Quantum Convolutional Neural Networks for Image Classification: Perspectives and Challenges

Fabio Napoli[1][a], Lelio Campanile[1][b], Giovanni De Gregorio[1,2][c] and Stefano Marrone[1][d]

[1]*Dipartimento di Matematica e Fisica, Università degli Studi della Campania "Luigi Vanvitelli",*
viale Lincoln 5, Caserta, Italy
[2]*Istituto Nazionale di Fisica Nucleare, Complesso Universitario di Monte S. Angelo, Via Cintia, Napoli, I-80126, Italy*
{*fabio.napoli, lelio.campanile, giovanni.degregorio, stefano.marrone*}*@unicampania.it*

Keywords: Quantum Convolutional Neural Networks, Labelled Faces in the Wild, Face Recognition.

Abstract: Quantum Computing is becoming a central point of discussion in both academic and industrial communities. Quantum Machine Learning is one of the most promising subfields of this technology, in particular for image classification. In this paper, the model of Quantum Convolutional Neural Networks and some related implementations are explored in their potential for a non-trivial task of image classification. The paper presents some experimentations and discusses the limitations and the strengths of these approaches when compared with classical Convolutional Neural Networks. Furthermore, an analysis of the impact of the noise level on the quality of the classification task has been performed. This paper reports a substantial equivalence of the perfomance of the model with respect the level of noise.

1 INTRODUCTION

Machine Learning (ML) has become a powerful tool for pattern recognition, enabling Artificial Intelligence (AI) diffusion. Transformer-based models exemplify its capabilities, enabling human-like text generation and natural language processing. Meanwhile, the volume of stored data grows exponentially, exceeding several hundred exabytes and increasing at 20% annually (Hilbert and López, 2011). This surge has spurred interest in novel ML approaches capable of managing and extracting insights from vast datasets, with Quantum Computing (QC) emerging as a promising frontier (Schuld and Petruccione, 2021).

By leveraging superposition and entanglement, QC offers new paradigms for optimization, image processing, and complex data analysis, promising significant speed-ups over classical methods. However, realizing Quantum Machine Learning (QML)'s full potential requires overcoming key challenges, including quantum noise, scalability, and integration with classical models. Noisy Intermediate-Scale Quantums (NISQs) devices introduce decoherence and gate errors, complicating reliable computation. Consequently, many algorithms are tested in simulated environments, where classical resources approximate quantum behaviour, enabling stability analysis and performance comparisons with other ML approaches.

This study investigates Quantum Convolutional Neural Networks (QCNNs) for image classification, evaluating their performance in both ideal and noisy simulated environments. The impact of quantum noise on model accuracy is analyzed to assess the feasibility of QCNNs in practical scenarios. Experimental results highlight both the potential and the current limitations of QCNNs, emphasizing the gap between quantum advantage and existing hardware constraints.

The paper is structured as follows: Section 2 reports a review of the relevant papers; Section 3 describes the foundation of the work and the followed methodology; Section 4 describes the experimentations carried on in the work and discusses the main results; Section 6 ends the paper and draws conclusions.

2 RELATED WORK

Recent studies explore various QML and Deep Learning (DL) architectures for image classifica-

[a] https://orcid.org/0009-0001-0396-7968
[b] https://orcid.org/0000-0003-4021-4137
[c] https://orcid.org/0000-0003-0253-915X
[d] https://orcid.org/0000-0003-1927-6173

509

tion (Kharsa et al., 2023), including Quantum Support Vector Machines (QSVM), Quantum K-Nearest Neighbors (Q-KNNs), QCNNs, Variational Quantum Circuits (VQCs), and Quantum Tensor Networks (QTNs). These works highlight the promise of QC while acknowledging the limitations imposed by NISQs devices and the need for improved quantum image encoding and larger datasets.

The study in (Lu et al., 2021) presents a QCNN model applied to Modified National Institute of Standards and Technology (MNIST), leveraging hierarchical quantum layers inspired by classical Convolutional Neural Networks (CNNs). Using amplitude encoding and approximate state preparation, the model optimizes qubit usage, demonstrating its effectiveness in digit classification. Similarly, (Chen et al., 2023) investigates how different image types affect QCNN performance, emphasizing quantum initial state preparation and exploring two local feature construction schemes, Scale Layer Unitary Circuits (SLUCs) and Box-Counting Based Fractal Features (BCBFFs). The study examines how QCNNs integrate local features into a global circuit, assessing their adaptability for classical image classification.

In (Easom-Mccaldin et al., 2024), a single-qubit-based deep Quantum Neural Network (QNN) is proposed to improve parameter efficiency and scalability in high-dimensional image classification tasks. The authors examine VQCs expressibility, entanglement capabilities, and noise resilience, identifying a saturation point where increased circuit depth no longer improves classification accuracy. While dataset details are not specified, the study likely includes benchmarks like MNIST.

The research in (Hassan et al., 2024) applies a hybrid QCNN-Residual Neural Network (ResNet) architecture to the MNIST Medical dataset, integrating a quantum component for preprocessing and feature extraction while leveraging a pre-trained ResNet for classification. With an accuracy of 99.7%, this hybrid model outperforms standalone QCNNs and classical ResNet, demonstrating the benefits of quantum-classical integration.

The paper (Gong et al., 2024) focuses on binary classification tasks without specifying a particular dataset. The study explores parameter adjustments and uniform normalization techniques to mitigate data representation distortions, highlighting preprocessing strategies for optimized quantum computation. The proposed QCNN architecture is based on VQCs and follows a standard QCNN structure, including data encoding, convolutional layers, and pooling layers, but omitting a fully connected layer to align with the binary classification objective. The

paper investigates different convolutional kernel circuits, including Tree Tensor Networks (TTNs) (Wall and D'Aguanno, 2021) and SU(4) circuit (Lazzarin et al., 2022), to enhance model training. While specific performance metrics are not provided, the study centres on circuit design optimizations and their implications for classification accuracy.

The research in (Sebastianelli et al., 2022) applies a hybrid quantum-classical CNN to the EuroSAT dataset, consisting of Sentinel-2 satellite images across 13 spectral bands and ten land-use categories. A quantum layer is integrated into a modified LeNet-5 architecture, comparing different circuit configurations against classical models. The results indicate that hybrid QCNNs achieve superior classification accuracy, particularly when incorporating entanglement.

A hybrid classical-quantum transfer learning model is introduced in (Zhang et al., 2023), combining a ResNet network for feature extraction with a tensor quantum circuit to fine-tune parameters on the RSI dataset. The study demonstrates improved classification accuracy and reduced parameter requirements compared to conventional CNN-based Remote Sensing Image Scene Classification (RSISC) methods, particularly for small datasets.

The study in (Oh et al., 2021) integrates Quantum Random Access Memory (QRAM) to efficiently load classical image data into quantum states, using quantum convolutional layers to perform feature extraction akin to classical CNNs. While performance details are not explicitly stated, the study suggests advantages in memory efficiency and computational speed.

Lastly, (Huang et al., 2023) proposes a hybrid Hybrid Quantum-Classical Convolutional Neural Network (HQ-CNN) model, integrating quantum convolutional layers with classical neural networks. He utilizes the MNIST dataset, albeit with a reduced-dimensionality subset due to the computational constraints of quantum simulations. The approach leverages parameterized VQCs as quantum convolutional filters, improving feature extraction and robustness against adversarial attacks. Evaluations against Fast Gradient Sign Method (FGSM) and Random Plus FGSM attacks show that HQ-CNNs maintain higher classification accuracy and exhibit slower accuracy degradation than classical CNNs, emphasizing the resilience of quantum kernels in adversarial scenarios.

3 METHODOLOGY

We implemented and evaluated QCNNs for image classification using a pipeline that includes data pro-

cessing, quantum encoding, convolutional layer design, and noise integration to simulate quantum hardware limitations.

The Labelled Faces in the Wild (LFW) dataset (Huang et al., 2008)—comprising grayscale face images—was reduced to a binary classification problem (distinguishing "Colin Powell" from "George W. Bush") with at least 200 images per class; pixel values were normalized to the range [0, 1]

To address hardware constraints, dimensionality reduction was performed using Principal Component Analysis (PCA) to project the data onto a lower-dimensional space with minimal information loss, as described by Eq. 1.

$$X_{PCA} = X \cdot W_{PCA}, \qquad (1)$$

Classical data is then encoded into quantum states via angle embedding, where each input vector is mapped through X-axis rotations (Eq. 2).

$$|\psi\rangle = \prod_{i=1}^{d} R_X(x_i)|0\rangle. \qquad (2)$$

The quantum circuit comprises multiple layers of parameterized single-qubit rotations and entangling Controlled NOT (CNOT) gates (Eq. 3), with dropout implemented via depolarizing channels (Eq. 4) to reduce overfitting.

$$U_{layer}(\theta) = \prod_{i=1}^{d} CNOT(i, (i+1) \bmod d) \cdot$$

$$\prod_{i=1}^{d} R_Z(\theta_{i,1}) R_Y(\theta_{i,2}) R_Z(\theta_{i,3}), \qquad (3)$$

$$\mathcal{E}_{dropout}(\rho) = (1-p)\rho + \frac{p}{2^n}I, \qquad (4)$$

To simulate realistic conditions, depolarizing and phase damping noise channels (Eqs. 5 and 6) are applied after each layer.

$$\mathcal{E}_{\text{depolarizing}}(\rho) = (1-p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z) \qquad (5)$$

$$\mathcal{E}_{\text{phase damping}}(\rho) = (1-p)\rho + pZ\rho Z \qquad (6)$$

The cost function, combining mean squared error and L2 regularization (Eq. 7), is minimized using the Adam optimizer with a decaying learning rate (Eq. 8).

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(x_i;\theta))^2 + \lambda\|\theta\|_2^2, \qquad (7)$$

$$\eta_t = \frac{\eta_0}{\sqrt{t}+\varepsilon}, \qquad (8)$$

Figures 1 and 2 illustrate the quantum circuit of a single QCNN layer and the overall network architecture, respectively.

4 EXPERIMENTATION AND DISCUSSION

This section presents a comparative analysis of QCNNs under two experimental conditions: an ideal noise-free scenario and a realistic noisy environment. The study is based on a binary classification task derived from the LFW dataset, specifically identifying images of Colin Powell among a mixed set of his images and those of other individuals. The experimental implementation leveraged PennyLane for quantum computation, Scikit-learn for dataset preprocessing, and Matplotlib for result visualization. The full experimental code is available in the provided implementation details.

To establish a performance baseline, we built a classical CNN with a level of architectural complexity similar to the QCNN. The CNN consists of three convolutional layers, a fully connected layer, and a ReLU output. It was trained with the same dataset (LFW), preprocessing pipeline, and optimization settings (Adam optimizer, 0.001 learning rate) to ensure fair comparisons.

The results obtained show that the CNN outperforms the QCNN in terms of test accuracy, achieving a solid 86,82%. This performance gap is widely expected, given the efficiency and optimization in training of the classical DL architectures. However, the QCNNs demonstrate other remarkable characteristics that will be exploited in the following analysis.

5 QUANTUM OPERATIONS AND MEASUREMENTS

This section introduces the fundamental quantum operations used in the QCNNs, providing a formal description of state encoding, parameterized rotations, entanglement, and measurement.

Classical input vectors $x \in \mathbb{R}^d$ are embedded into quantum states via angle embedding, which applies single-qubit rotations along the X-axis as follows:

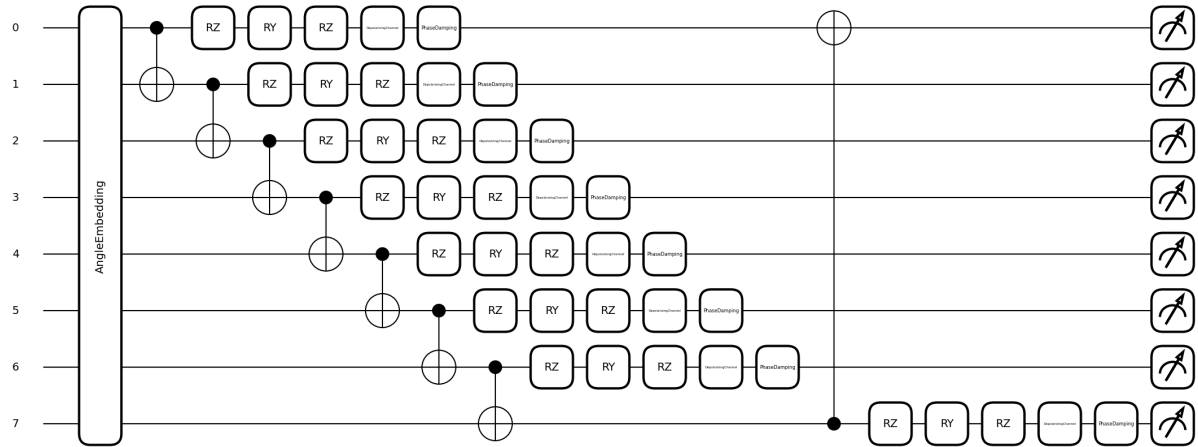$$|\psi(x)\rangle = \bigotimes_{i=1}^{d} R_X(x_i)|0\rangle, \qquad (9)$$

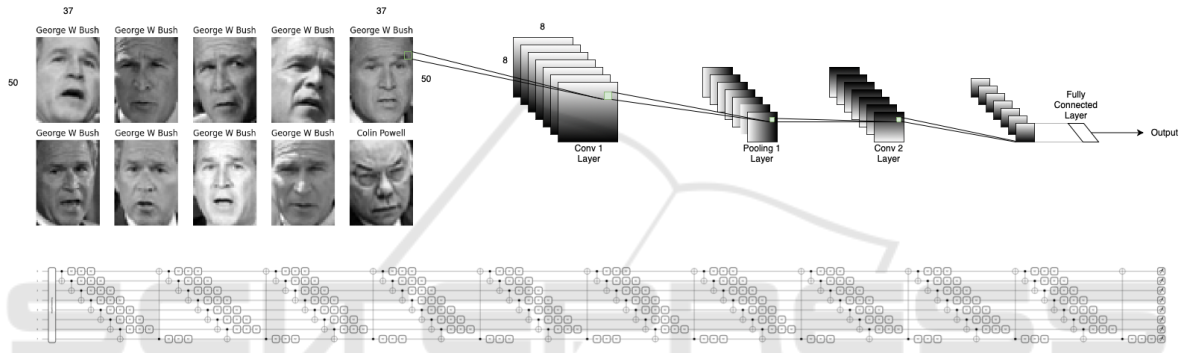Figure 1: Quantum circuit representation of a single QCNN layer.

Figure 2: Overview of the Quantum Convolutional Neural Network applied to the LFW dataset.

where each feature x_i sets the rotation angle via the Pauli-X gate:

$$R_X(\theta) = \exp\left(-i\frac{\theta}{2}X\right) = \begin{bmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{bmatrix}.$$
(10)

Trainable parameters are introduced using single-qubit rotations along the Z- and Y-axes. The Pauli-Z rotation is given by:

$$R_Z(\theta) = \exp\left(-i\frac{\theta}{2}Z\right) = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}. \quad (11)$$

The Pauli-Y rotation is expressed as:

$$R_Y(\theta) = \exp\left(-i\frac{\theta}{2}Y\right) = \begin{bmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{bmatrix}.$$
(12)

Within the QCNN architecture, each qubit undergoes a sequence of rotations $R_Z(\theta_1)$, $R_Y(\theta_2)$, and $R_Z(\theta_3)$ to achieve universal single-qubit operations. Entanglement is introduced using the CNOT gate, whose action in the computational basis is:

$$\text{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \quad (13)$$

This gate flips the target qubit's state when the control qubit is in $|1\rangle$, establishing correlations among qubits.

Finally, classical information is extracted by measuring the expectation value of the Pauli-Z operator:

$$\langle Z \rangle = \langle \psi | Z | \psi \rangle, \quad (14)$$

with the Pauli-Z matrix defined as:

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \quad (15)$$

This measurement provides the classical output used for classification.

5.1 Experimental Framework

The dataset was retrieved using $fetch\_lfw\_people()$ from Scikit-learn, followed by pixel intensity normalization and transformation into one-dimensional feature vectors. PCA was applied to reduce dimensionality, retaining the top eight principal components, and the data was subsequently scaled to the range for quantum embedding. The dataset was divided into training (60%) and testing (40%) subsets, while preserving class balance.

512

Each 8-dimensional feature vector was encoded into quantum states via Angle Embedding along the X-axis. The QCNN structure is implemented in PennyLane, consisting of ten layers, each containing trainable single-qubit rotations and followed by CNOT gates for entanglement. The final circuit measurements were performed using Pauli-Z expectation values across multiple qubits, yielding the classification outcome.

Training was performed using the Adam optimizer, initialized with a learning rate of 0.05. The loss function combined Mean Squared Error (MSE) loss with L2 regularization, aiming to prevent overfitting. The training was conducted for five epochs as an initial assessment and extended to ten epochs for evaluating long-term convergence trends.

The noise-free scenario was executed on the *default.qubit* simulator, ensuring an idealized quantum execution. The noisy setting introduced depolarizing noise (probability of 0.01) and phase damping noise (probability of 0.02) through the *default.mixed* backend, simulating real-world quantum errors.

Each layer of the QCNN consists of 40 quantum gates, calculated as follows. Each qubit undergoes three parameterized single-qubit rotations, contributing to 24 gates for 8 qubits. The entangling CNOT gates contribute another 16 operations, as each qubit is coupled with its adjacent neighbour in a circular pattern. The total number of trainable parameters is 240 across all layers, computed as, corresponding to the three rotation angles per qubit per layer.

If noise is introduced, additional operations are required to model decoherence effects. Specifically, Depolarizing and Phase Damping noise channels are applied after each layer, adding two noise gates per qubit, resulting in 32 additional operations per layer. Therefore, the total number of quantum gates per layer increases from 40 in the noise-free case to 72 in the noisy setting.

The full implementation of the QCNN is available on GitHub[1].

5.2 Performance of the Noise-Free QCNN

The noise-free QCNN demonstrated a training accuracy of 79.34% and a test accuracy of 73.21% after five epochs, showcasing the model's ability to generalize effectively under ideal conditions. The second experiment, extending training to ten epochs, resulted in a training accuracy of 82.45% and a test accuracy of

75.62%, highlighting the benefits of additional training in an idealized setting.

Figure 3 report training accuracy for 5 epochs in the case of no noise.
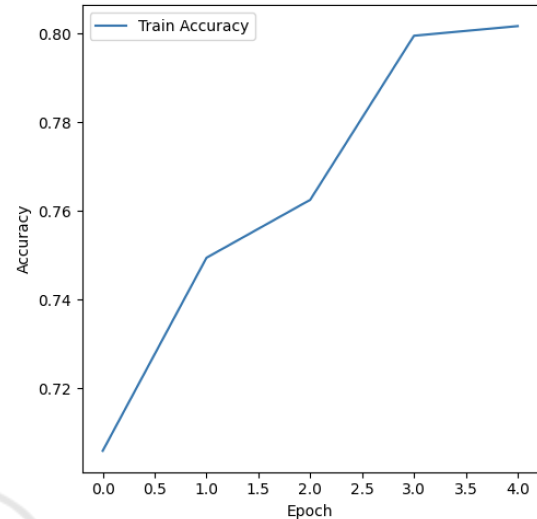

Figure 3: Training accuracy for the noise-free QCNN (5 epochs).

Figure 4 report training accuracy for 10 epochs in the case of no noise.
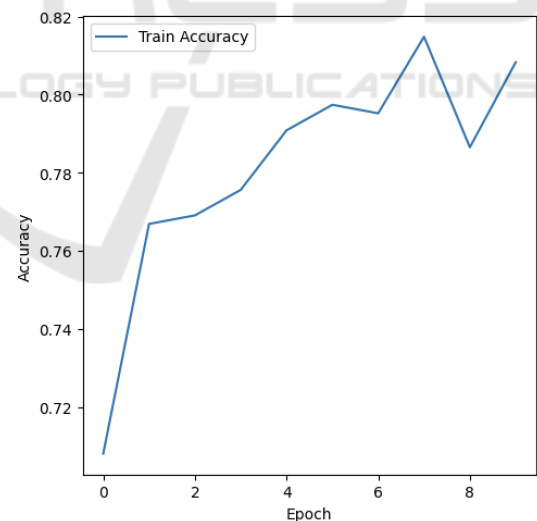

Figure 4: Training accuracy for the noise-free QCNN (10 epochs).

Table 1 summarises both parameters and performance indices for the noise-free QCNN.

---

[1]https://github.com/leliocampanile/IoTBDS-quantumAI

Table 1: Noise-Free QCNN Parameters and Performance.

Parameters	
Name	**Value**
Number of Layers	10
Number of Features (PCA)	8
Number of Trainable Parameters	240
Number of Quantum Gates per Layer	40
Optimizer	Adam
Learning Rate	0.05
Noise Model	None
Indices	
Name	**Value**
Training Accuracy (5 Epochs)	79.34%
Test Accuracy (5 Epochs)	73.21%
Training Loss (5 Epochs)	0.16
Training Accuracy (10 Epochs)	82.45%
Test Accuracy (10 Epochs)	75.62%
Training Loss (10 Epochs)	0.15

5.3 Performance of the Noisy QCNN

With the introduction of noise, the QCNN exhibited a training accuracy of 71.25% and a test accuracy of 69.45%, reflecting the detrimental effects of decoherence and gate errors. In the second experiment, where the number of training epochs was increased to ten, the training accuracy decreased to 69.28% while the test accuracy stagnated at 69.05%. This suggests that prolonged training under noisy conditions does not yield performance improvements, but instead leads to an early plateau. The accumulation of quantum noise disrupts the optimization process, preventing the model from effectively refining its decision boundary. As noise-induced errors accumulate over multiple layers and training iterations, they overshadow the benefits of longer training, ultimately limiting the network's learning capacity and reducing its ability to generalize.

Figure 5 report training loss for 5 epochs in the case of noise.

Figure 6 report training loss for 10 epochs in the case of noise.

Table 2 summarises both parameters and performance indices for the noisy QCNN.

5.4 Theoretical Runtime Analysis

The noise-free QCNN serves as a benchmark for evaluating the theoretical performance of quantum neural networks. The results confirm that in the absence of hardware imperfections, QCNNs can achieve robust classification performance. Conversely, the noisy variant demonstrates the vulnerability of quantum cir-
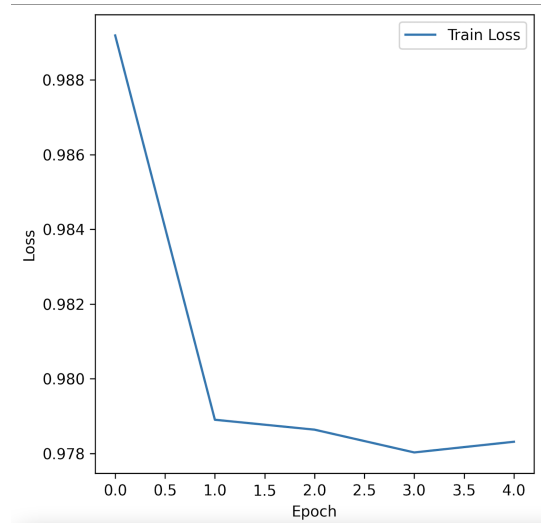

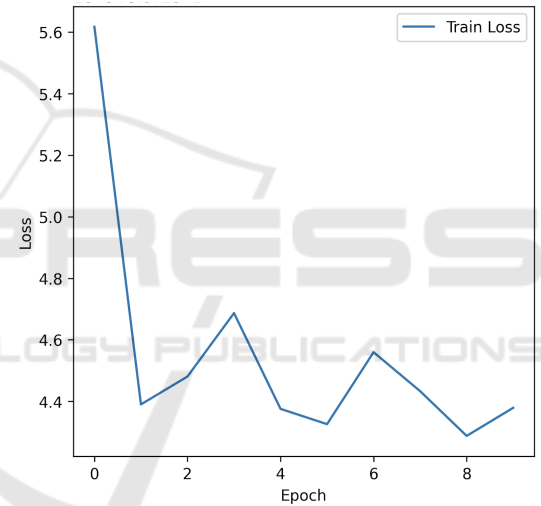Figure 5: Training loss for the noisy QCNN (5 epochs).

Figure 6: Training loss for the noisy QCNN (10 epochs).

Table 2: Noisy QCNN Parameters and Performance.

Parameters	
Name	**Value**
Number of Layers	10
Number of Features (PCA)	8
Number of Trainable Parameters	240
Number of Quantum Gates per Layer (without noise)	40
Number of Additional Noise Gates per Layer	32
Total Number of Quantum Gates per Layer	72
Optimizer	Adam
Learning Rate	0.05
Indices	
Name	**Value**
Training Accuracy (5 Epochs)	71.25%
Test Accuracy (5 Epochs)	69.45%
Training Loss (5 Epochs)	0.97
Training Accuracy (10 Epochs)	69.28%
Test Accuracy (10 Epochs)	69.05%
Training Loss (10 Epochs)	0.44

cuits to real-world imperfections, with noticeable accuracy degradation due to quantum noise.

While increasing the number of layers may improve model expressiveness, it also exacerbates noise-related issues. The results highlight the necessity of quantum error correction and noise-aware circuit design to make QCNNs viable for near-term quantum devices. The findings suggest that while noise presents a significant challenge, careful tuning of learning parameters and error mitigation strategies can help maintain competitive performance levels.

The runtime of the QCNN depends on several key factors: the number of qubits, the number of layers, the batch size, and the computational cost of optimization. The total runtime is derived from three main phases: preprocessing, quantum circuit execution, and optimization. The preprocessing steps include PCA, feature normalization, and train-test splitting. PCA dominates the complexity with an order of $O(Nd^2)$, where N is the number of samples and d is the original feature dimension.

The execution of the quantum circuit for each batch depends on the number of qubits n, the number of layers L, and the number of quantum gates per layer. Each batch requires evaluating all quantum gates, leading to a complexity of $O(BLn)$ per batch. Since the parameter-shift rule for gradient computation requires two additional evaluations per parameter, the total number of quantum circuit evaluations per batch is $O(6BL^2n^2)$.

The Adam optimizer updates parameters iteratively. Each update is proportional to the number of parameters, $O(3nL)$, and for T epochs, the total optimization complexity is $O(TBLn)$. Combining this with gradient evaluations, the dominant term remains $O(6TBL^2n^2)$.

Summing preprocessing, circuit execution, and optimization, the total asymptotic runtime complexity is in Eq. 16.

$$O(Nd^2) + O(6TBL^2n^2) \qquad (16)$$

For a large dataset, quantum training dominates, yielding $O(TBL^2n^2)$.

In the noisy QCNN, depolarizing and phase damping noise are introduced through the *default.mixed* simulator, which requires simulating density matrices instead of pure states. This increases the quantum state dimension from 2^n to 2^{2n}, leading to an exponential increase in computational cost, as in Eq. 17.

$$O(TBL^2n^22^{2n}). \qquad (17)$$

The exponential term reflects the additional cost of simulating a noisy system that must track all pos-

sible mixed states. Table 3 summarises all these considerations.

Table 3: Computational Complexity of QCNN Training.

Scenario	Computational Complexity
Preprocessing (PCA, Normalization)	$O(Nd^2)$
Quantum Circuit Execution	$O(BLn)$
Parameter-Shift Rule for Gradients	$O(6BL^2n^2)$
Optimization (Adam Updates)	$O(TBLn)$
Total Runtime (Noise-Free)	$O(TBL^2n^2)$
Total Runtime (With Noise)	$O(TBL^2n^22^{2n})$

The model without noise scales polynomially with the number of qubits and layers, the model with noise scales exponentially with the number of qubits, significantly limiting the ability to simulate large quantum networks on classical computers.

6 CONCLUSION AND FUTURE WORK

The comparative analysis of QCNNs under noise-free and noisy conditions reveals key insights for facial recognition tasks. In an ideal setting, the QCNNs achieve a test accuracy of 75.62% after 10 epochs, demonstrating strong generalization when quantum decoherence and gate errors are negligible. However, the introduction of noise reduces accuracy to 69.05%, highlighting the detrimental effect of quantum noise. The loss and accuracy curves indicate that the model captures significant patterns, although discrepancies between training and test performances suggest some degree of overfitting, especially in the noise-free scenario. With 40 gates per layer and 240 trainable parameters, the architecture offers sufficient expressivity yet remains constrained by the current limitations of quantum hardware. Increasing the number of epochs improves accuracy in both environments, albeit more modestly in the presence of noise, emphasizing the need for regularization techniques such as quantum dropout and noise-aware training. All experiments were conducted on the HPC server whose specifications are provided in Table 4 4.

Table 4: Specifications of the HPC system used for QCNN training.

Component	Specification
HPC Server	SuperServer 7089P-TR4T (Supermicro)
Hostname	magicbox
CPU	8x Intel Xeon Platinum 8168 (2.7 GHz, 24 cores, 48 threads)
Cache Memory	33MB L3 per CPU
GPU	8x Nvidia P100 (16GB on-board RAM per GPU)
RAM	48x 32GB DDR4 2666 MHz (Total: 1536GB)
Storage (OS)	2x 480GB SSD (RAID 1)
Storage (Data)	4x 960GB SSD (RAID 6)
Operating System	Linux CentOS 7.5

6.1 Future Directions

This study opens several promising avenues to enhance the effectiveness and practical applicability of QCNNs. Incorporating noise mitigation strategies, including quantum error mitigation and correction techniques, could help to bridge the performance gap between noise-free and noisy models. Optimizing the quantum circuit architecture by reducing the number of entangling CNOT gates or by employing more robust quantum feature maps may further improve resilience against hardware imperfections. Expanding the current binary classification framework to multi-class scenarios would provide a more comprehensive assessment of scalability and generalization. Finally, validating QCNNs on real quantum hardware is essential for directly assessing noise effects and making realistic comparisons with classical CNNs. In summary, while this study confirms the feasibility of QCNNs for image classification, further advances in error correction, hardware performance, and training strategies are critical to fully harnessing the potential of quantum neural networks.

ACKNOWLEDGEMENTS

REFERENCES

Chen, G., Chen, Q., Long, S., Zhu, W., Yuan, Z., and Wu, Y. (2023). Quantum convolutional neural network for image classification. *Pattern Analysis and Applications*, 26(2):655 – 667.

Easom-Mccaldin, P., Bouridane, A., Belatreche, A., Jiang, R., and Al-Maadeed, S. (2024). Efficient quantum image classification using single qubit encoding. *IEEE Transactions on Neural Networks and Learning Systems*, 35(2):1472 – 1486.

Gong, L.-H., Pei, J.-J., Zhang, T.-F., and Zhou, N.-R. (2024). Quantum convolutional neural network based on variational quantum circuits. *Optics Communications*, 550.

Hassan, E., Hossain, M. S., Saber, A., Elmougy, S., Ghoneim, A., and Muhammad, G. (2024). A quantum convolutional network and resnet (50)-based classification architecture for the MNIST medical dataset. *Biomedical Signal Processing and Control*, 87.

Hilbert, M. and López, P. (2011). The world's technological capacity to store, communicate, and compute information. *Science*, 332(6025):60–65.

Huang, G. B., Mattar, M., Berg, T., and Learned-Miller, E. (2008). Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*.

Huang, S.-Y., An, W.-J., Zhang, D.-S., and Zhou, N.-R. (2023). Image classification and adversarial robustness analysis based on hybrid quantum–classical convolutional neural network. *Optics Communications*, 533.

Kharsa, R., Bouridane, A., and Amira, A. (2023). Advances in quantum machine learning and deep learning for image classification: A survey. *Neurocomputing*, 560.

Lazzarin, M., Galli, D. E., and Prati, E. (2022). Multi-class quantum classifiers with tensor network circuits for quantum phase recognition. *Physics Letters A*, 434:128056.

Lu, Y., Gao, Q., Lu, J., Ogorzalek, M., and Zheng, J. (2021). A quantum convolutional neural network for image classification. volume 2021-July, page 6329 – 6334.

Oh, S., Choi, J., Kim, J.-K., and Kim, J. (2021). Quantum convolutional neural network for resource-efficient image classification: A quantum random access memory (QRAM) approach. volume 2021-January, page 50 – 52.

Schuld, M. and Petruccione, F. (2021). *Machine learning with quantum computers*, volume 676. Springer.

Sebastianelli, A., Zaidenberg, D. A., Spiller, D., Le Saux, B., and Ullo, S. (2022). On circuit-based hybrid quantum neural networks for remote sensing imagery classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:565 – 580.

Wall, M. L. and D'Aguanno, G. (2021). Tree-tensor-network classifiers for machine learning: From quantum inspired to quantum assisted. *Physical Review A*, 104(4):042408.

Zhang, Z., Mi, X., Yang, J., Wei, X., Liu, Y., Yan, J., Liu, P., Gu, X., and Yu, T. (2023). Remote sensing image scene classification in hybrid classical–quantum transferring cnn with small samples. *Sensors*, 23(18).