参赛队员姓名：黄博尧，王锴睿，高博言
中学:北京一零一中学
省份:北京市
国家:中国
指导教师姓名:李萌/蔡蕾
指导教师单位:中国科学院计算技术研究所/北京一零一中学
论文题目: MNIST Handwritten Digit Classification with Quantum Neural Network

MNIST Handwritten Digit Classification with Quantum Neural Network

Boyao Huang *
bhuang2024@gmail.com

Kairui Wang *
gerry.wang25@beijing101id.com

Boyan Gao *
13552007125@163.com

Abstract

Using quantum image representation algorithms as encoders for quantum neural networks(QNN), we explore the strengths and weaknesses of different ansatz and representations in terms of image classification. The designed quantum machine learning scheme is based on an implementable quantum circuit. In this article, various experiments are conducted to explore the factors that affect accuracy. The results derived from MNIST show that brick-pattern entangler offers the best reliability and accuracy among other entanglement structures. Regarding gray level alteration, an increase in gray level increases accuracy for complex problems but decreases for simple ones. For image encoding methods, FRQI demonstrated better stability and accuracy than NEQR. Readout from either position or color qubit would not cause a significant difference from one another. We also found that CR_y gates are volatile in our trials, and are potentially caused by incompatibility with FRQI and NEQR image representations.

Keywords: Quantum Neural Network, Quantum Image Representation, Handwritten Digital Classification, Loss Function

Contents

*Beijing 101 Middle School.

1 Introduction

In recent years, both applications and performances of quantum computers have improved greatly [HCS+20]with the development of theory and hardware[W+]. Compared with quantum computers, traditional computers are in service for a longer period and are generally better than quantum computers in terms of theoretical development, hardware manufacturing, and commercial applications[NC10]. However, quantum computers are expected to have a breakthrough in problems that are tough for conventional computers to solve and eventually surpass the latter in all aspects[Aha99].

In this study, we want to show the methodology of QNN[SSP14], and how it can improve the accuracy of handwritten digit classification. Our research demonstrates the influence of encoding methods, entanglement structures, and image characteristics on classification accuracy. We also examine the broader applications of quantum computing in various fields in the future.

Handwritten digit classification[LCJB+89] is a crucial issue in optical character recognition, and it has been employed as a case study to evaluate theories of pattern recognition and machine learning algorithms[Den12]. Traditional conventional neural networks(CNN) with training and improvement frameworks excel in this task [LJB+95]. Reference [LBD+89] reports a 1% error rate due to the highly specialized and constrained back-propagation network. However, CNN also has several state limits that have caused problems. These include high-performance hardware, lots of memory, and time efficiency measures. Meanwhile, quantum computing offers a new solution to these computational issues due to its characteristic of superposition and entanglement[LZX+20].

This research investigates the impact of QNN on the accuracy and computational efficiency of handwritten digit classification and presents a practical application of the QNN. Two quantum image representation methods are utilized in this study. According to [KJL+23], a study that focuses on quantum phenomena, research in quantum image processing uses a variety of quantum image representation formats. The Flexible Representation of Quantum Images (FRQI)[LDH11] method standardizes quantum computer images and encodes color and spatial information. FRQI can determine the correlation property of multipartite quantum pictures, as stated by [LDH11][LP23] in their studies. Another method, the Novel Enhanced Quantum Representation (NEQR)[ZLGW13], uses the basis state of a series of qubits to store the gray-scale value of individual pixels in an image instead of the probability amplitude used in FRQI. By encoding gray-scale information with more qubits, the NEQR quantum image model improves adaptability[ZLGW13].

Our study discusses the results of the experiments and determines the most successful combination of encoding methods, entanglement structures, and image features, to significantly enhance the accuracy of handwritten digit classification.

In this article, we focus on the advantages that quantum computing offers and study the MNIST handwritten digit classification task based on quantum neural networks. We discuss the progress of quantum computing and handwritten digital classification issues and provide quantum computing background knowledge. Further, we offer the basic methodology in the QNN and

present the outline and procedure of handwritten digit classification experiments. By evaluating the results of repeated experiments, the QNN approach's classification accuracy, worst-case and best-case scenarios for various entanglement structures, encoding methods, ansatzes, and quantum gates are present. Finally, we investigate how the study improves digital handwritten classification comprehension using QNN, evaluate its limitations, and encourage future quantum computing research. The contribution of this work is separated into five parts:

- The "brick-pattern" structure, which is positioned before the parameterized layer, demonstrates the highest performance.
- Entangler layer order rarely affects CX gate entanglement. The CX entangler after the entangler layer enhances the "color-target" structure. The entangler's position before CZ gates profoundly impacts "color-control" and "color-target" systems. After the parameterized layer, the CH entangler usually performs better.
- CR_y entangler results in slow and unstable convergence and overfitting in our model.
- In all cases, FRQI slightly outperforms NEQR.
- Readout from either the color qubit or the position qubit has little effect on QNN classification accuracy.

This paper is organized as follows. In section 2, we briefly outline the fundamental concepts of quantum gate and quantum measurement upon which our method will be based. Then, in section 3, We propose our quantum machine learning scheme for classifying handwritten digits, consisting of three essential procedures: encoding framework, ansatz design, measurement, and iteration. Next, in section 4, we demonstrate and compare the experiment results to illustrate the effects of different encoding methods, different ansatz entanglement structures, different digital pairs, and gray levels on classification. Finally, we end with a summary and outlook in section 5.

2 Preliminaries

2.1 Quantum gate

The basic unit of quantum computing is qubit, which can be represented as $|\psi\rangle = a|0\rangle + b|1\rangle$, where a and b are complex numbers which satisfy $|a|^2 + |b|^2 = 1$. Qubit operations are unitary, meaning they are reversible. The Pauli matrices can span a complex two-dimensional Hilbert space, which includes the 2×2 unitary matrices.

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -\mathrm{i} \\ \mathrm{i} & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

The single qubit operations also have rotation operators that rotate θ degrees around the $\hat{x}$, $\hat{y}$, $\hat{z}$ axis, which can be described as:

$$R_x(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & -\mathrm{i}\sin\frac{\theta}{2} \\ -\mathrm{i}\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}, \quad R_y(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}, \quad R_z(\theta) = \begin{pmatrix} e^{-\frac{\mathrm{i}\theta}{2}} & 0 \\ 0 & e^{\frac{\mathrm{i}\theta}{2}} \end{pmatrix}.$$

In fact, an arbitrary unitary operator can be demonstrated as a combination of rotation operators, which can be seen in the following theorem.

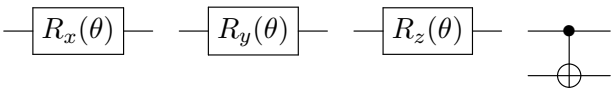
Theorem 1 (ZYZ decomposition [NC10]) *Suppose U is a unitary operation on a single qubit. Then there exist real numbers α, β, γ and δ such that $U = e^{i\alpha} R_z(\beta) R_y(\gamma) R_z(\delta)$.*

The prototypical controlled operation is the controlled-X, often referred to as CNOT, is a quantum gate with two input qubits, control and target qubits, respectively. The matrix of controlled-X (CX) operation is represented below.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

.

To simplify these operations in quantum circuits, we exhibit R_x, R_y, R_z, and CX gates as below.



2.2 Quantum measurement

Quantum measurement is usually placed at the end of the quantum circuit to extract and recover information from the results of quantum state evolution, which can be described by the measurement operator. For example, suppose the quantum state being measured is $|\psi\rangle = a|0\rangle + b|1\rangle$, the measurement operators we use are $M_0 = |0\rangle\langle 0|$ and $M_1 = |1\rangle\langle 1|$, thus the probability of obtaining measurement outcome 0 and 1 is $P(0) = \langle\psi|M_0^\dagger M_0|\psi\rangle = |a|^2$ and $P(1) = |b|^2$, respectively. So, the measurement result depends on the given probability distribution associated with the being measured quantum state. The final result can be calculated by the expected value of all the outcomes.

In this paper, we mainly use the projective measurement to extract information about the evolved system, An observable M, a Hermitian operator on the state space of the observed system, describes a projective measurement. The observable possesses a spectral decomposition, $M = \sum_m mP_m$, where P_m is the projector onto the eigenspace of M with eigenvalue m. For a given quantum state $|\psi\rangle$, the projective measurement average value is calculated:

$$\begin{aligned} E(M) &= \sum_m mp(m) \\ &= \sum_m m\langle\psi|P_m|\psi\rangle \\ &= \langle\psi|(\sum_m mP_m)|\psi\rangle \\ &= \langle\psi|M|\psi\rangle \\ &= \mathrm{Tr}(M|\psi\rangle\langle\psi|) \end{aligned}$$

3 Method

Quantum machine learning mainly includes several parts: image preprocessing, image encoding, ansatz design, measurement feedback, and iteration. First of all, the given classical data set is preprocessed to make its size and pixel information meet the requirements of our experiment, which can be done by using classical approaches and thus is not the focus of our discussion.

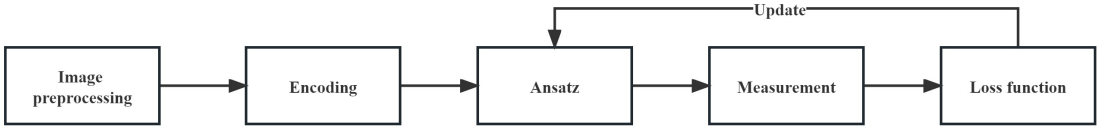


Figure 1: Overview

Then, through image encoding, the classical information in the image is encoded into quantum information, that is, the form of a quantum state. Next, the ansatz is designed to map the prepared quantum state. The ansatz layer is composed of parameterized quantum gates, which can be repeated several times. The number of repetitions should be adjusted according to the experiment. The parameter set of this layer would be iteratively adjusted according to ansatz's performance. Finally, the quantum state is measured and its result can be evaluated according to the loss function. The scheme would be iterated by updating the parameters until the model converges and achieves better performance. The overview of our design is illustrated in Figure 1. In the following subsections, we will discuss these essential parts individually.

3.1 Encoding

Quantum computing has strong parallelism due to the effects of quantum superposition, quantum entanglement, and quantum coherence, which often speed up classical computation. The image itself contains sufficient information, such as position coordinates, pixel intensity, curvature magnitude, geometric relationship, and so on. A well-designed encoder converts the classical information of the image into quantum information, which is stored in the form of quantum states. In recent years, various quantum image representation methods have been proposed, that combine quantum computation with digital image processing. Meanwhile, the number of qubits used in these quantum image representation formats, as well as the depth of the corresponding quantum circuit, are also in polynomial order, which can be used for NISQ(noisy intermediate-scale quantum computers) applications.

Le et al. put forward a flexible representation of quantum images (FRQI) [LDH11] in 2011, which captures position information and color information by using angular encoding. Later, the novel enhanced quantum representation (NEQR) of images was proposed in 2013 [ZLGW13], which encodes the color information in the basis states of a sequence of qubits instead of a single qubit in FRQI.

In our paper, we will adopt the two popular encoding schemes, FRQI and NEQR, to complete the task of MNIST handwritten digit classification.

3.1.1 FRQI

The Flexible representation of quantum images (FRQI) [LDH11] scheme maps the gray-scale value of each pixel to the amplitude while introducing a corresponding number of qubit sequences to denote the position coordinate of each pixel. Then, the whole image is prepared into a quantum superposition state as follows:

$$|I(\alpha)\rangle = \frac{1}{2^n}\sum_{i=0}^{2^{2n}-1}|c_i\rangle \otimes |i\rangle = \frac{1}{2^n}\sum_{i=0}^{2^{2n}-1}(\cos\alpha_i|0\rangle + \sin\alpha_i|1\rangle) \otimes |i\rangle,$$

where $\alpha_i \in [0, \frac{\pi}{2}]$, $i = 0, 1, \cdots, 2^{2n}-1$. $|c_i\rangle$ is an angular expression of pixel color and $|i\rangle$ is a binomial expansion of pixel location, respectively.

Given a $2^n \times 2^n$ image, it can be generated using only $2n+1$ qubits, where $2n$ qubits is needed for the position and one qubit for the color values. FRQI also guarantees that we can efficiently transform the initial state $|0\rangle^{\otimes 2n+1}$ into the target state using a polynomial number of quantum gates.

3.1.2 NEQR

Novel enhanced quantum representation (NEQR) [ZLGW13] scheme uses the basis state of a qubit sequence to store the gray-scale value of each pixel in the image. Thus, NEQR is useful for powerful intensity representation, which supports large images and reads intensity determinedly. It is an algorithm based on the Espresso heuristic logic minimizer [BHMSV12], providing a better logic gates compression rate in contrast to FRQI. NEQR also uses more qubits to encode quantum images. For an image with a size of $2^n \times 2^n$ and a gray-scale of 2^q (where q equals 8 in case of 256 intensity levels), the NEQR scheme uses a total of $2n+q$ qubits, of which $2n$ qubits are used to encode the position and q qubits are used to encode the color. The specific coding rules are as follows:

$$|I\rangle = \frac{1}{2^n} \sum_{X=0}^{2^n-1} \sum_{Y=0}^{2^n-1} \bigotimes_{i=0}^{q-1} |C_{XY}^i\rangle |XY\rangle,$$

where $|C_{XY}^i\rangle$ and $|XY\rangle$ are binomial expansions representing pixel intensity and position coordinate respectively.

3.2 Ansatz

An ansatz in quantum computation is a parameterized circuit. It is usually used in variational algorithms. A well-selected ansatz can improve the accuracy of computational results significantly. We selected a hardware-efficient ansatz that follows the "Single-qubit gate + Entangler + Single-qubit gate + Entangler ....." pattern [KMT+17]. The hardware-efficient ansatz, in contrast to the problem-inspired ansatz, is beneficial in the way that its qubit gates can be easily implemented in the aspect of hardware, and it requires less precision for the entangler. The design of different ansatz is the core of quantum machine learning, which uses parameterized quantum circuits to perform the unitary transformation on the encoded quantum states to realize quantum superposition and entanglement, maximize quantum parallelism, and extract the relevant information of images. Here, we employ some representative parameterized quantum circuits, including downstairs (Figure 2), full-connected (Figure 3), color-control/target (Figure 4), brick-pattern (Figure 5). In fact, all these parameterized quantum circuits are composed of two parts, namely, the unitary transformation layer and the entanglement layer. The unitary transformation layer is based on ZYZ decomposition (Theorem 1) and the entanglement layer is specially designed.

3.3 Measurement and iteration

After measurement, we can extract and determine the picture information, and then evaluate the gap between the output result and the picture label through the loss function. Then, after this evaluation, the feedback is fed into the parametric circuit to further update the parameters of the next iteration. The above steps are repeated until the model converges and better performance is achieved.

Loss function The loss function is a crucial part of the training process, which quantifies the difference between the measurement output and the real result. Many loss functions can be

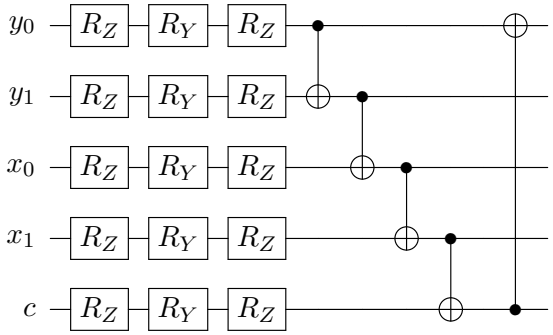


Figure 2: The downstairs structure. Each qubit in the entanglement layer will be connected to its adjacent one, and the last will be connected to the first. In the example,Ry and Rz are the parameter layers (CH, CRy, and CZ gates are also used for the entanglement layer in experiments).

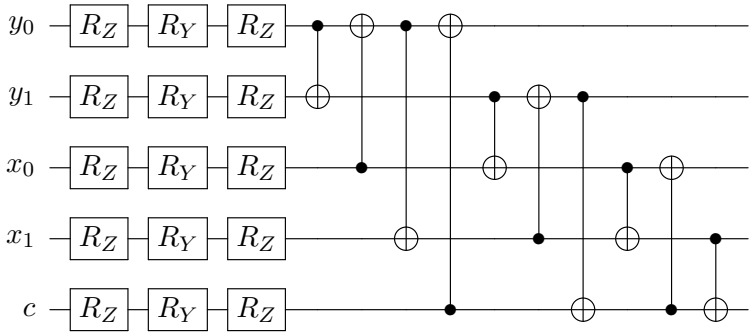


Figure 3: The full-connected structure. Every qubit is connected to all the other qubits, which forms the representation (CX gate here is one of the ways, in the experiment we also used such gates as CH, CRy and CZ).

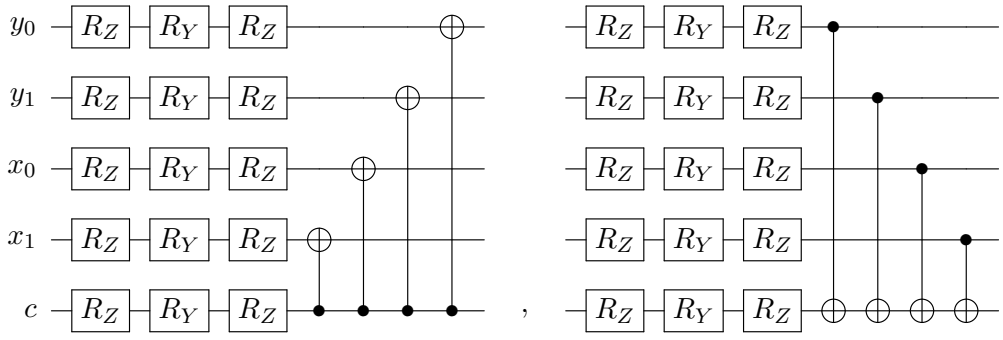


Figure 4: The color-control and color-target structure. All qubit connect to a color qubit(CX gate here is one of the ways, in the experiment we also used gates such as CH, CRy and CZ.).

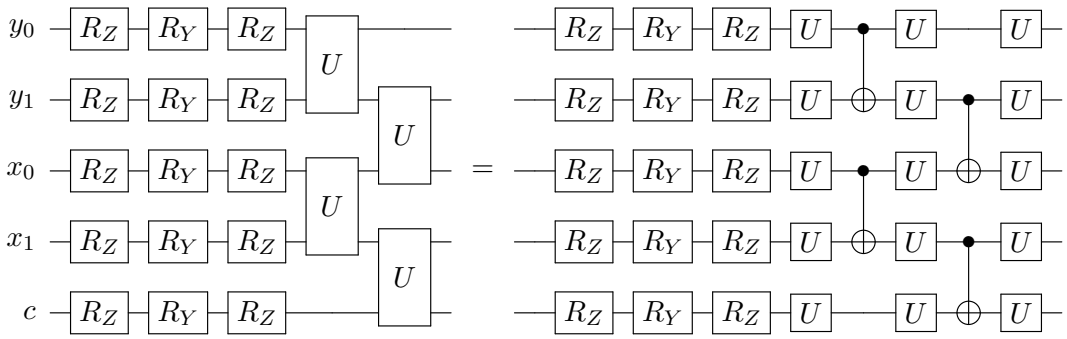


Figure 5: The brick-pattern structrs. The parameter layer is similar to the parameter layer in the previous structures. The entanglement layer consists of multiple brick-like units, as shown in the figure. Each brick consists of a control gate and two U-gates on each side.

chosen to execute the classification task, which is based on the optimization approach, including gradient descent, parameter shift, and so on. The two loss functions that are popular in QNN are Mean Squared Error (MSE) and Binary Cross Entropy (BCE, also known as log loss). Here we choose BCE to measure the distance between the generated label and the correct label. The equation of BCE is as follows:

$$\text{Binary Cross-Entropy}(y, \hat{y}) = -\left(y \cdot \log(\hat{y}) + (1-y) \cdot \log(1-\hat{y})\right),$$

where y is the true binary label (0 or 1), $\hat{y}$ is the predicted probability of the positive class. BCE is more fruitful in our case because it is specifically designed for binary classification problems, and that it is sensitive to probabilistic distribution.

4 Experiment result

We conducted a series of experiments to assess the influence of various factors on the classification performance of QNN. All experiments are performed using Python 3.9 and the quantum computing package QuICT [oQCTCS23] for noiseless simulation of quantum systems. For circuits with a width larger than 7, we employ RTX3090 for GPU simulation, while CPU simulation is used for the remaining cases.

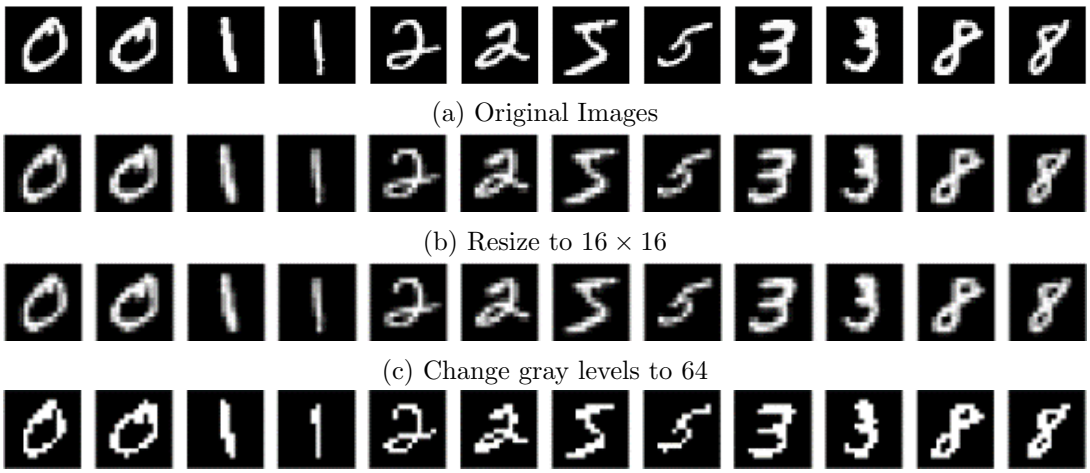
Dataset We utilize the MNIST handwriting dataset [Den12, XRV17], which is a widely used benchmark for image classification in machine learning. This dataset consists of 60,000 images of handwritten Arabic numerals ranging from 0 to 9. The task difficulty is moderate, and the image resolution is only 28×28 pixels, making it suitable for image encoding in current quantum systems. To ensure the credibility of our experiments, we select three pairs of numbers: 0 and 1 (low difficulty), 3 and 8 (high difficulty), and 2 and 5 (moderate difficulty). We use the torchvision library to load the dataset, with approximately 85% of the images used for training and the remaining 15% used for testing.

Image preprocessing In our work, image preprocessing consists of three main steps:

1. Downscale. The original image resolution of 28×28 is too large for the current quantum simulation. Additionally, both FRQI and NEQR only support images with resolutions of $2^n \times 2^n (n \in \mathbb{N})$. Therefore, we resize the images to 16×16 using bilinear interpolation. This resolution is clear enough to preserve image features effectively. The images after a downsampling operation are shown in Figure 6.

2. Change gray levels. The images in the MNIST dataset are grayscale images with 256 levels of gray. In some experimental settings, we reduce the number of gray levels by setting uniform intervals and limiting the gray values to the edges of these intervals.

3. Remove conflicts. After applying step (2), a small portion of the images may be labeled as belonging to both classes simultaneously. To ensure fairness in the experiments, we remove all such ambiguous instances.



(a) Original Images

(b) Resize to 16×16

(c) Change gray levels to 64

(d) Binary Images

Figure 6: Original images and preprocessed images of MNIST dataset.

Implementation details For the 4 entanglement structures, namely "downstairs," (Figure 2) "full-connected" (Figure 3), "color-control" (Figure 4 left), and "color-target" (Figure 4 right), we utilize four types of 2-qubit gates to achieve different entanglement effects in accordance with the requirements of the hardware-efficient ansatz. These gates include controlled X gates (CX), controlled Z gates (CZ), controlled H gates (CH), and controlled RY gates (CR_y). It is worth noting that CR_y gates are parameterized quantum gates that contain trainable parameters. As for the "brick-pattern" structure (Figure 5), since it uses both 1-qubit parameterized quantum gates and 2-qubit gates, we employ two sets of combinations: R_z and CX gates combined, and R_x and CZ gates combined.

Furthermore, we explore two different configurations for the entangler placement in relation to the parameterized layer, as shown in Figure 7. Some works such as [KMT+17, NPST22] position the entangler before the parameterized layer, while others, such as [BH23, DHL+21], place the entangler after the parameterized layer. We experiment with both cases to evaluate their impact on the classification performance of QNN.

Since the qubits used to encode position and color in both FRQI and NEQR have different states, we attempt to read out the first position qubit and the first color qubit separately.

Besides, to ensure relatively stable convergence, we set the total number of training epochs to 10. The batch size is set to 32, and the data order is shuffled and divided into batches before each epoch of training to prevent the model from learning the order of the data.

The ansatz depth, denoted as d, is set to 5 to ensure the expressiveness of the model. We calculate the binary cross-entropy loss of the classification results and update the parameters using the Adam optimizer with a learning rate of 0.001.

Evaluation criteria We select the highest test accuracy of a certain epoch after the convergence has become relatively stable as the final accuracy for an experiment. This approach

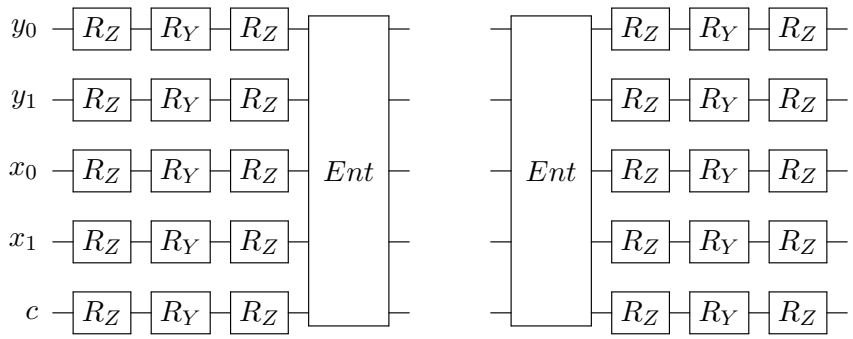


Figure 7: Two configurations for the entangler placement: after (left) or before (right) the parameterized layer.

helps to avoid accidental situations and ensures that the reported accuracy is a reliable measure of the model's performance.

4.1 Different entanglement structures

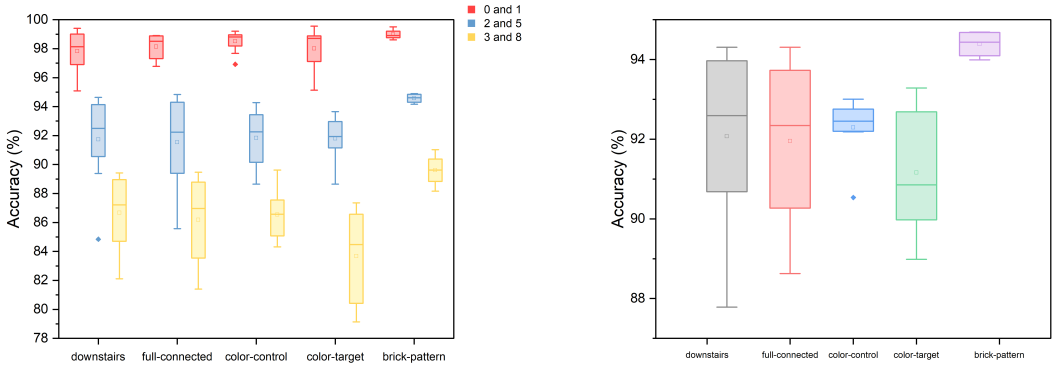
This set of experiments aims to compare the impact of different entanglement structures on the performance of QNN classification. In these experiments, we employ FRQI encoding on 16×16 binary images and uniformly read out from the unique color qubit. In addition to the "brick-pattern" structure, each experiment consists of 8 distinct entanglement structures, including 4 types of 2-qubit gate entanglers and 2 entangler positions. The experiments regarding the "brick-pattern" structure consist of 4 entanglement structures, including 2 types of gate combinations and 2 entangler positions. In order to effectively assess the influence of each entanglement structure on the classification performance of QNN, we will comprehensively compare their average accuracy, prediction stability, and performance for each entanglement structure when classifying the 3 difficulty classification problems.

Based on the experimental results presented in Figure 8, it is evident that the "brick-pattern" structure outperforms several other structures. Not only does it exhibit the highest average accuracy, but it also demonstrates superior performance in both the best and worst cases. In comparison to the other four structures, the "brick-pattern" structure is the most stable, as the difference between the best and worst cases is minimal. The prediction stability of the "color-control" structure is second only to the "brick-pattern" structure, but its prediction accuracy is less likely to be high. Additionally, it has a slight advantage when classifying simple problems. On the other hand, the "downstairs" and "full-connected" structures exhibit less stability, with high accuracy in favorable cases but generally poor accuracy in unfavorable cases. The "color-target" structure shows disadvantages in difficult classification problems.

4.2 Different entangler positions and entanglement effects of different gates

In this set of experiments, our objective is to assess the performance of different ansatz by employing various 2-qubit gates or gate combinations for entanglement. Additionally, we aim to investigate the influence of placing the entangler before or after the parameterized layer on the overall performance. We define the classification performance of the two sets of experiments as close if the difference in their accuracy is less than 1%. The experimental details remain the same as mentioned above.

We present the previous experimental results in an alternative format, which is displayed in Table 1. The blue background color indicates that placing the entangler after the parameterized layer yields better results compared to placing it before. Conversely, the yellow background



(a) (b)

Figure 8: (a) The performance of various entanglement structures on three groups of classification problems. (b) The overall average performance of these different entanglement structures.

color indicates that placing the entangler before the parameterized layer yields better results. The white background color indicates that the order in which the entangler is placed has little effect on the results (<1%). The entry highlighted in bold red represents the highest accuracy in classifying the corresponding pair of numbers, while the entry highlighted in bold green represents the lowest accuracy.

It can be observed that in most cases, the order in which the entangler and the parameterized layer are placed has little effect on the result when using CX gates for entanglement. However, for the "color-target" structure, placing the CX entangler after the parameterized layer generally shows better performance. On the other hand, for the "color-control" and "color-target" structures, the placement of the CZ entangler has a greater impact on the final result. It is usually better to place it before the parameterized layer than after. Conversely, the results of the CH entangler after the parameterized layer are usually better than those before. The convergence of the model tends to be unstable when using CR_y gates for entanglement. However, occasionally, when the CR_y entangler is placed before the parameterized layer, the results can be significantly better (accuracy increased by about 1% to 4.5%). Conversely, when the CR_y entangler is placed after, the results often do not show much improvement (accuracy increased by about 1% to 1.5%).

For the "brick-pattern" structure, the placement of the entangler has almost no impact on the results when faced with simple and medium-difficulty classification problems. However, for difficult problems, the entangler composed of the gate combination of R_x and CZ is better placed after the parameterized layer, while the R_z and CX gate combination is better placed before the parameterized layer.

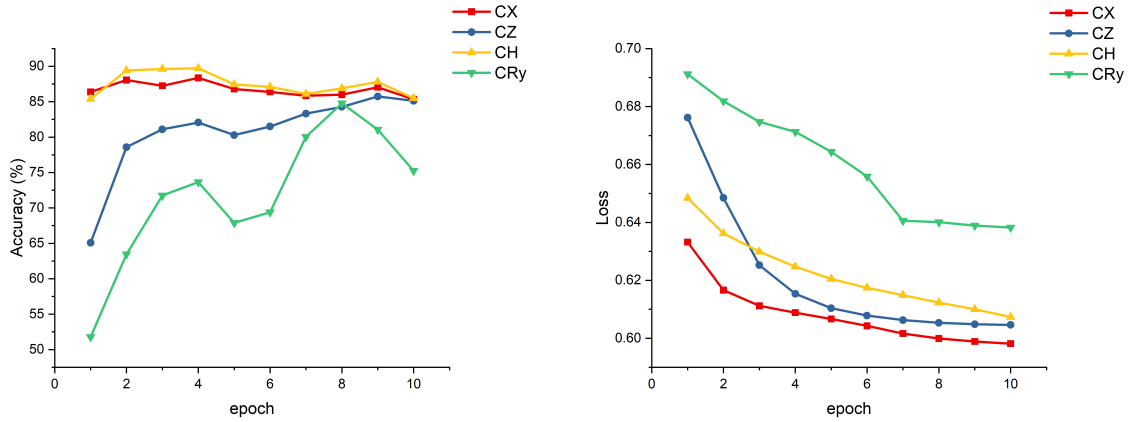
In summary, the "brick-pattern" structure, which uses an entangler combining R_z and CX gates and is placed before the parameterized layer, exhibits the best performance.

Entanglement with CR_y gates We have observed that the utilization of CR_y entanglers in our model leads to slow and unstable convergence, and is also prone to overfitting. To illustrate this, we randomly selected a set of experiments involving FRQI encoding on 16×16 binary images, using a "downstairs" entanglement structure, and reading out from the first position qubit to classify 3 and 8. We then plotted the validation loss and accuracy curves during the training process, where various types of 2-qubit gates were employed for entanglement, as shown in Figure 9. The figure clearly demonstrates that, in comparison to other 2-qubit gates,

	Entangler Structure	CX		CZ		CH		CR_y		R_x - CZ		R_z - CX	
		before	after	before	after	before	after	before	after	before	after	before	after
Average	downstairs	94.31	94.27	91.88	92.09	93.10	93.67	**87.78**	89.49				
	full-connected	93.69	93.65	90.04	91.04	93.76	94.31	88.63	90.50				
	color-control	92.60	92.47	92.91	90.55	92.44	93.00	92.19	92.21				
	color-target	91.17	92.47	92.91	90.55	89.97	93.29	88.98	89.98				
	brick-pattern									93.99	94.20	**94.69**	94.67
1 and 0	downstairs	99.40	99.06	96.83	98.56	97.72	98.96	**95.09**	96.97				
	full-connected	98.86	98.41	97.02	96.78	98.91	98.61	98.91	97.62				
	color-control	99.21	98.81	98.81	96.92	98.86	97.67	99.06	98.71				
	color-target	97.32	**99.55**	98.81	96.92	98.91	98.86	95.14	98.61				
	brick-pattern									98.91	98.91	98.61	99.50
2 and 5	downstairs	94.11	94.64	91.77	91.72	94.17	93.23	**84.84**	89.38				
	full-connected	94.27	94.32	91.46	90.36	93.02	94.84	85.57	88.44				
	color-control	93.91	94.27	92.81	88.65	92.97	91.72	90.05	90.26				
	color-target	91.98	93.13	92.81	88.65	91.88	93.65	91.67	90.63				
	brick-pattern									**94.90**	94.17	94.43	94.79
3 and 8	downstairs	89.42	89.11	87.04	85.99	87.40	88.81	83.42	82.11				
	full-connected	87.95	88.21	81.65	85.99	89.36	89.47	81.40	85.43				
	color-control	84.68	84.32	87.10	86.04	85.48	89.62	87.45	87.65				
	color-target	84.22	84.73	87.10	86.04	**79.13**	87.35	80.14	80.70				
	brick-pattern									88.16	89.52	**91.03**	89.72

Table 1: Accuracy of QNN using various structures of hardware-efficient ansatz.

the model utilizing CR_y entanglers exhibits minimal convergence in the initial epochs and significant fluctuations in the final epochs. Conversely, models employing alternative entanglers have already achieved stable convergence.



(a) Validation accuracy

(b) Validation loss

Figure 9: QNN training curve.

It is important to note that this slow convergence and overfitting phenomenon is not solely attributed to an increase in the number of parameters. As Table 2 shows, when employing the "brick-pattern" structure, which also involves a substantial number of trainable parameters, the model does not exhibit similar issues as observed with the CR_y entangler. Based on this observation, we infer that the entanglement provided by CR_y gates may not be conducive to effective classification for QNN when utilizing FRQI and NEQR encoding methods.

Entangler Structure	CX, CZ, CH	CR_y	R_x - CZ, R_z - CX
downstairs	135	180	
full-connected	135	315	
color-control	135	175	
color-target	135	175	
brick-pattern			270

Table 2: The number of training parameters required for each entanglement structure when using FRQI encoding.

4.3 Different encoding methods and gray levels

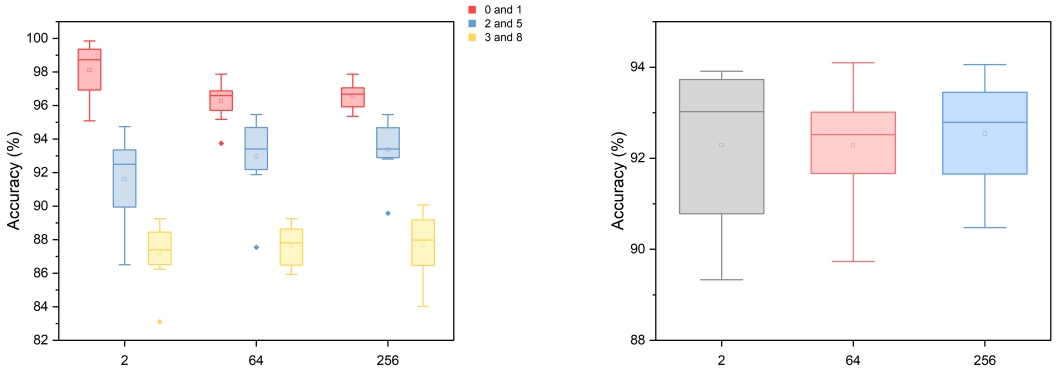
To the human eye, images with a higher number of gray levels are perceived to contain more information and features. Therefore, the objective of this set of experiments is to investigate whether classical image encoding methods based on quantum systems can effectively encode different levels of grayscale information that can be captured by QNN and utilized as classification features. By conducting these experiments, we can gain valuable insights into the potential of quantum image encoding for enhancing the representation of grayscale information and its applicability in QNN-based classification tasks.

The experiments are divided into two parts. In the first part, we utilize FRQI to encode 16×16 binary images, 64-level grayscale images, and 256-level grayscale images. And we employ 8 different "full-connected" structure entanglers. In the second part, we separately employ FRQI and NEQR to encode images with these three gray levels. Among the 8 types of "downstairs" entanglers, we select 2 with better performance, namely entangling with CX gates and CH gates. These 2 entanglers are placed after the parameterized layer. In both parts of the experiments, the resulting data is read out from the first position qubit.

From Figure 10, it is evident that increasing the number of gray levels can enhance the average accuracy and stability of QNN classification, particularly for medium-difficulty classification problems. However, for simple classification problems, the additional image information provided by increasing the gray levels may disrupt the original "concise" features and confuse the model, leading to a decrease in classification performance. Similarly, for difficult classification problems, a significant increase in the number of gray levels can also reduce the stability of model predictions. Table 3 demonstrates that increasing the number of gray levels does indeed provide more information, resulting in a slight improvement in classification accuracy for both FRQI and NEQR. Additionally, it is evident that encoding with FRQI outperforms NEQR in all scenarios, although the difference between the two methods is not substantial. However, as the gray level increases, NEQR significantly expands the circuit width and the number of training parameters. This, in turn, requires more computing resources and prolongs the training time, which may not be worth the marginal gain achieved.

Gray-levels	Encoding	0 and 1	2 and 5	3 and 8
64	FRQI	**96.78**	**95.73**	**89.16**
	NEQR	94.79	94.43	86.34
256	FRQI	**96.78**	**95.78**	**89.26**
	NEQR	94.89	95.26	87.85

Table 3: Accuracy of FRQI and NEQR at different gray levels.



(a) (b)

Figure 10: (a) The performance of various gray levels on three groups of classification problems. (b) The overall average performance of these different gray levels.

4.4 Readout from position qubit and color qubit

When utilizing FRQI and NEQR for image encoding, the qubits are categorized into position qubits and color qubits. These two types of qubits hold distinct meanings and statuses. To examine the potential impact of the readout qubit on the classification outcome of QNN, we conducted this experiment. First, we employed FRQI encoding on 16×16 binary images (FRQI and NEQR produce identical outcomes when encoding binary images). Additionally, we utilized two types of entanglers: the "downstairs" structure and the "full-connected" structure, each consisting of 8 variations. Subsequently, we separately readout from the first position qubit and the sole color qubit, respectively, and calculated the average accuracy.

From Table 4, it can be observed that the readout from both the position qubit and the color qubit has a minimal impact on the classification accuracy of QNN (<1%).

	Entangler structure	position qubit	color qubit
0 and 1	downstairs	97.90	97.82
	full-connected	98.12	98.14
2 and 5	downstairs	92.25	91.73
	full-connected	91.61	91.54
3 and 8	downstairs	85.97	86.66
	full-connected	87.14	86.18

Table 4: Accuracy of readouts from different qubits.

5 Summary

In this paper, we experimented with QNN handwritten digit recognition based on the MNIST handwritten digit database. Our results revealed that the brick-patterned structure is the best-performance entanglement structure among the five structures. Besides, CR_y gates are the most unsuitable and simply over-fitting. In addition, when evaluating the grayscale values of an image, an increase in the range of gray values can reduce the simplicity, accuracy, and stability of the image. At the same time, it will improve the performance of moderately complex images

but reduce the stability of highly complex images. Among the representation methods, FRQI is better than NEQR, as the former demonstrated unparalleled efficiency; NEQR operates with a wider circuit width and is poor in executing simple tasks. Lastly, accuracy is not greatly affected by readout from color qubit or position qubit. In the future, our work can be extended to designing more efficient encoding schemes, creating the higher performance ansatz, exploiting QNN application in multi-classification, and investigating the performance of the hybrid quantum-classical neural network to explore the potential of quantum neural network and quantum computation.

References

[Aha99] Dorit Aharonov. Quantum computation. *Annual Reviews of Computational Physics VI*, pages 259–346, 1999.

[BH23] Qi Bai and Xianliang Hu. Quantity study on a novel quantum neural network with alternately controlled gates for binary image classification. *Quantum Information Processing*, 22(5):184, Apr 2023.

[BHMSV12] R.K. Brayton, G.D. Hachtel, C. McMullen, and A.L. Sangiovanni-Vincentelli. *Logic Minimization Algorithms for VLSI Synthesis*. The Springer International Series in Engineering and Computer Science. Springer US, 2012.

[Den12] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[DHL+21] Yuxuan Du, Min-Hsiu Hsieh, Tongliang Liu, Shan You, and Dacheng Tao. Learnability of quantum neural networks. *PRX Quantum*, 2:040337, Nov 2021.

[HCS+20] Vikas Hassija, Vinay Chamola, Vikas Saxena, Vaibhav Chanana, Prakhar Parashari, Shahid Mumtaz, and Mohsen Guizani. Present landscape of quantum computing. *IET Quantum Communication*, 1(2):42–48, 2020.

[KJL+23] Hyunji Kim, Kyungbae Jang, Sejin Lim, Yeajun Kang, Wonwoong Kim, and Hwajeong Seo. Quantum neural network based distinguisher on speck-32/64. *Sensors*, 23(12):5683, 2023.

[KMT+17] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242–246, Sep 2017.

[LBD+89] Yann LeCun, Bernhard Boser, John Denker, Donnie Henderson, Richard Howard, Wayne Hubbard, and Lawrence Jackel. Handwritten digit recognition with a backpropagation network. *Advances in neural information processing systems*, 2, 1989.

[LCJB+89] Yann Le Cun, Lawrence D Jackel, Brian Boser, John S Denker, Hans Peter Graf, Isabelle Guyon, Don Henderson, Richard E Howard, and William Hubbard. Handwritten digit recognition: Applications of neural network chips and automatic learning. *IEEE Communications Magazine*, 27(11):41–46, 1989.

[LDH11] 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.

[LJB+95] Yann LeCun, Larry Jackel, Leon Bottou, A Brunot, Corinna Cortes, John Denker, Harris Drucker, Isabelle Guyon, UA Muller, Eduard Sackinger, et al. Comparison of learning algorithms for handwritten digit recognition. In *International conference on artificial neural networks*, volume 60, pages 53–60. Perth, Australia, 1995.

[LP23] Marina Lisnichenko and Stanislav Protasov. Quantum image representation: A review. *Quantum Machine Intelligence*, 5(1):2, 2023.

[LZX+20] YaoChong Li, Ri-Gui Zhou, RuQing Xu, Jia Luo, and WenWen Hu. A quantum deep convolutional neural network for image recognition. *Quantum Science and Technology*, 5(4):044003, 2020.

[NC10] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

[NPST22] T. Nguyen, I. Paik, H. Sagawa, and T. C. Thang. Quantum machine learning with quantum image representations. In *2022 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 2022 IEEE International Conference on Quantum Computing and Engineering (QCE), pages 851–854, 2022.

[oQCTCS23] Library of Quantum Computation and CAS Theoretical Computer Science, ICT. Quict, 2023.

[SSP14] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. The quest for a quantum neural network. *Quantum Information Processing*, 13:2567–2586, 2014.

[W+] Colin P Williams et al. *Explorations in quantum computing*. Springer.

[XRV17] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[ZLGW13] 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.

致谢

近年来，量子计算领域取得了迅猛的发展，吸引了大量学者进行深入研究。本文的重点是探讨神经网络的基本任务——数字分类，并综合考量编码，拟设，数字对量子神经网络的性能影响并运用于实际任务之中。行文至此，我们特别感谢聂均鸿、郭子瑛、李萌三位老师，在理论指导、实验指导和论文指导方面的倾力指导与协助。本文的作者包括高博言、黄博尧和王锴睿。王锴睿负责第一部分的引言，第二部分预备知识的介绍，以及实验数据的收集。黄博尧在本文中负责第三部分介绍了本文的方法，以及实验数据的收集。高博言承担了文章的引言部分和第四部分实验结果的呈现，同样也参与了实验数据的收集。