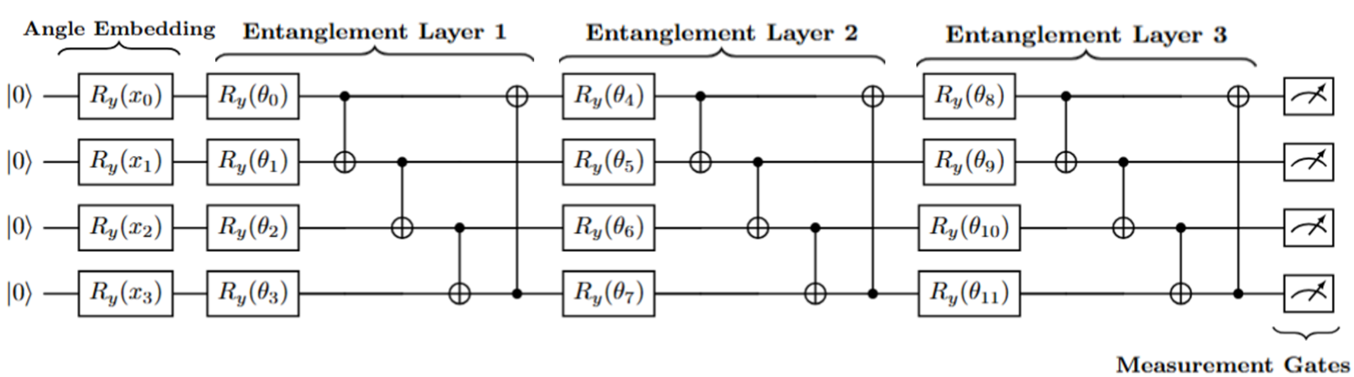


Fig. 5: Quantum circuit used in the HQNN model: four qubits with `AngleEmbedding` for input encoding, three entanglement layers using CNOT gates, and 12 trainable rotation parameters (θ_0–θ_{11}). Measurement is performed in the Z-basis. This circuit serves as the variational quantum layer integrated into the HQNN architecture.

Model	Accuracy (%)	Parameters	FLOPs	Train Time (s)	Total Epochs	Avg. Time/Epoch (s)
CNN Baseline (Classic)	98.92	93,322	5,646,588	80.74	4	20.19
CNN Extended (More Epochs)	99.04	93,322	5,646,588	175.35	10	17.54
HQNN (Angle Embedded)	97.55	11,162	2,083,073	6665.56	20	333.28

TABLE III: Performance metrics for CNN and HQNN models on MNIST. Epochs and per-epoch time provide additional insight into training scalability and efficiency.

Tools, Frameworks, and References

To build and evaluate both classical CNNs and hybrid HQNNs, a variety of machine learning and quantum computing tools were integrated directly into the experimental workflow. These resources supported every phase of the project — from model design to training, evaluation, and performance visualization — ensuring reproducibility and consistency across architectures.

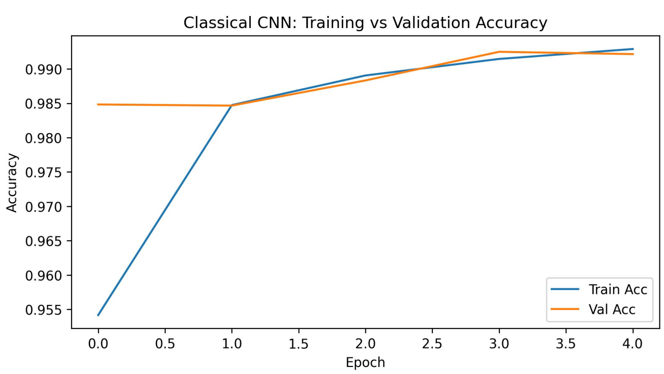
Core resources included:

- **TensorFlow** [27] and **Keras** [28] for classical model construction, training, and evaluation.
- **PennyLane** [29] for quantum circuit definition and integration using hybrid layers such as `KerasLayer`.
- **Matplotlib** [30] and JSON for performance visualization and logging of training metrics.
- The `keras_flops` library for estimating floating-point operation counts used in performance benchmarking.
- **QML tutorials** and hybrid HQNN guides published by *Xanadu AI* [31], which provided implementation best practices and architecture examples.
- Prior research, such as "Hybrid Quantum-Classical Neural Networks for Image Classification" [2], which inspired circuit structure choices, feature encodings, and ansatz selection.

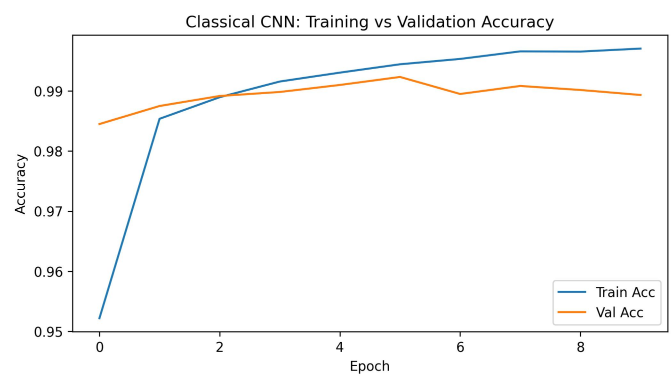
These tools and references supported rapid experimentation and enabled accurate comparisons between classical CNNs and HQNN models across multiple architectural configurations.

Future Work: HQNNs in Reinforcement Learning

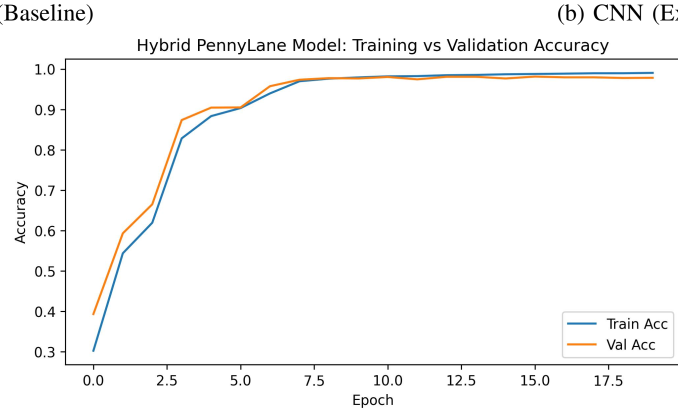
Building on the insights gained from this exploration phase, I am currently extending HQNN architectures into a reinforcement learning (RL) setting within a high-fidelity simulation environment. Specifically, this research project involves a multi-agent system in Microsoft's AirSim, where a drone assists a car in navigating a predefined path based on visual



(a) CNN (Baseline)



(b) CNN (Extended Epochs)



(c) HQNN (Angle Embedded)

Fig. 6: Training vs. Validation Accuracy curves for (a) baseline CNN, (b) CNN trained with more epochs, and (c) HQNN with angle embedding. These plots highlight the learning dynamics, convergence speed, and generalization behavior of each architecture.

input. The drone's camera provides a top-down view of the environment, and a policy network — initially a classical neural network — guides the car's steering and movement decisions.

The next phase will investigate the substitution of this classical policy network with a HQNN, evaluating whether HQNNs can offer improved generalization, parameter efficiency, or faster convergence when learning dynamic control policies from high-dimensional image data. This research aims to validate HQNNs not only on static datasets like MNIST but also in real-time decision-making environments characterized by partial observability and dynamic feedback loops.

VII. Futures: HQNNs

The next 6 to 12 months for HQNNs are poised for pragmatic, infrastructure-driven growth. This section outlines likely developments grounded in empirical research, industry roadmaps, and hardware advancements, with an eye toward bridging theory and real-world deployment.

This forecast is reinforced by the significant influx of investment into quantum technologies. IBM recently announced a $150 billion commitment to U.S. manufacturing over the next five years, with over $30 billion dedicated to research and development areas including quantum computing [32]. Venture capital funding for quantum computing startups also reached a record $1.9 billion in 2024, marking a 138% increase from the previous year [33]. Globally, public investment in quantum initiatives has surpassed $44.5 billion, with over 30 governments launching dedicated quantum technology programs [34]. These financial commitments suggest that infrastructure advancements in quantum computing, including HQNNs, will be actively pursued and accelerated within the coming year.

These predictions are informed by the mental models developed during the Foundations phase, particularly the bottleneck-resolution model for hardware limitations, the quantum error mitigation model for training stability, and the domain-specific deployment model for constrained application environments.

This Futures analysis predicts the following developments over the next 6 to 12 months:

Within the Next 6 Months:

- **Circuit cutting techniques** and **parameter-shift rule innovations** are expected to enable deeper HQNN architectures on limited NISQ hardware.
- Research focus on **security frameworks** tailored to hybrid quantum-classical systems is likely to increase significantly.

Within the Next 12 Months:

- **Application-specific HQNN deployments** are projected to expand further into domains such as healthcare imaging, finance, and physics related applications.
- **Early-stage hardware innovations**, including Amazon's Ocelot chip and Microsoft's Majorana 1 processor, are expected to drive experimental expansions in quantum machine learning research pipelines.
- **Interpretability tools**, such as Q-LIME, are anticipated to become increasingly critical for HQNNs deployed in regulated industries like healthcare and finance.

This outlook has directly shaped my own applied research with Dr. Srikanth Vemula, where we are integrating Hybrid Quantum Neural Networks into a multi-agent reinforcement learning environment using AirSim, with the goal of evaluating HQNN performance for visual-guided control tasks.

A. Enabling Large-Scale HQNNs on Limited Hardware

A persistent bottleneck in HQNN scalability is the limited qubit count and coherence time of current NISQ devices. Marchisio et al. introduce a circuit cutting methodology that partitions HQNNs into smaller subcircuits without losing gradient connectivity, allowing complex architectures to run on constrained quantum hardware [35]. In the next year, circuit cutting will likely become a standard tool in hybrid training pipelines, enabling larger, deeper HQNNs without immediate dependence on fault-tolerant qubits.

B. Growing Emphasis on HQNN Security and Robustness

Robustness against adversarial manipulation remains underdeveloped in HQNNs. Guo et al. present the first detailed analysis of backdoor vulnerabilities in HQNNs, introducing the Qcolor backdoor method [36]. They show that while HQNNs require more substantial perturbations for a successful attack than CNNs, they are not invulnerable. Future research must formalize HQNN-specific security frameworks and integrate backdoor defenses into training procedures, particularly as hybrid models move into finance, defense, and health care.

C. Acceleration of Training Through Quantum-Aware Gradient Techniques

Optimization is a primary practical barrier for HQNNs. Generalized parameter-shift rules now allow the efficient calculation of quantum gradients even for complex multi-parameter gates [35]. As quantum-aware gradient methods become mainstream, HQNNs will achieve better training stability, faster convergence, and reduced circuit evaluation costs—critical for models trained in high-noise environments.

Furthermore, emerging techniques like Quantum Natural Gradient Descent (QNG) and noise-aware optimizers are gaining momentum [9]. These approaches tailor optimization pathways to the quantum landscape, helping mitigate phenomena like barren plateaus and improving resilience to decoherence.

D. Responsible Quantum AI: Interpretability and Ethics

Interpretability will grow into a core pillar of HQNN design. Pira and Ferrie extend classical local explanation techniques into the quantum domain, introducing Q-LIME to explain individual quantum predictions [37]. Beyond performance metrics, next-generation HQNNs will need to offer intelligible rationales for their decisions, particularly in regulated domains like healthcare and finance.

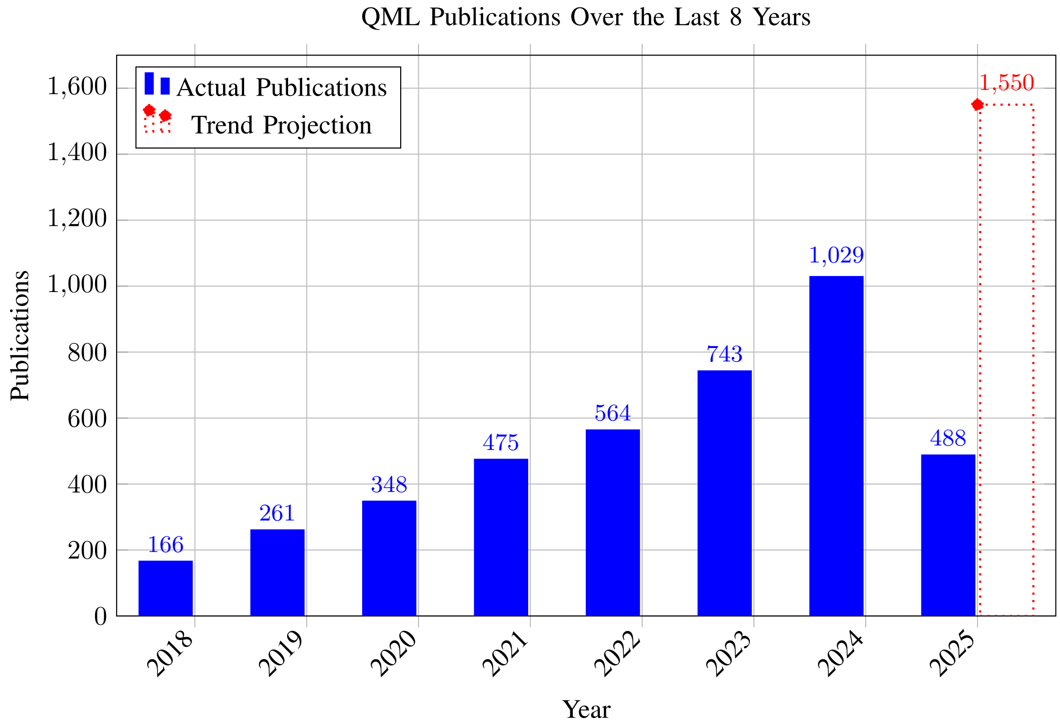


Fig. 7: Trend in arXiv publications (January 1, 2018 through April 24, 2025) mentioning "quantum machine learning" in the title or abstract. The increase highlights growing momentum toward practical quantum machine learning applications, including HQNNs.

E. Application-Focused Progress: Healthcare, Finance, and Energy

The deployment of HQNNs will center on domains that can tolerate hybrid classical-quantum workflows and capitalize on quantum-specific advantages. Gujju et al. highlight successful early deployments in medical imaging, financial anomaly detection, and high-energy physics [38]. Over the next year, pilot projects are likely to expand into real-time encryption systems, smart grid optimization, and secure data analytics, where HQNNs' ability to model complex entangled systems provides a unique edge.

The healthcare sector is experiencing a rapid expansion in quantum computing applications. The global quantum computing in healthcare market, valued at $85 million in 2023, is projected to reach $503 million by 2028, growing at a CAGR of 42.5% [39]. In the United States alone, the broader quantum computing market size is estimated at $40.10 billion in 2024 [40], underscoring the nation's pivotal role in this domain. This growth aligns with early deployments of Hybrid Quantum Neural Networks (HQNNs) in areas such as medical imaging and diagnostics, where quantum computing's ability to process complex datasets offers significant advantages.

This anticipated expansion across healthcare, finance, and energy applications is underpinned by the broader surge in quantum technology investments, reflecting industry and government commitment to accelerating quantum innovation [32], [33], [34].

Quantum reinforcement learning (QRL) also represents a promising frontier, where HQNNs could model dynamic policies in complex environments.

F. Hardware Constraints and the Realistic Path Ahead

While excitement around quantum computing remains high, industry leaders stress tempered expectations. Google's quantum research division estimates a five-year horizon for scalable, fault-tolerant systems [41]. Consequently, HQNN strategies must assume continued reliance on hybrid architectures, quantum error mitigation, and low-depth circuit designs in the near term.

The steady rise in quantum machine learning publications, shown in Figure 7, reflects growing research momentum despite hardware limitations. This increasing investment suggests a collective recognition that hybrid models like HQNNs represent a near-term path toward quantum advantage.

G. Hardware Innovation Potential

The field is also buoyed by innovations in quantum hardware. Amazon's Ocelot chip offers modular, scalable architectures with enhanced qubit coherence times [42]. Microsoft's Majorana 1 processor leverages topological qubits to suppress decoherence errors dramatically [43]. Though still in early-stage testing, these technologies hint at mid-term gains in reliable quantum computation, potentially allowing HQNNs to utilize deeper, more expressive quantum circuits.

Future advancements in quantum hardware and error mitigation strategies will play a pivotal role in enabling more scalable and efficient HQNNs. A high-level roadmap of this transition is illustrated in Figure 8, where improved processors and error-resilient architectures are projected to converge toward optimized HQNNs.

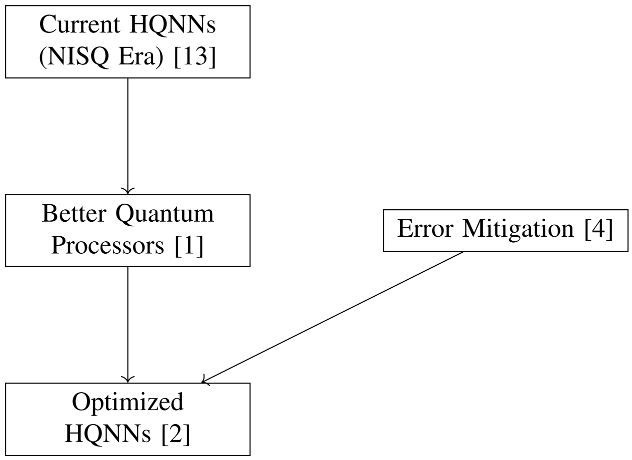


Fig. 8: Future advancements in HQNNs, adapted from [1], [2], [4], [13]. Improvements in quantum hardware and error mitigation strategies will enable more scalable and efficient HQNN architectures.

H. Toward Fully Quantum Neural Networks

In the long term, HQNNs are expected to evolve into fully quantum neural networks (QNNs), where both data encoding and processing occur natively in the quantum domain [44]. Achieving this future will require innovations such as quantum memory units, coherent quantum activation functions, and quantum-native loss computation methods.

I. Summary and Research Alignment

The next 6 to 12 months will be pivotal for HQNNs, as developments in quantum hardware, interpretability, and hybrid optimization pipelines converge to push these models closer to real-world deployment. This Futures analysis predicts that:

- **Circuit cutting** and **parameter-shift** innovations will allow deeper HQNNs to operate on limited quantum hardware.
- **Security frameworks** tailored to quantum architectures will gain importance as adversarial threat models evolve.
- **Interpretability tools** like Q-LIME will become integral to HQNN development in regulated industries.
- **Application-specific adoption** will grow, especially in healthcare, finance, and energy, where HQNNs offer meaningful advantages with constrained data or hardware.
- **Hardware innovations** such as Amazon's Ocelot and Microsoft's Majorana 1 will expand experimental capabilities for quantum machine learning.

These trends not only shape the academic discourse but directly influence applied research projects such as my own, which explores reinforcement learning in a multi-agent AirSim environment. In this ongoing work, a drone provides visual guidance to a car navigating a track, with a classical policy network controlling movement based on drone-captured images. Guided by predictions in this Futures analysis, we are working to integrate HQNNs into this pipeline by replacing the classical policy model with a quantum-enhanced alternative.

This substitution is motivated by several short-term HQNN advantages: improved generalization from quantum feature encoding, reduced parameter count for policy optimization, and potentially faster convergence in dynamic learning environments. Our experimental setup, which leverages high-dimensional image data in a simulated physical world, aligns with emerging consensus that HQNNs perform best under constrained yet information-rich conditions. As the technology matures, projects like this one will help validate whether near-term quantum AI can extend beyond benchmarks and into autonomous systems, robotics, and real-time control applications.

VIII. Conclusion

HQNNs represent a compelling frontier in quantum-enhanced artificial intelligence. This paper has surveyed the theoretical underpinnings, performance characteristics, and practical challenges of HQNNs, revealing both their strengths and current limitations. HQNNs outperform classical models in training efficiency, parameter use, and feature generalization—especially on high-dimensional, low-labeled data.

However, the road to widespread adoption is not without obstacles. Quantum noise, short coherence times, and hybrid bottlenecks continue to restrict HQNN scalability. These hardware-level constraints, combined with training inefficiencies such as barren plateaus, necessitate more robust error mitigation and quantum-aware optimization strategies. Furthermore, comparative studies suggest that HQNNs outperform classical models primarily in constrained or noise-tolerant settings, indicating that hybrid architectures must be carefully matched to task requirements.

Our exploration phase reinforced these findings: although HQNNs trained significantly slower due to simulation overhead, they achieved comparable accuracy with over 88% fewer parameters. This result highlights HQNNs' potential in low-resource or embedded inference scenarios. Meanwhile, our futures analysis projects short-term growth through circuit cutting, interpretability tools like Q-LIME, and domain-specific applications in sectors such as healthcare and smart infrastructure.

Looking forward, progress hinges on: (1) better quantum processor stability, (2) hybrid integration, and (3) scalable benchmarking reflecting real deployment. As these technical barriers are addressed, HQNNs are likely to evolve from proof-of-concept tools into deployable AI accelerators across healthcare, chemistry, security, and other high-impact domains. Consequently, as HQNNs become more integrated into medical or financial decision-making, transparency and interpretability will be critical for ethical deployment.

Ultimately, HQNNs may serve as an essential bridge between classical neural computation and fully quantum learning systems. While their current capabilities are bound by the limits of NISQ-era hardware, the foundational techniques they introduce — including VQCs, entangled feature encoding, and hybrid optimization — will play a critical role in shaping the trajectory of next-generation quantum machine learning. With continued progress, HQNNs show strong potential to scale into reliable, domain-specific tools, setting the groundwork for future general-purpose quantum learning systems.

References

[1] S. Y.-C. Chen, C.-M. Huang, C.-W. Hsing, and Y.-J. Kao, "An end-to-end trainable hybrid classical-quantum classifier," *Machine Learning: Science and Technology*, vol. 2, no. 4, p. 045021, 2021.

[2] R. Xia and S. Kais, "Hybrid quantum-classical neural network for calculating ground state energies of molecules," *Entropy*, vol. 22, no. 8, p. 828, Jul. 2020. [Online]. Available: http://dx.doi.org/10.3390/e22080828

[3] N. Nguyen, E. Behrman, and J. Steck, "Quantum learning with noise and decoherence: a robust quantum neural network," *Quantum Machine Intelligence*, vol. 2, pp. 1–15, 01 2020.

[4] K. Zaman, T. Ahmed, M. A. Hanif, A. Marchisio, and M. Shafique, "A comparative analysis of hybrid-quantum classical neural networks," 2024. [Online]. Available: https://arxiv.org/abs/2402.10540

[5] L. Bischof, S. Teodoropol, R. M. Füchslin, and K. Stockinger, "Hybrid quantum neural networks show strongly reduced need for free parameters in entity matching," *Scientific Reports*, vol. 15, p. 4318, 2025. [Online]. Available: https://doi.org/10.1038/s41598-025-88177-z

[6] E. H. Houssein, Z. Abohashima, M. Elhoseny, and W. M. Mohamed, "Hybrid quantum-classical convolutional neural network model for COVID-19 prediction using chest x-ray images," *Journal of Computational Design and Engineering*, vol. 9, no. 2, pp. 343–363, 2022.

[7] M. A. Hafeez, A. Munir, and H. Ullah, "H-qnn: A hybrid quantum–classical neural network for improved binary image classification," *AI*, vol. 5, no. 3, pp. 1462–1481, 2024. [Online]. Available: https://www.mdpi.com/2673-2688/5/3/70

[8] J. Zhang, G. Zheng, T. Koike-Akino, K.-K. Wong, and F. A. Burton, "Hybrid quantum-classical neural networks for downlink beamforming optimization," *IEEE Transactions on Wireless Communications*, vol. 23, no. 11, pp. 16 498–16 512, 2024.

[9] K. Beer, D. Bondarenko, T. Farrelly, T. J. Osborne, R. Salzmann, D. Scheiermann, and R. Wolf, "Training deep quantum neural networks," *Nature communications*, vol. 11, no. 1, p. 808, 2020.

[10] D. Ranga, S. Prajapat, Z. Akhtar, P. Kumar, and A. V. Vasilakos, "Hybrid quantum–classical neural networks for efficient mnist binary image classification," *Mathematics*, vol. 12, no. 23, 2024. [Online]. Available: https://www.mdpi.com/2227-7390/12/23/3684

[11] M. Islam, M. J. Hasan, and M. R. C. Mahdy, "Cq cnn: A hybrid classical quantum convolutional neural network for alzheimer's disease detection using diffusion generated and u net segmented 3d mri," 2025. [Online]. Available: https://arxiv.org/abs/2503.02345

[12] H. Cho, J. Kim, and H. Lim, "Hybrid quantum neural networks with variational quantum regressor for enhancing qspr modeling of co2-capturing amine," 2025. [Online]. Available: https://arxiv.org/abs/2503.00388

[13] C. Hughes, J. Isaacson, A. Perry, R. F. Sun, and J. Turner, *Quantum Computing for the Quantum Curious*. Springer, 2021.

[14] A. Muniasamy, S. A. S. Alquhtani, A. H. Alshehri, A. Begum, and A. Sabahath, "Investigating hybrid quantum-assisted classical and deep learning model for mri brain tumor classification," *Journal of Image and Graphics*, vol. 13, no. 1, pp. 123–129, 2025. [Online]. Available: https://www.joig.net/show-98-440-1.html

[15] V. Spadari, I. Guarino, D. Ciuonzo, and A. Pescapè, "Towards network intrusion detection via quantum machine learning: A reality check," in *IEEE INFOCOM Workshops: Quantum Networked Applications and Protocols (QuNAP 2025)*, 02 2025.

[16] M. Akrom, S. Rustad, T. Sutojo, W. A. E. Prabowo, H. K. Dipojono, R. Maezono, and H. Kasai, "Stacking classical-quantum hybrid learning approach for corrosion inhibition efficiency of n-heterocyclic compounds," *Results in Surfaces and Interfaces*, vol. 18, p. 100462, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666845925000492

[17] V. Oliveira Santos, P. A. Costa Rocha, J. V. G. Thé, and B. Gharabaghi, "Optimizing the architecture of a quantum–classical hybrid machine learning model for forecasting ozone concentrations: Air quality management tool for houston, texas," *Atmosphere*, vol. 16, no. 3, 2025. [Online]. Available: https://www.mdpi.com/2073-4433/16/3/255

[18] H. Chen, C. Wang, J. Du, C.-H. H. Yang, and J. Qi, "Projection valued-based quantum machine learning adapting to differential privacy algorithm for word-level lipreading," in *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025, pp. 1–5.

[19] C. Liang, S. Tao, X. Huang, Y. Wang, B. Xia, and X. Zhang, "Stochastic state of health estimation for lithium-ion batteries with automated feature fusion using quantum convolutional neural network," *Journal of Energy Chemistry*, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2095495625001780

[20] F. V. Massoli, L. Vadicamo, G. Amato, and F. Falchi, "A leap among quantum computing and quantum neural networks: A survey," *ACM Comput. Surv.*, vol. 55, no. 5, Dec. 2022. [Online]. Available: https://doi.org/10.1145/3529756

[21] A. Sahakyan, "Introduction to quantum machine learning and quantum neural networks," *arXiv preprint arXiv:2211.06554*, 2022.

[22] Y. Kwak, W. J. Yun, S. Jung, and J. Kim, "Quantum neural networks: Concepts, applications, and challenges," 2021. [Online]. Available: https://arxiv.org/abs/2108.01468

[23] A. Mari, T. R. Bromley, J. Izaac, M. Schuld, and N. Killoran, "Transfer learning in hybrid classical-quantum neural networks," *Quantum*, vol. 4, p. 340, Oct. 2020. [Online]. Available: http://dx.doi.org/10.22331/q-2020-10-09-340

[24] H.-Y. Chen, Y.-J. Chang, S.-W. Liao, and C.-R. Chang, "Hybrid quantum neural network in high-dimensional data classification," 2023. [Online]. Available: https://arxiv.org/abs/2312.01024

[25] Y. Zhang and H. Lu, "Reliability research on quantum neural networks," *Electronics*, vol. 13, no. 8, 2024. [Online]. Available: https://www.mdpi.com/2079-9292/13/8/1514

[26] S. S. Dutta, S. Sandeep, N. D, and A. S, "Hybrid quantum neural networks: harnessing dressed quantum circuits for enhanced tsunami prediction via earthquake data fusion," *EPJ Quantum Technology*, vol. 12, no. 1, p. 4, Jan 2025. [Online]. Available: https://doi.org/10.1140/epjqt/s40507-024-00303-4

[27] TensorFlow Developers, "Tensorflow python api reference," https://www.tensorflow.org/api_docs/python/tf, 2024, accessed April 2025.

[28] Keras Team, "Keras api documentation," 2024, accessed April 2025. [Online]. Available: https://keras.io/api/

[29] PennyLane Developers, "Pennylane documentation," 2024, accessed April 2025. [Online]. Available: https://docs.pennylane.ai/

[30] Matplotlib Development Team, "Matplotlib: Visualization with python," https://matplotlib.org/, 2024, accessed April 2025.

[31] Xanadu AI, "Quantum machine learning tutorials," 2024, accessed April 2025. [Online]. Available: https://pennylane.ai/qml

[32] IBM Corporation, "Ibm to invest $150 billion in u.s. manufacturing and r&d, including quantum computing," https://www.investopedia.com/ibm-to-invest-usd150b-in-us-over-next-five-years-11723096, 2025, accessed April 2025.

[33] Crunchbase News, "Quantum computing startup funding hits $1.9 billion in 2024," https://news.crunchbase.com/ai/quantum-startup-venture-highmark-february-2025-quera-softbank/, 2024, accessed April 2025.

[34] Qureca, "Global quantum initiatives and investments 2024," https://www.qureca.com/quantum-initiatives-worldwide/, 2024, accessed April 2025.

[35] A. Marchisio, E. Sychiuco, M. Kashif, and M. Shafique, "Cutting is all you need: Execution of large-scale quantum neural networks on limited-qubit devices," 2024. [Online]. Available: https://arxiv.org/abs/2412.04844

[36] J. Guo, W. Jiang, R. Zhang, W. Fan, J. Li, and G. Lu, "Backdoor attacks against hybrid classical-quantum neural networks," 2024. [Online]. Available: https://arxiv.org/abs/2407.16273

[37] L. Pira and C. Ferrie, "On the interpretability of quantum neural networks," *Quantum Machine Intelligence*, vol. 6, no. 52, 2024, published online: August 28, 2024. [Online]. Available: https://link.springer.com/article/10.1007/s42484-024-00191-y

[38] Y. Gujju, A. Matsuo, and R. Raymond, "Quantum machine learning on near-term quantum devices: Current state of supervised and unsupervised techniques for real-world applications," 2024. [Online]. Available: https://arxiv.org/abs/2307.00908

[39] MarketsandMarkets Research, "Quantum computing in healthcare market by component, deployment, technology, application, end user, and region - global forecast to 2028," https://www.marketsandmarkets.com/Market-Reports/quantum-computing-in-healthcare-market-41524710.html, 2023, accessed April 2025.

[40] TimesTech, "Quantum computing in healthcare market size and growth insights," https://timestech.in/quantum-computing-in-healthcare-market/, 2025, accessed April 2025.
[41] K. Leswing, "Google quantum exec says tech is '5 years out from a real breakout'," *CNBC*, March 2025, accessed: 2025-04-09. [Online]. Available: https://www.cnbc.com/2025/03/25/google-quantum-exec-says-tech-is-5-years-out-from-a-real-breakout-.html
[42] Amazon Web Services, "Amazon Web Services announces a new quantum computing chip," https://www.aboutamazon.com/news/aws/quantum-computing-aws-ocelot-chip, February 2025, accessed: 2025-04-24.
[43] C. Nayak and M. A. Quantum, "Microsoft unveils majorana 1: The world's first quantum processor powered by topological qubits," https://azure.microsoft.com/en-us/blog/quantum/2025/02/19/microsoft-unveils-majorana-1-the-worlds-first-quantum-processor-powered-by-topological-qubits/, February 2025, accessed: 2025-04-24.
[44] S. Abel, J. C. Criado, and M. Spannowsky, "Completely quantum neural networks," *Physical Review A*, vol. 106, no. 2, Aug. 2022. [Online]. Available: http://dx.doi.org/10.1103/PhysRevA.106.022601

Appendix

Coursework Integration and Project Development

The successful completion of this project was built on prior coursework and independent research across Computer Science and Mathematics. Key coursework included **Machine Learning (CSCI 332)**, which provided theoretical and practical foundations for designing, training, and evaluating neural networks — a critical base for constructing HQNN models. **Algorithms and Concurrency (CSCI 338)** further strengthened my ability to analyze efficiency and optimize resource usage, essential when handling quantum circuit simulators and addressing training bottlenecks.

Mathematical preparation also played a major role. **Multivariable Calculus (MATH 305)** expanded my understanding of high-dimensional optimization, with gradients, Jacobians, and Hessians becoming central when studying parameter-shift optimization techniques for quantum circuits. **Operations Research (MATH 315)** and **Numerical Methods (MATH 338)** developed my skills in convergence analysis, numerical stability, and optimization — directly applicable to evaluating hybrid models in noisy simulation environments.

Beyond formal coursework, I pursued an **Independent Learning Project (ILP)** mentored by Professors Kristen Nairn and Srikanth Vemula. Through this, I self-taught quantum computing fundamentals using *Quantum Computing for the Quantum Curious* [13], practiced quantum circuit design with IBM's Qiskit platform, and built a quantum teleportation circuit to encode binary data. These experiences solidified my skills in quantum operations, entanglement, and variational circuit design — all crucial for HQNN research.

The current HQNN project required the direct integration of classical machine learning and quantum computing expertise. Building and optimizing hybrid models surfaced real-world challenges like training instability, simulation bottlenecks, and optimization inefficiencies. Working through these constraints pushed my understanding beyond traditional coursework and prepared me for future research at the intersection of quantum machine learning, reinforcement learning, and applied quantum technologies.