ACCEPTED MANUSCRIPT

Research on Image Classification Using Quantum Convolutional Neural Networks Based on Multiple Approaches

To cite this article before publication: Yumin Dong *et al* 2025 *Phys. Scr.* in press https://doi.org/10.1088/1402-4896/adc20e

Research on Image Classification Using Quantum Convolutional Neural Networks Based on Multiple Approaches

Yumin Dong*[1], Xiaofeng Ni†[1],Tonglei Sun[1],Gaojie Wu[1],Changjiang Lin[1]

[1]College of Computer and Information Science, Chongqing Normal University, Chongqing, 401331, China.

*Corresponding author(s). E-mail(s): dym@cqnu.edu.cn;

†Contributing authors: 2023210516059@stu.cqnu.edu.cn;

†These authors contributed equally to this work.

Abstract

With the significant enhancement of quantum computing capabilities and their expanding applications in various fields, quantum image processing combined with quantum computing has become a key area of research in quantum computing, holding important theoretical value. In the face of the challenges posed by the digital information era, image classification technology demands higher standards from machine learning algorithms in terms of accuracy, computational speed, generalization capability, and complexity. This paper studies a series of quantum data encoding methods and parametric quantum circuit architectures, and proposes the application of Hybrid Hierarchical Encoding (HHE) and Hybrid Quantum Convolutional Neural Network (HQCNN). The new encoding method HHE integrates the benefits of angle encoding and enhances the hybrid amplitude encoding design to achieve optimal classification performance without compromising the depth of quantum circuits.HQCNN demonstrates outstanding classification performance and remarkable control over model complexity by leveraging the data processing capabilities of classical convolution and the complex computational power of quantum convolution. Experimental results show that HHE and HQCNN achieve better classification accuracy on the MNIST and Fashion-MNIST datasets compared to traditional classical CNN models, particularly when the number of parameters is significantly reduced.

Key words: Quantum computing; Image classification; Quantum data encoding; Parametric quantum circuit architectures; Hybrid hierarchical encoding; Hybrid quantum convolutional neural network

1 Introduction

With the rapid advancement of information technology, image classification has emerged as a pivotal technology within the realm of computer vision, playing an essential role across various application scenarios[1]. Traditional image classification methodologies predominantly rely on classical computing systems and conventional machine learning algorithms[2]. However, as Moore's

Law gradually wanes, the computational capabilities of classical computers have reached a critical impasse[3]. Consequently, it is imperative to explore novel and efficient computational approaches to meet the escalating demands associated with image classification tasks.

As an innovative computing paradigm, quantum computing introduces new possibilities in machine learning through its potential for large-scale information storage and efficient parallel processing capabilities[4]. Recent years have witnessed significant advancements in quantum computing. In 2019, Google's research team published a groundbreaking article in Nature that announced their superconducting quantum computer had achieved a task requiring thousands of years for classical computers—a milestone referred to as 'quantum supremacy'[5]. This achievement underscores the substantial advantages quantum computers possess for specific computational tasks, heralding a new era in their practical applicability[6]. IBM unveiled a 433-qubit quantum processor at the 2022 Quantum Computing Summit and announced plans to launch a 1,000-qubit processor by 2023[7]. This major development is poised to propel advancements in quantum computing significantly and is expected to furnish scientists and researchers with unprecedented computational power[8]. Ongoing breakthroughs in quantum hardware have established a robust foundation for applying quantum computing to real-world problems[9]. Notably, the concept of quantum machine learning has emerged—an integration of quantum computing and artificial intelligence that offers fresh perspectives on addressing practical challenges[10].

Quantum Image Processing (QIP) is one of the popular tools in the realm of quantum computing-related fields. Representing images in the form of normalized states on a quantum computer facilitates addressing numerous image processing problems, and quantum image representation methods have a decisive impact on the types of QIP tasks and their execution effectiveness[11]. There are now many effective ways to map images into quantum states. In 2009, Le et al.[12] proposed the Flexible Representation of Quantum Images (FRQI). FRQI not only maps grayscale values to the amplitudes of quantum states but also captures and integrates positional information for each pixel within the image, maintaining the normalization of the quantum state. Due to the superposition effect in quantum mechanics, the required representation space decreases exponentially compared to classical images. The representation method proposed by Zhang et al.[13] uses sequences of qubits' basis states to store the grayscale value of each pixel, rather than encoding it into probability amplitudes as in FRQI. This approach is better suited for processing color images and can accurately retrieve digital images from quantum images.Quantum algorithms have also achieved commendable results in this field[14],Grover's algorithm, a quantum search algorithm, can find specific items in an unsorted database faster than traditional computers. On classical computers, searching N entries typically requires O(N) checks, whereas using Grover's algorithm on a quantum computer requires approximately $O(\sqrt{N})$ operations. In relevant scenarios, the Quantum Fourier Transform can be completed with a complexity of $O(\log_2 N)$, offering an exponential speed-up compared to the classical Fast Fourier Transform's $O(N\log_2 N)$.

In the domain of image classification, quantum technology also shows immense potential. Support Vector Machines (SVM) and their quantum versions can more efficiently exploit high-dimensional spaces and generate complex feature mappings and decision boundaries. Quantum versions of K-means clustering can provide exponential speed-ups on very high-dimensional input vectors. Of course, the hottest area has to be quantum machine learning.As an emerging field of study, quantum machine learning amalgamates principles from both quantum computing and machine learning with the objective of enhancing algorithmic performance by leveraging the strengths inherent in quantum computation[15]. Compared to traditional machine learning techniques, quantum machine learning can identify superior solutions more rapidly; thus facilitating more efficient data process-

ing and analysis endeavors[16]. Furthermore, this domain exhibits considerable promise along with extensive application prospects across various tasks such as image classification, speech recognition, and recommendation systems[17]. Within the context of noisy intermediate-scale quanta (NISQ), core computations within these algorithms are typically executed using parametric quantum circuits (PQC)[18], which adapt their functionality based on parameter adjustments tailored to diverse task requirements. Thanks to phenomena like quantum parallelism, PQC can substantially diminish algorithmic complexity when handling high-dimensional datasets that pose challenges for traditional systems while also enabling data mapping into quantized state spaces[19] . A notable research approach within this domain involves employing Quantum Convolutional Neural Networks (QCNNs), which ingeniously incorporate fundamental concepts from convolutional neural networks into frameworks governed by principles of quantum mechanics[20] . By harnessing superposition and entanglement characteristics intrinsic to qubits , QCNNs effectively extract salient features from images while expressing them efficiently[21] .QCNNs have advantages over traditional CNNs, such as reducing algorithm complexity, increasing operational speed, and decreasing energy consumption[22]. This means that when dealing with large-scale image data and high-resolution image classification tasks, QCNNs can extract key information from images more quickly, thereby improving classification accuracy and practical application outcomes. In particular, Cong and colleagues introduced a quantum circuit architecture that is fully parameterized and draws inspiration from CNNs. They demonstrated its effectiveness in solving certain quantum many-body problems[23]. However, both classical CNNs and QCNNs still need improvements in terms of grading performance when handling small sample image grading tasks[24]. To address this challenge, we introduced a parameterized quantum circuit model for supervised classification problems. The study used parametric quantum circuits (PQC) and harnessed the properties of quantum entanglement, which allows quantum convolution to potentially surpass classical CNNs that only capture local correlations. We employed different quantum encoding methods, proposed hybrid hierarchical encoding (HHE), combined with parameterized quantum gate circuits, and tried a mixed use of classical convolution and quantum convolution ——Hybrid Quantum Convolutional Neural Network(HQCNN), conducting separate tests. With the help of cost functions and optimizers, we benchmarked parameterized quantum convolutional neural networks. Using a smaller number of parameters, ranging from 12 to 51 free parameters, HHE and HQCNN models demonstrated experimental precision superior to classical CNNs. When tested on standard datasets MNIST and fashion MNIST on the Pennylane[25, 26] platform, the best results showed that the precision for MNIST was approximately 98%, and for fashion MNIST, it was around 94%. We concluded that HQCNN has advantages over traditional QCNN and CNN.

2 Methods

Convolutional Neural Networks (CNNs) are one of the representative deep feedforward neural network algorithms of deep learning, showing excellent performance when dealing with grid-structured image data [27, 28]. These models are currently some of the most commonly used in the realm of image classification[29, 30]. A typical CNN is composed of fundamental modules such as the input layer, convolutional layers, pooling layers, fully connected layers, and the output layer, with the benefit of capturing the local features of images.In the past few years,there has been a rise in research on Quantum Convolutional Neural Networks(QCNNs). The following is an overview of the design of QCNNs utilizing Parameterized Quantum Circuits(PQC).

2.1 Quantum Convolutional Neural Networks

Based on the reference[31], we became acquainted with a very practical hierarchical quantum classifier that utilizes Multi-Scale Entanglement Renormalization Ansatz (MERA) rather than simple tree tensor networks (TTNs), representing a more complex network architecture. This structure uses n qubits to represent a 2n dimensional data vector, enabling the model to express more complex circuits with higher representational power. Consequently, this allows for relatively shallow circuit depths. Most importantly, these structures usually have fewer parameters and shallower depths, suggesting they may mitigate the impact of the "barren plateau" phenomenon. Specifically, the design of MERA circuits takes into account the hierarchical structure of quantum information, which aids in better gradient propagation and avoids vanishing gradients, thus ensuring the trainability of the model[32].

Based on the characteristics of the aforementioned model, we opted for Parameterized Quantum Circuits (PQCs). In our selected quantum architecture, since two-qubit gates require up to 15 parameters, the number of optimized parameters within a given layer m is $15(\mathrm{l_m}+1)$. According to the inspiration from[31], the maximum number of tunable parameters in the circuit can be $15\sum_{m=1}^{\log_2(n)}(l_m+1)$, the convolution operation will iteratively proceed based on the number of tunable parameters in the circuit. We know that QCNNs, due to their translational variance feature, ensure that the blocks of parametrized quantum gates are identical within the same layer. The m-layer quantum state can be represented as

$$|\psi_m(\theta_m)\rangle\langle\psi_m(\theta_m)| = \mathrm{Tr}_{A_m}(U_m(\theta_m)|\psi_{m-1}\rangle\langle\psi_{m-1}|U_m(\theta_m)^\dagger) \tag{1}$$

In which $\mathrm{Tr}_{A_m}(.)$ is the trace operation on subsystem $A_m \in \mathbb{C}^{2^{n/2^m}}$,U_m includes unitary operations within the parameterized circuit, and $|\psi_0| = |0|^{\otimes n}$.

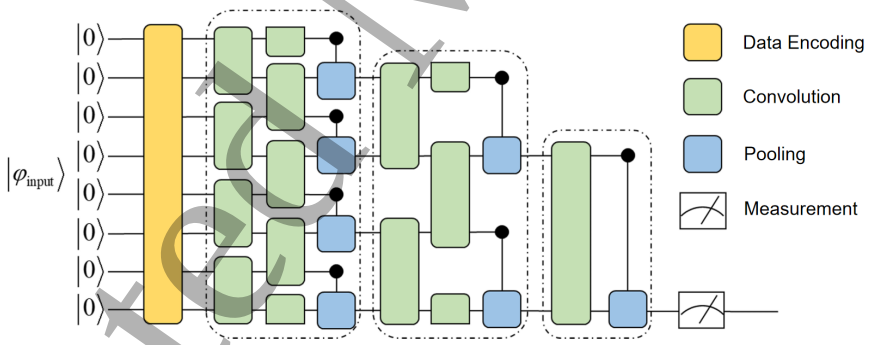


Figure 1: 8-qubit QCNN Structure Diagram

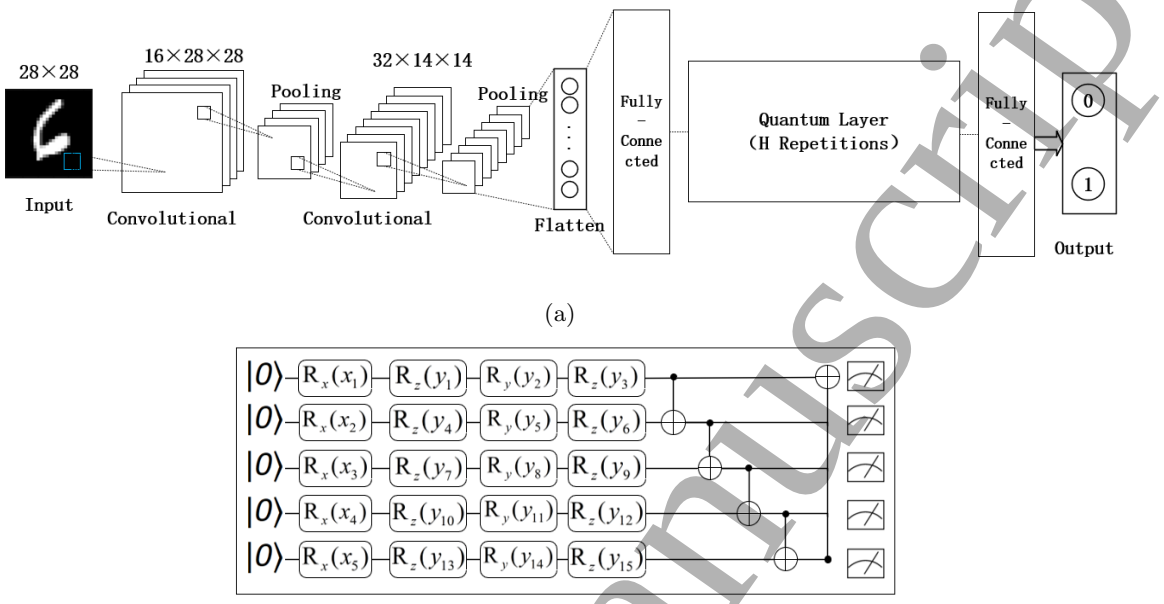
The quantum convolutional layer (QCL) and the quantum pooling layer (QPL) fully exploit unique properties of quantum mechanics, such as superposition, entanglement, and the translational invariance of quantum gate operations. These characteristics enable quantum circuits to exhibit capabilities that surpass those of traditional CNN models when processing classical data. The QCL leverages the global attribute of quantum entanglement, not just local correlations, allowing it to capture broader or deeper patterns and achieve good classification accuracy even with

a relatively small number of free parameters. The QPL simplifies data by reducing the dimensionality of representations while retaining the most critical information, effectively decreasing the amount of information that subsequent layers need to process. This reduces model complexity, enhances training efficiency, and maintains high classification performance. Additionally, because the operation of the QPL is based on the translational invariance of quantum mechanics—where the same parameters are applied to each pair of qubits—it can efficiently handle spatial or temporal translations, which is crucial for tasks like image recognition.The convolution and pooling parts are designed to use the same quantum circuit module to fit the two-qubit composed circuits. If in the case of classical neural networks, the number of parameters to be optimized grows in O(log(n)). If we make full use of this characteristic for encoding the convolutional structure, we can suppress the number of parameters exponentially. We present the 8-qubit binary classification quantum convolutional structure shown in Figure 1. After classical data undergoes quantum encoding, it will pass through convolution and pooling operations via quantum gates. During this process, due to the translational invariance of quantum mechanics, each layer of the network will use the same parameters to operate on each pair of qubits. Through the halving of features via pooling, the final data feature is reduced to the size of a single qubit for measurement.

2.2 Hybrid Quantum Convolutional Neural Network

To fully utilize the advantages of both classical and quantum systems, we constructed a hybrid system by combining a classic convolutional layer with a parallel quantum layer, and tested it on the MNIST dataset using the same coding method as a regular QCNN. The quantum layer functions similarly to a fully connected layer in a traditional system.

The figure 2 above shows the basic structure of hybrid convolutions, consisting of two parts that form the classic convolutional component, followed by a fully connected layer. In the classic convolution section, after receiving input from the input layer, we will use a 5×5 convolution kernel to create output channels, generating 16 output channels. Compared to the commonly used 3×3 convolution kernels, we utilize this larger size to capture more image features[33, 34]. By repeatedly applying convolution and pooling operations, the data samples that maintain the original image size are dimensionally reduced and relevant image features are extracted in the classic convolution layers. After being transformed by the classic convolution layers, the input data is flattened and fed into the hybrid quantum layer of the HQCNN.As shown in the figure, the quantum layers within the hybrid quantum layer are implemented using PQC. This leads to a decrease in the overall runtime due to its parallel processing capabilities.The 3277 convolutional part's features are used as the first part's input, transforming the feature map into n features determined by the code. The output of the quantum layer is sent to the second classic fully connected layer, which maps the features to the final output features for image classification. Next, we introduce the composition of the quantum layer. We use the principle of parameter shift to determine the final quantum layer's circuit structure, consisting of h parallel PQC quantum layers that include embedding, entangling gates, and measurements. The quantum layer receives n features from the classic convolution, divided into h parts, each with q values satisfying $x = (\varphi_1, \varphi_2, ..., \varphi_q) \in \mathbb{R}^q$, generated by quantum coding. The coding part ends with the PQC part, which consists of rotation gates and CNOT gates for subsequent operations[35]. Rotation gates are responsible for data transformation, while CNOT gates generate entanglement. It is important to recognize that the differentiation parameters vary at different depths within the PQC. To calculate weights, if the depth of the PQC is H, the weight calculation of L parallel quantum lines is q*3H*L. After completing these operations, measurements



(a)

(b)

Figure 2: Basic structure of HQCNN. (a) represents the complete HQCNN architecture, and (b) shows the quantum layer architecture.The data is first subjected to dimensionality reduction through a classical structure. The data first undergoes dimensionality reduction through a classical structure. Once processed to match the qubit number requirements of the quantum layer, the data is flattened and sent to the quantum layer. The quantum layer is implemented with PQCs, allowing for simultaneous execution, which reduces runtime. The quantum layer consists of a repeated design of 5-qubit layers with 3 strongly entangled layers, determined by the parameter shift rule, and this pattern is repeated H times. After processing the data, the quantum layer outputs are measured and then fed into a second classical fully connected layer.

are performed on Pauli matrices, resulting in

$$M(i) = \langle 0|E(\varphi_i)^\dagger T(\vartheta)^\dagger \Gamma_i T(\vartheta) E(\varphi_i)|0\rangle \tag{2}$$

Where Γ_i represents i-th quantum bit's Pauli-Y matrix,$E(\varphi_i)^\dagger$ and $T(\vartheta)^\dagger$ are the coding and trainable parts,ϑ is the parameter. The remaining output data can be connected by establishing new fully connected layers to form new quantities, and probability distribution forms the output of each output image is 0, 1 between correlation, each neuron represents the class probability. The model can solve different classes' classification problems by setting the input and output channels of the quantum layer and fully connected layer, having high classification potential.

2.3 Quantum Data Encoding

After classic data preprocessing, images are converted to quantum states using quantum rotation gates Rx(θ) etc., and quantum state information becomes the input of the quantum convolutional neural network after a series of transformations, known as feature mapping, only quantized data can allow classic data to train on quantum lines[8]. The following section presents the data encoding method utilized in this article.

2.3.1 Amplitude Encoding

As one of the most common methods of quantum-classical coding, amplitude encoding is a way to project classic information onto the probability amplitudes of quantum states[36]. Amplitude encoding allows us to store information in an exponential compression manner because n quantum bits can represent 2n different states, each with its own complex probability amplitude. Assume we have a conventional data vector $\vec{x} = (x_1, x_2, ..., x_N)^T$,$N = 2^n$,n is the number of quantum bits. The objective of amplitude encoding is to transform this vector into a normalized quantum state $|\psi\rangle$,so that the probability amplitude of the quantum state corresponds to the element of the data vector. That is

$$|\psi\rangle = \sum_{i=0}^{N-1} x_i |i\rangle \tag{3}$$

Here $|i\rangle$ is the computational basis state of n quantum bits,x_i is the normalized element of the classic data vector $\vec{x}$,meeting normalization conditions:

$$\sum_{i=0}^{N-1} |x_i|^2 = 1 \tag{4}$$

Through amplitude encoding, the amount of parameters and classic data in quantum computers can be presented in an exponential reduction. This could lead to significant benefits in the QCNN algorithm.

2.3.2 Angle Encoding

Angle encoding is a separate form of data representation in quantum computing that distinguishes itself from amplitude encoding by encoding data onto thephase or angle parameters of quantum states instead of directly mapping it to the probability amplitude of those states [37, 38].

Angle encoding normalizes all data, e.g.,$|\phi(x)\rangle = \cos\left(\frac{x_i}{2}\right)|0\rangle + \sin\left(\frac{x_i}{2}\right)|1\rangle$,where x_i is normalized to [0, π) (i=1, ... , N). Angle encoding embeds x values into the Bloch sphere through rotation gates, the rotation angle determined by corresponding values, for Ry rotation gate, the rotation angle is θ, the corresponding data coding can be expressed as

$$R_y(\theta(x)) = e^{-i\frac{\theta(x)}{2}Y} \tag{5}$$

Where Y is the Pauli-Y matrix, and $\theta(x)$ is the angle specified by data x. The quantum bits encoded by this method are unentangled, so the quantum state is separable.

2.3.3 Dense Angle Encoding

From theory, two orthogonal axes on the Bloch sphere can produce any single quantum bit state, so we can encode two types of data to the same quantum bit through rotation angles, called dense angle encoding[39]. By rotating around two orthogonal axes, we can code data x as:

$$|\psi\rangle = \bigotimes_{m=1}^{M/2} \mathrm{e}^{-\mathrm{i}\frac{x_m}{2}\sigma_x} \mathrm{e}^{-\mathrm{i}\frac{x_{M/2+m}}{2}\sigma_y} |0\rangle \tag{6}$$

Where i is the imaginary unit, σ_x and σ_y correspond to Pauli-X and Pauli-Y matrices. Respectively, through this dense angle encoding, the number of required qubits can be reduced by half, while still retaining the advantages of the original angle encoding.

2.3.4 Hybrid Hierarchical Encoding

Given the advantages of layered classifiers,we recognize the benefits of using shallower circuit depths.Both angle and amplitude encoding techniques can help cut down the depth of quantum circuits,with amplitude encoding being particularly advantageous in reducing the required number of qubits. For NISQ devices, deep circuits lead to decoherence and noise effects. To balance these factors, we consider combining angle and amplitude encoding. We first designed hybrid amplitude encoding[40, 41], considering the advantages of both angle and layered encoding. Essentially, hybrid amplitude encoding is the parallel realization of amplitude encoding across multiple independent qubit blocks, if m represents the number of qubits after amplitude encoding, then each block can code $\mathrm{O}(2^m)$ data. If we have n such blocks, the system will contain $n2^m$ data. Adjusting m, n values can adjust the width and depth of the circuit, when n is at its maximum value, the mixed encoding is equivalent to amplitude encoding. Its mixed encoding is represented as:

$$U_\phi(x): x \in \mathbb{R}^N \rightarrow |\phi(x)\rangle = \bigotimes_{j=1}^{n} \left(\frac{1}{\|x\|_j} \sum_{i=1}^{2^m} x_{ij} |i\rangle_j \right) \tag{7}$$

This encoding method is flexible, but due to its different regional amplification ratio problem, it still leads to information inaccuracy issues. Our proposed hybrid hierarchical encoding (HHE), similar to hybrid amplitude encoding, aims to resolve its inflexibility issue by no longer managing each region separately, instead, we standardize all classical data uniformly. HHE also divides N qubits into m independent n-qubit regions, the difference lies in the amplitude encoding within each block and other aspects. Assuming we have M-dimensional standardized classical data vector $x = (x_1, x_2, x_3, ..., x_M)$,where x_m satisfies normalization conditions:

$$\sum_{m=1}^{M} |x_m|^2 = 1 \tag{8}$$

In our m-blocked qubits, the n-qubit region meets condition $\mathrm{n} \geqslant 1\log_2(\mathrm{M})$,each area can code $2^n - 1$ classical data. For every n-qubit region, we use controlled rotation gates for amplitude encoding. We'll apply a series of rotation and controlled gates to each n-qubit region, these gates operate based on different control conditions. Given an n-qubit region, the encoding process can be expressed as:

$$|\psi\rangle = \left(\bigotimes_{i=1}^{n} \prod_{j=1}^{2^{i-1}} U_{ij}\left(x_{ij}\right) \right) |0\rangle^{\otimes n} \tag{9}$$

In the notation U_{ij}, the subscript i denotes the ith quantum bit line, while the subscript j indicates the jth unitary operation on that particular quantum bit line. Thus, U_{ij} represents this specific unitary operation.

2.4 Quantum Convolutional Neural Network Circuit Design

In the design of PQCs, we ensure that all filters use the same circuit, a constraint that also applies to pooling operations. The classification performance of QCNNs is directly affected by our choice of circuit, and the number of parameters for different convolution kernel circuits will gradually increase, as shown in Figure 3. Where $R(\theta)$ represents a rotation by a specific angle θ around an axis (denoted here by i) of the Bloch sphere, and $\mathrm{U3}(\theta, \Phi, \lambda)$ is an arbitrary single-qubit gate. Circuit 1 is referred to as a tree tensor network[42], Circuit 7 is the special unitary group SU(4), which can simulate combinations of any two-qubit gates, and Circuit 8 is an approximate equivalent circuit of SU(4).SU(4) refers to the set of 4×4 complex matrices that are unitary and have a determinant of 1. An SU(4) circuit is a quantum circuit capable of implementing SU(4) transformations, which can be used to perform all possible two-qubit operations.By adjusting the quantum states via X and Z rotation gates and establishing entanglement relationships through the CNOT gate, it is possible to realize SU(4) operations by tuning the adjustable parameters θ. Circuits 7 and 8 embody simplified configurations that have been found to exhibit optimal expressibility.

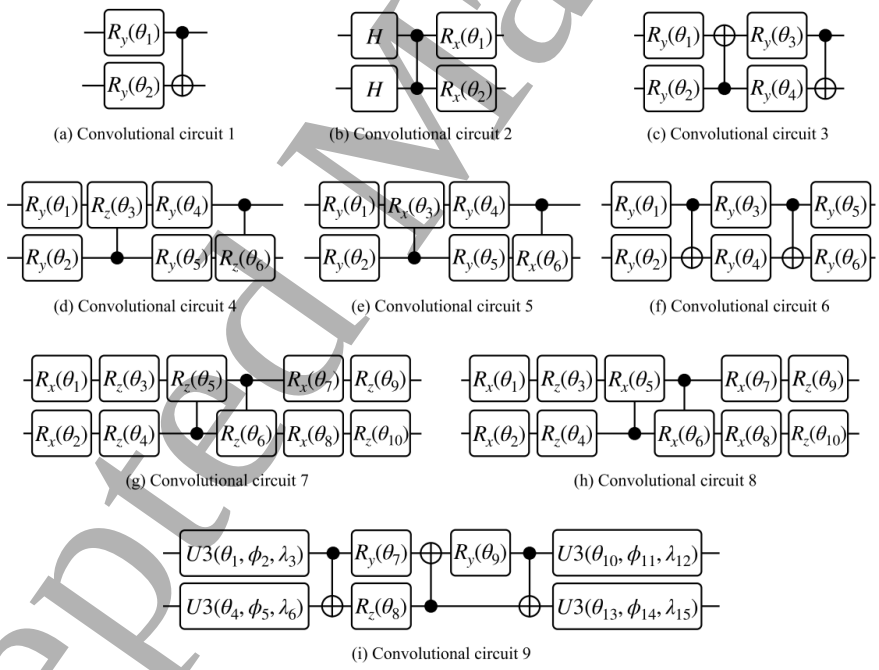


Figure 3: Design of Quantum Convolutional Circuits

In the design of the pooling layer, we apply controlled rotation X gates and Z gates, where the rotation operations are activated when the qubit is in state 1 (solid circle) or state 0 (hollow circle).

The control (first) qubit is traced out after the gate operations, as shown in the figure below. This approach achieves the purpose of dimensionality reduction.

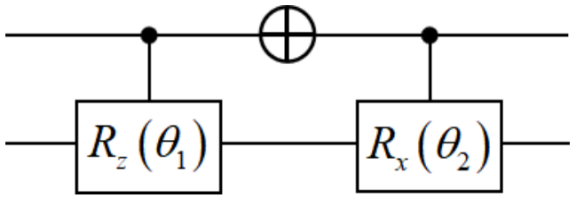


Figure 4: Design of Pooling Layer Circuits

2.5 Classical Data Preprocessing

Due to the limitations of current quantum computing hardware,especially considering that NISQ(Noisy Intermediate-Scale Quantum)devices can only support relatively shallow circuits for qubits,the system finds it difficult to directly and efficiently process high-dimensional image data.As a result,we implemented traditional data preprocessing techniques to decrease the data's dimensionality.We used two types of data preprocessing: Principal Component Analysis (PCA)[43] and autoencoders[44]. Both methods effectively reduce the dimensionality of the data,allowing images to be encoded and further processed on a limited number of qubits.An additional benefit of using PCA is that, for simple classification tasks, PCA not only performs stably but also requires minimal computational resources, making it an excellent preprocessing method.

3 Simulation Result

In this section, we will report the simulation experiment results of applying different coding methods to different quantum convolution circuits, comparing the results under PCA and autoencoder dimensional reduction methods, and training on different datasets. In testing, we will use binary cross-entropy loss as the loss function. Cross-entropy loss, as a common benchmark in neural network training tasks, is an important criterion for measuring model classification performance. In quantum mechanics, the binary cross-entropy loss can be expressed as:

$$L_C(\theta) = -\frac{1}{2m}\sum_{i=1}^{m}\left[y_i \ln C(x_i;\theta) + (1-y_i)\ln\left(1-C(x_i;\theta)\right)\right] \tag{10}$$

Where C is the model itself, m is the batch size,y_i is the corresponding label. During training, we use mini-batch gradient descent, where each iteration involves randomly selecting data from the dataset to create a mini-batch. This approach reduces simulation time and aids the gradients in escaping local minima.We will use the Nesterov momentum optimizer provided by the PennyLane framework to minimize the loss function.[45], with learning rate set to 0.01 and iteration times set to 200.

Before making the comparison, we first compared the loss and test accuracy of ordinary QCNN and hierarchical structures under different PCA and autoencoder processing, as shown in Figure 5.

In the comparison between 8 features and 16 features, it is clear from the figure that the hierarchical structure exhibits better convergence as the epochs progress. Additionally, the hierarchical structure demonstrates superior performance in terms of the time taken to reach maximum accuracy and the stability of accuracy, which also reflects its ability to better avoid the barren plateau phenomenon for the QCNN architecture.

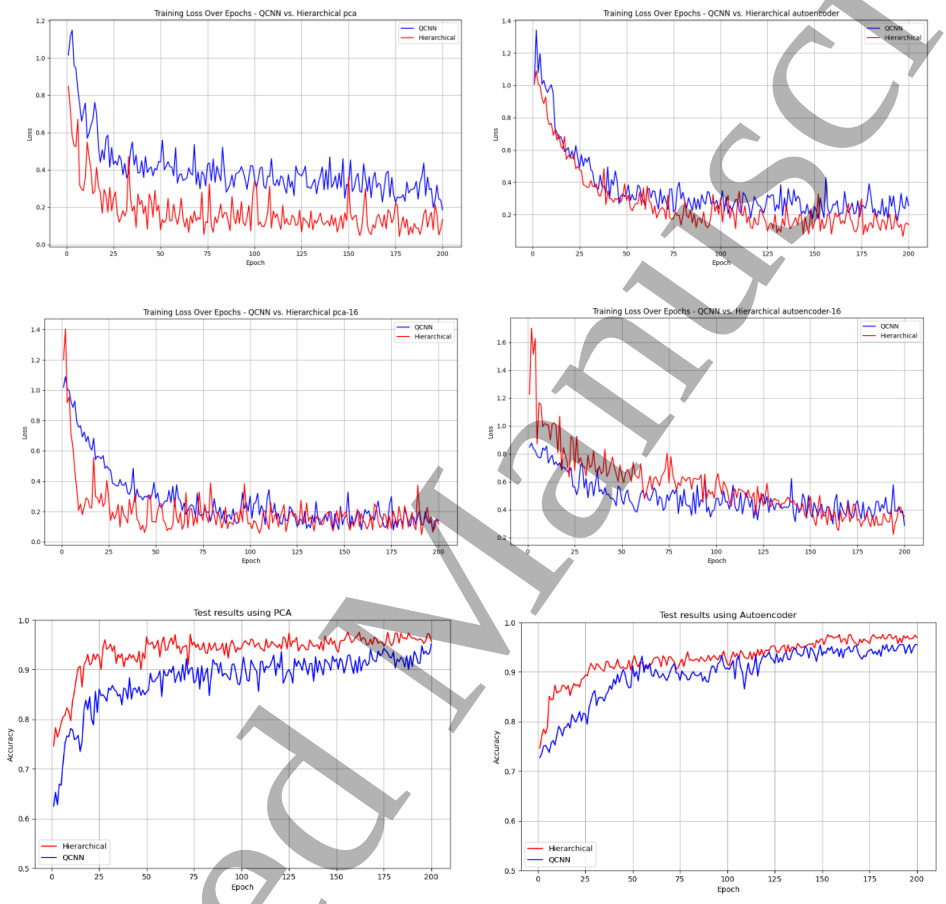


Figure 5: Training loss and test accuracy results Between QCNN and Hierarchical Structures

3.1 Classification on the MNIST Dataset

In this section, we conduct classification experiments on the MNIST dataset using different quantum circuits and various encoding methods. For all experimental results, we utilized the cross-entropy loss function. As shown in Figure 6, the quantum convolution circuit numbers in the table correspond to the quantum convolution circuits displayed in Figure 3. Additionally, we added a hybrid quantum convolution circuit as Circuit 10 for performance comparison. The average precision of all circuits across different encoding schemes is presented in Table 1.

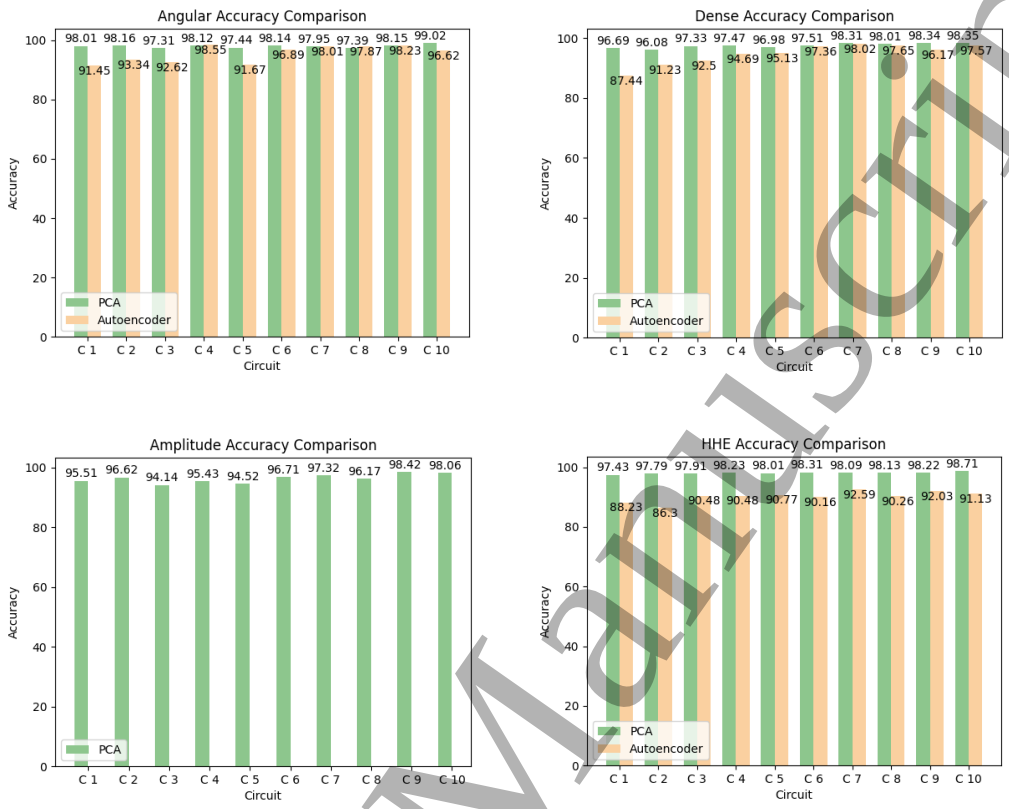


Figure 6: Accuracy of QCNN Under Different Conditions on the MNIST Dataset

The experimental results indicate that both standard angle encoding and dense angle encoding exhibit excellent performance across various circuits in the MNIST dataset, particularly under PCA processing. It is speculated that the inadequate performance of the autoencoder is attributed to a lack of sufficient iteration depth. Among all encodings and circuits, the highest accuracy achieved was 99.02% with angle encoding on Circuit 10. Additionally, Circuit 10, which uses hybrid quantum convolution, also demonstrated a relatively higher average accuracy across all encoding methods, preliminarily indicating its strong potential for classification tasks. From the experiments, we can also observe that HHE achieved good initial results under PCA processing, demonstrating satisfactory classification accuracy even in circuits with fewer parameters. In terms of average classification accuracy on the MNIST dataset, HHE generally outperformed traditional amplitude encoding, while showing mixed results compared to angle encoding.

3.2 Classification on the Fashion-MNIST Dataset

To better reflect broader generalization capabilities, we conducted a second set of experiments using the Fashion-MNIST dataset to avoid the over-simplicity of MNIST leading to poor generalization. The experimental results are shown in Figure 7 and Table 2.

Table 1: Average Accuracy for Each Circuit and Encoding Method

Circuit	Pre-processing	Angular	Dense	Amplitude	HHE
Circuit 1	pca	97.76	96.50	93.31	97.13
	autoencoder	89.45	83.44		85.23
Circuit 2	pca	97.76	95.58	94.52	95.69
	autoencoder	91.74	90.13		84.10
Circuit 3	pca	97.11	95.73	93.84	97.71
	autoencoder	90.82	91.38		89.78
Circuit 4	pca	97.72	97.07	96.03	97.73
	autoencoder	95.35	93.59		89.28
Circuit 5	pca	97.04	96.48	92.32	97.91
	autoencoder	89.57	94.03		89.47
Circuit 6	pca	97.74	97.21	94.51	98.11
	autoencoder	90.67	94.63		89.76
Circuit 7	pca	97.65	97.16	95.12	97.99
	autoencoder	96.01	95.82		90.59
Circuit 8	pca	96.99	97.91	95.47	98.03
	autoencoder	95.47	94.85		89.66
Circuit 9	pca	97.65	97.94	97.22	98.02
	autoencoder	94.03	94.77		90.03
Circuit 10	pca	98.42	97.55	97.86	98.61
	autoencoder	94.42	95.87		90.13

Compared to the MNIST dataset, the complexity increase in the Fashion-MNIST dataset led to an overall decline in model classification accuracy. Under these conditions, we found that, unlike the patterns observed in MNIST, the overall accuracy of autoencoders for dimensionality reduction is noticeably higher than that of PCA. This suggests that autoencoders are more suitable for complex tasks and are more flexible. From the figure, it is evident that HHE, although experiencing a drop in accuracy with more complex tasks, still achieves an overall higher average accuracy compared to conventional amplitude encoding and angle encoding. For Circuit 10, the hybrid classical convolutional circuit, we still see its outstanding potential for complex classification tasks. The hybrid quantum convolutional circuit (Circuit 10) we proposed achieves excellent results across various datasets, attaining a best accuracy of 99.02% on simple tasks and the highest average precision. It also delivers superior average performance on complex tasks regardless of the encoding method. Of course, each circuit and encoding method has its own strengths, and selection should be based on specific needs.

3.3 Comparative Experiments

3.3.1 Comparison of Different QCNNs

In recent years, the field of quantum convolution has seen rapid development, and through literature reviews, we have come across several notable QCNN models. In this section, we compare similar QCNN models to those proposed in our work. According to[46], the structure described

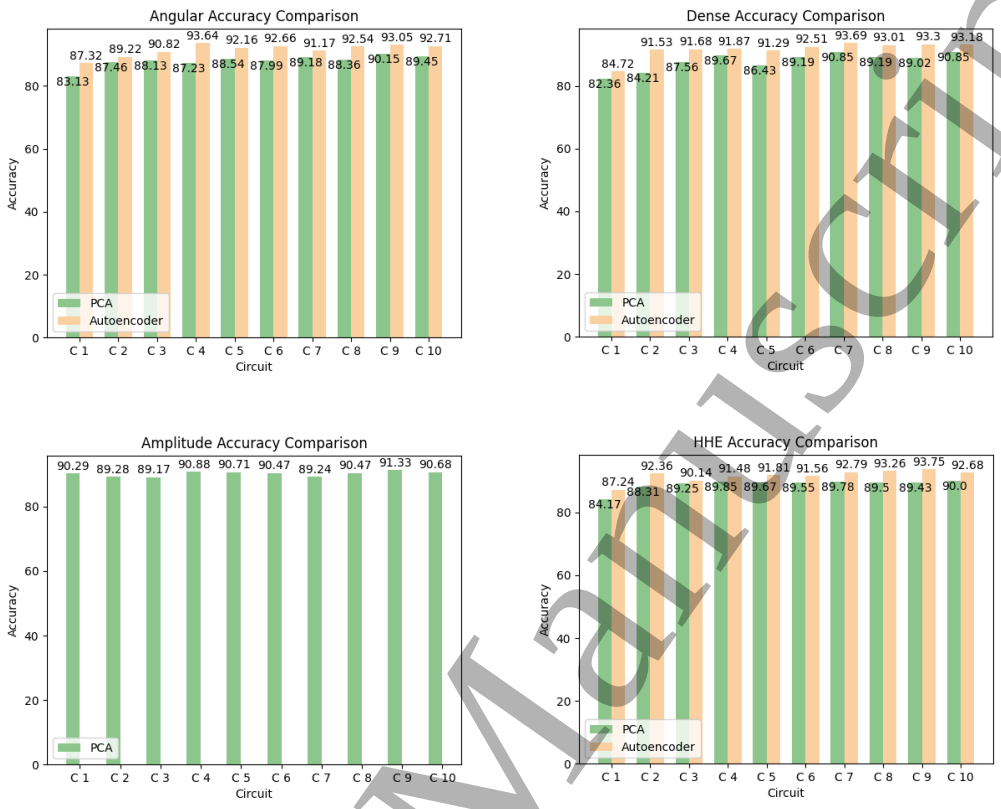


Figure 7: Accuracy of QCNN Under Different Conditions on the Fashion-MNIST Dataset

in the literature resembles the quantum convolutional structure presented in this chapter. We will compare their circuits with some of ours. Circuits 1, 9, and 10 will be used for comparative experiments, continuing with classification on the MNIST dataset, and the accuracy and parameter results are shown in Table 3. The structure in [46] uses PCA for dimensionality reduction, reducing the data dimensions to 64 and employing conventional amplitude encoding. In our experiment,we utilize PCA to decrease the data dimensions to 30,and HHE for encoding purposes. Compared to the model in the literature, the model using HHE for encoding demonstrates clearly better classification performance, attributed to different designs in preparing quantum states. The model in the literature aimed to reduce the drawbacks of amplitude encoding, i.e., excessive resource consumption, by designing an approximate encoding model, which successfully reduced the depth of the quantum circuit but at the cost of accuracy. The circuits that utilize HHE preserve the benefits of angle encoding and enhance the design of traditional hybrid amplitude encoding, ensuring the quantum circuit's depth remains low while sustaining superior classification performance. Hybrid Quantum Convolutional Neural Networks (HQCNN) outperform other QCNNs in terms of classification accuracy and parameter efficiency, primarily attributed to the incorporation of classical convolution modules. This integration endows hybrid quantum convolutions with both the computational strengths of quantum convolutions and the data processing capabilities of classical

Table 2: Average Accuracy for Each Circuit and Encoding Method

Circuit	Pre-processing	Angular	Dense	Amplitude	HHE
Circuit 1	pca	80.91	81.05	89.29	83.47
	autoencoder	86.12	82.52		85.64
Circuit 2	pca	85.86	83.41	87.88	87.11
	autoencoder	87.82	89.33		90.36
Circuit 3	pca	86.53	86.56	87.47	88.05
	autoencoder	88.42	89.18		88.78
Circuit 4	pca	85.03	88.07	88.18	88.35
	autoencoder	92.24	90.27		90.08
Circuit 5	pca	86.04	84.13	88.61	88.57
	autoencoder	90.76	90.89		90.81
Circuit 6	pca	85.09	86.79	88.01	88.17
	autoencoder	91.06	90.61		90.36
Circuit 7	pca	86.38	87.95	88.48	88.08
	autoencoder	90.67	91.49		91.99
Circuit 8	pca	87.46	86.59	88.57	88.18
	autoencoder	90.54	91.41		91.76
Circuit 9	pca	87.65	88.02	90.03	88.23
	autoencoder	91.15	91.80		91.95
Circuit 10	pca	87.95	90.35	89.98	89.00
	autoencoder	91.61	91.68		92.08

convolutions.

Model	Params	Accuracy
[[46]]	54	96.53%
Circle 1	12	97.43%
Circle 9	51	98.22%
Circle 10	45	98.71%

Table 3: Comparison of Highest Classification Accuracies on the MNIST Dataset Across Different Circuits

3.3.2 Comparison with CNN

The purpose of this section is to demonstrate the benefits of quantum convolution with HHE encoding compared to conventional CNNs. We will conduct two comparative experiments. The first comparison involves testing QCNN with HHE encoding against a classic CNN, ensuring that the number of parameters in both QCNN and CNN is approximately equal. Both will use PCA for dimensionality reduction. For the classic CNN, the number of parameters at different dimensions after data reduction are as follows: 26 parameters when the dimension is 8, and 34 parameters

when the dimension is 16. Therefore, we will use circuits with similar parameter counts for the comparison. We will select Circuit 8, which uses HHE encoding, with parameter counts of 24 and 36 under similar conditions. Figure 8 illustrates the comparison of loss values between QCNN and CNN. By comparing them,it is evident that a typical QCNN has slightly inferior convergence performance compared to a classic CNN. In terms of classification accuracy, as shown in Figure 9, the QCNN using HHE encoding achieves better classification results on both the MNIST and Fashion-MNIST datasets compared to the classic CNN.

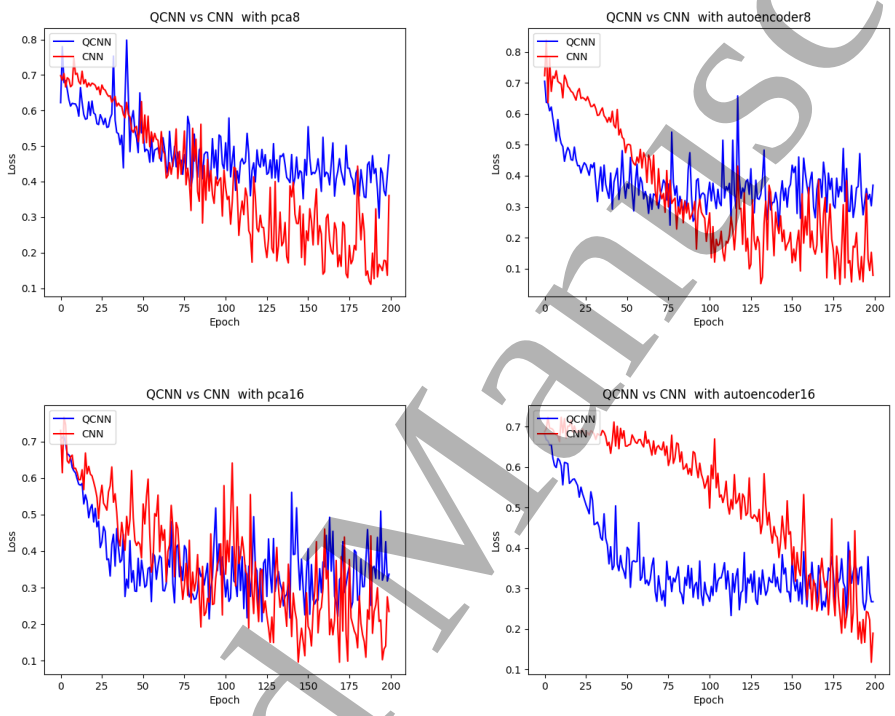


Figure 8: Training Comparison Between QCNN and CNN

For the second comparative experiment, we specifically focused on comparing the performance of HQCNN with that of a classic CNN. From the previous comparison in Section 3.3.2, we have already validated that HQCNN's classification performance is generally superior to that of traditional QCNN. In the earlier experiments, we established that the classification performance of traditional QCNN is better than that of a classic CNN. In this section of the study,our objective is to evaluate and compare the performance of HQCNN with that of the traditional CNN model.We altered the CNN design by integrating n classical fully connected layers in place of the quantum layer section of HQCNN.Next,we conducted training on the MNIST dataset for both HQCNN and the adjusted CNN. To directly verify the impact of the architecture on the model, this comparative experiment did not involve preprocessing of the data, simplifying the operational process.

The comparison results for training and classification are shown in Figure 10. From the training results in Figure 10(a), we can see that HQCNN and CNN have comparable convergence, with HQCNN exhibiting stronger learning capability compared to traditional QCNN models. Based

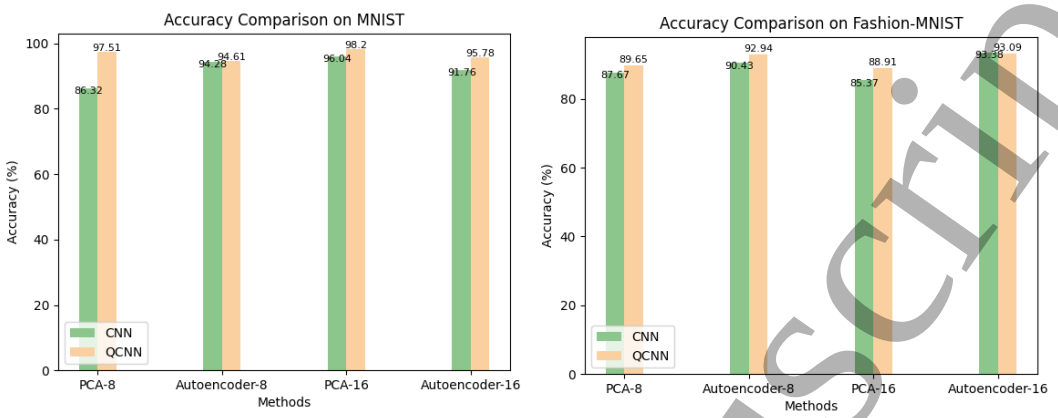
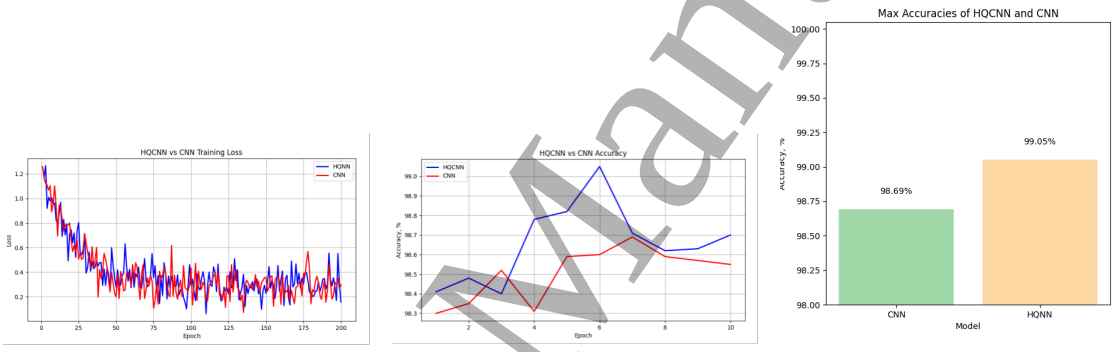


Figure 9: Classification Accuracy Comparison Between QCNN and CNN



(a) Training performance (b) Top 10 training runs (c) Highest training accuracy comparison

Figure 10: Testing and Training Results of HQCNN and CNN, where (a) is the training performance, (b) is the snapshot of the top 10 training runs with the highest classification accuracy, and (c) is the comparison of the highest training accuracy between HQCNN and CNN.

on Figures 10(b)and 10(c),it can be deduced that HQCNN attains a superior classification accuracy of 99.05% on the MNIST training set,surpassing the 98.69% achieved by the traditional CNN.Our experiments have demonstrated that the traditional CNN consists of 372,234 parameters,while HQCNN has 45,194 parameters.This implies that HQCNN outperforms the classic CNN in classification performance by utilizing just one-eighth of its parameters.This efficiency is due to the ability of quantum circuits to learn complex patterns more efficiently during non-linear mapping and feature extraction,thereby reducing the number of parameters required while maintaining model performance.In the hybrid model, the utilization of parallelism and quantum entanglement further reduces the parameter count needed to achieve the same level of task complexity.

4 Conclusion

With the advent of the big data era and the development of social information, the demand for image classification tasks has grown significantly across various fields, offering substantial prospects for development. Conventional machine learning algorithms encounter obstacles and constraints when handling extensive datasets. Leveraging the unique computational advantages, quantum computing emerges as a new approach to solving image classification tasks. This paper explores the integration of quantum computing and deep learning for investigating the utilization of Quantum Convolutional Neural Networks(QCNN)in image classification,introducing the Hybrid Quantum Convolutional Neural Network(HQCNN)and Hybrid Hierarchy Encoding(HHE). The research results show that HHE encoding not only inherits the advantages of traditional encoding methods, such as the high precision of angle encoding and the efficiency of amplitude encoding, but also overcomes their respective limitations, providing a new strategy for preparing quantum states.

This paper also introduces the Hybrid Quantum Convolutional Neural Network model (HQCNN), which integrates the advantages of classical convolution and quantum convolution. Through simulations of different QCNN models, we find that HQCNN not only exhibits higher classification accuracy when dealing with the MNIST dataset but also demonstrates better convergence compared to traditional QCNN circuits. Compared with traditional QCNN, HQCNN also shows strong classification performance and potential when handling more complex datasets like Fashion-MNIST. These experimental results prove that HQCNN not only possesses powerful classification capabilities but also maintains good generalization performance in complex tasks. Furthermore, this study presents comparative experiments with traditional CNNs, demonstrating that the classification performance of HQCNN on the MNIST dataset surpasses that of conventional classical CNN models while significantly reducing the number of parameters. Leveraging the nonlinear mapping and feature extraction capabilities inherent in quantum convolution alongside the data processing strengths of classical convolution, HQCNN is able to maintain or even exceed the performance levels of classical CNNs while simultaneously decreasing complexity. Although there is no substantial difference in convergence speed, HQCNN exhibits markedly higher classification accuracy compared to classical CNNs. This indicates that HQCNN not only inherits the data processing capabilities characteristic of classical CNNs but also harnesses the formidable computational power offered by quantum convolution. By integrating both classical and quantum methodologies, HQCNN enhances classification accuracy and achieves a significant breakthrough in parameter efficiency.

Our subsequent experiments revealed that HQCNN can effectively handle simple binary classification tasks and also has potential for multi-class classification tasks. Its classical architecture, after modification, remains adaptable to the requirements of multi-classification tasks. Future research efforts will continue to explore the application potential of HQCNN in more complex and diverse image classification tasks. In particular, whether HQCNN can effectively extend to multi-classification tasks will be a key focus. We expect to improve HQCNN in network architecture, such as modifying the structure of quantum layers or attempting changes in classical convolutional layers to adapt to multi-classification tasks, thus further enhancing its classification performance. With advancements in quantum computing technology and the maturation of quantum machine learning algorithms, HQCNN is expected to realize greater potential in the field of image classification.

References

[1] Meng-Hao Guo, Tian-Xing Xu, Jiang-Jiang Liu, Zheng-Ning Liu, Peng-Tao Jiang, Tai-Jiang Mu, Song-Hai Zhang, Ralph R Martin, Ming-Ming Cheng, and Shi-Min Hu. Attention mechanisms in computer vision: A survey. *Computational visual media*, 8(3):331–368, 2022.

[2] Yan-ling Tian, Wei-tong Zhang, Qie-shi Zhang, Gang Lu, and X Wu. Review on image scene classification technology [j]. *Acta Electronica Sinica*, 47(04):915–926, 2019.

[3] M Mitchell Waldrop. The chips are down for moore's law. *Nature News*, 530(7589):144, 2016.

[4] Sukhpal Singh Gill, Minxian Xu, Carlo Ottaviani, Panos Patros, Rami Bahsoon, Arash Shaghaghi, Muhammed Golec, Vlado Stankovski, Huaming Wu, Ajith Abraham, et al. Ai for next generation computing: Emerging trends and future directions. *Internet of Things*, 19:100514, 2022.

[5] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[6] Sukhpal Singh Gill, Adarsh Kumar, Harvinder Singh, Manmeet Singh, Kamalpreet Kaur, Muhammad Usman, and Rajkumar Buyya. Quantum computing: A taxonomy, systematic review and future directions. *Software: Practice and Experience*, 52(1):66–114, 2022.

[7] Charles Q Choi. Ibm's quantum leap: The company will take quantum tech past the 1,000-qubit mark in 2023. *IEEE Spectrum*, 60(1):46–47, 2023.

[8] Rainer Alt. On the potentials of quantum computing–an interview with heike riel from ibm research. *Electronic Markets*, 32(4):2537–2543, 2022.

[9] Frederic T Chong, Diana Franklin, and Margaret Martonosi. Programming languages and compiler design for realistic quantum hardware. *Nature*, 549(7671):180–187, 2017.

[10] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, 2017.

[11] Alok Anand, Meizhong Lyu, Prabh Simran Baweja, and Vinay Patil. Quantum image processing. *arXiv preprint arXiv:2203.01831*, 2022.

[12] Phuc Q Le, Fangyan Dong, and Kaoru Hirota. A flexible representation of quantum images for polynomial preparation, image compression, and processing operations. *Quantum Information Processing*, 10:63–84, 2011.

[13] Yi Zhang, Kai Lu, Yinghui Gao, and Mo Wang. Neqr: a novel enhanced quantum representation of digital images. *Quantum information processing*, 12:2833–2860, 2013.

[14] Glenn Beach, Chris Lomont, and Charles Cohen. Quantum image processing (quip). In *32nd Applied Imagery Pattern Recognition Workshop, 2003. Proceedings.*, pages 39–44. IEEE, 2003.

[15] Alexey Melnikov, Mohammad Kordzanganeh, Alexander Alodjants, and Ray-Kuang Lee. Quantum machine learning: from physics to software engineering. *Advances in Physics: X*, 8(1):2165452, 2023.

[16] Essam H Houssein, Zainab Abohashima, Mohamed Elhoseny, and Waleed M Mohamed. Machine learning in the quantum realm: The state-of-the-art, challenges, and future vision. *Expert Systems with Applications*, 194:116512, 2022.

[17] Maria Schuld and Nathan Killoran. Is quantum advantage the right goal for quantum machine learning? *Prx Quantum*, 3(3):030101, 2022.

[18] Kosuke Mitarai, Makoto Negoro, Masahiro Kitagawa, and Keisuke Fujii. Quantum circuit learning. *Physical Review A*, 98(3):032309, 2018.

[19] Tobias Haug, Kishor Bharti, and MS Kim. Capacity and quantum geometry of parametrized quantum circuits. *PRX Quantum*, 2(4):040309, 2021.

[20] Amine Zeguendry, Zahi Jarir, and Mohamed Quafafou. Quantum machine learning: A review and case studies. *Entropy*, 25(2):287, 2023.

[21] Bishrelt Khurelsukh. Hybrid quantum convolutional neural networks in tensorflow quantum. In *2022 33rd Irish Signals and Systems Conference (ISSC)*, pages 1–7. IEEE, 2022.

[22] Fangyu Huang, Xiaoqing Tan, Rui Huang, and Qingshan Xu. Variational convolutional neural networks classifiers. *Physica A: Statistical Mechanics and its Applications*, 605:128067, 2022.

[23] Iris Cong, Soonwon Choi, and Mikhail D Lukin. Quantum convolutional neural networks. *Nature Physics*, 15(12):1273–1278, 2019.

[24] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, 53(3):1–34, 2020.

[25] Marco Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, et al. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021.

[26] Ville Bergholm, Josh Izaac, Maria Schuld, Christian Gogolin, Shahnawaz Ahmed, Vishnu Ajith, M Sohaib Alam, Guillermo Alonso-Linaje, B AkashNarayanan, Ali Asadi, et al. Pennylane: Automatic differentiation of hybrid quantum-classical computations. *arXiv preprint arXiv:1811.04968*, 2018.

[27] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[28] Tara N Sainath, Brian Kingsbury, George Saon, Hagen Soltau, Abdel-rahman Mohamed, George Dahl, and Bhuvana Ramabhadran. Deep convolutional neural networks for large-scale speech tasks. *Neural networks*, 64:39–48, 2015.

[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[30] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al. Recent advances in convolutional neural networks. *Pattern recognition*, 77:354–377, 2018.

[31] Edward Grant, Marcello Benedetti, Shuxiang Cao, Andrew Hallam, Joshua Lockhart, Vid Stojevic, Andrew G Green, and Simone Severini. Hierarchical quantum classifiers. *npj Quantum Information*, 4(1):65, 2018.

[32] Arthur Pesah, Marco Cerezo, Samson Wang, Tyler Volkoff, Andrew T Sornborger, and Patrick J Coles. Absence of barren plateaus in quantum convolutional neural networks. *Physical Review X*, 11(4):041011, 2021.

[33] Mingxing Tan and Quoc V Le. Mixconv: Mixed depthwise convolutional kernels. *arXiv preprint arXiv:1907.09595*, 2019.

[34] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V Le. Mnasnet: Platform-aware neural architecture search for mobile. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2820–2828, 2019.

[35] Adriano Barenco, Charles H Bennett, Richard Cleve, David P DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John A Smolin, and Harald Weinfurter. Elementary gates for quantum computation. *Physical review A*, 52(5):3457, 1995.

[36] Su Yeon Chang, Michele Grossi, Bertrand Le Saux, and Sofia Vallecorsa. Approximately equivariant quantum neural network for p4m group symmetries in images. In *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, volume 1, pages 229–235. IEEE, 2023.

[37] Kyriaki A Tychola, Theofanis Kalampokas, and George A Papakostas. Quantum machine learning—an overview. *Electronics*, 12(11):2379, 2023.

[38] Chris Bernhardt. *Quantum computing for everyone*. Mit Press, 2019.

[39] Yanhu Chen, Cen Wang, Hongxiang Guo, Xiong Gao, and Jian Wu. Accelerating spiking neural networks using quantum algorithm with high success probability and high calculation accuracy. *Neurocomputing*, 493:435–444, 2022.

[40] Yunqian Wang, Yufeng Wang, Chao Chen, Runcai Jiang, and Wei Huang. Development of variational quantum deep neural networks for image recognition. *Neurocomputing*, 501:566–582, 2022.

[41] Tyler J Johnson, Stephen D Bartlett, and Barry C Sanders. Continuous-variable quantum teleportation of entanglement. *Physical Review A*, 66(4):042326, 2002.

[42] Y-Y Shi, L-M Duan, and Guifre Vidal. Classical simulation of quantum many-body systems with a tree tensor network. *Physical Review A—Atomic, Molecular, and Optical Physics*, 74(2):022320, 2006.

[43] P Bickel, P Diggle, S Fienberg, U Gather, I Olkin, and S Zeger. Springer series in statistics. *Principles and Theory for Data Mining and Machine Learning. Cham, Switzerland: Springer*, 2009.

[44] Andreas Kamilaris and Francesc X Prenafeta-Boldú. Deep learning in agriculture: A survey. *Computers and electronics in agriculture*, 147:70–90, 2018.

[45] Yurii Nesterov. A method for solving the convex programming problem with convergence rate o (1/k2). In *Dokl akad nauk Sssr*, volume 269, page 543, 1983.

[46] Guoming Chen, Qiang Chen, Shun Long, Weiheng Zhu, Zeduo Yuan, and Yilin Wu. Quantum convolutional neural network for image classification. *Pattern Analysis and Applications*, 26(2):655–667, 2023.