QGAN-based data augmentation for hybrid quantum-classical neural networks

Run-Ze He[a,b,c], Jun-Jian Su[a], Su-Juan Qin[a,*], Zheng-Ping Jin[a,*], Fei Gao[a]

[a]*Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China*
[b]*National Engineering Research Center of Disaster Backup and Recovery, Beijing University of Posts and Telecommunications, Beijing, 100876, China*
[c]*School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing University of Posts and Telecommunications, Beijing, 100876, China*

Abstract

Quantum neural networks converge faster and achieve higher accuracy than classical models. However, data augmentation in quantum machine learning remains underexplored. To tackle data scarcity, we integrate quantum generative adversarial networks (QGANs) with hybrid quantum-classical neural networks (HQCNNs) to develop an augmentation framework. We propose two strategies: a general approach to enhance data processing and classification across HQCNNs, and a customized strategy that dynamically generates samples tailored to the HQCNN's performance on specific data categories, improving its ability to learn from complex datasets. Simulation experiments on the MNIST dataset demonstrate that QGAN outperforms traditional data augmentation methods and classical GANs. Compared to baseline DCGAN, QGAN achieves comparable performance with half the parameters, balancing efficiency and effectiveness. This suggests that QGANs can simplify models and generate high-quality data, enhancing HQCNN accuracy and performance. These findings pave the way for applying quantum data augmentation techniques in machine learning.

*Corresponding author, E-mail:qsujuan@bupt.edu.cn; zhpjin@bupt.edu.cn

1. Introduction

Quantum computing revolutionizes computational science by using quantum principles such as superposition, entanglement, and interference to solve problems beyond classical systems [1]. This has given rise to Quantum Machine Learning (QML), which combines quantum computing with artificial intelligence to enhance learning algorithm efficiency [2]. QML leverages quantum properties to handle high-dimensional data and complex tasks, offering advantages over classical machine learning in areas like optimization, pattern recognition, and quantum chemistry [3–5]. As quantum hardware advances, QML is emerging as a key driver of computational innovation across various fields.

Hybrid quantum-classical neural networks (HQCNNs) integrate the computational efficiency of classical neural networks with the unique capabilities of quantum circuits, such as superposition and entanglement, offering promise in quantum machine learning. Classical components perform feature extraction and preprocessing, preparing data for quantum processing. Quantum circuits use superposition to process multiple states simultaneously and entanglement to capture complex data correlations, improving efficiency. This hybrid architecture outperforms traditional neural networks in tasks such as image classification, financial forecasting, and drug design [6–11]. However, data scarcity remains a major challenge for HQCNNs. Like other models, HQCNNs require large, high-quality datasets for robust generalization and to avoid overfitting. Data acquisition in quantum computing faces challenges: preparing quantum states requires advanced equipment and time, while noisy intermediate-scale quantum (NISQ) hardware has limitations such as few qubits and high error rates [12, 13]. Classical data augmentation methods, like geometric transformations, are less effective for HQCNNs due to differences in quantum data processing [14, 15]. Therefore, developing customized data augmentation strategies for HQCNNs is a critical research priority.

Quantum generative adversarial networks (QGANs) offer a promising solution to the data scarcity issue in quantum machine learning (QML). By adapting the generative adversarial network (GAN) framework to the quantum domain, QGANs use parameterized quantum circuits to generate synthetic quantum states or classical data distributions [16]. Theoretical studies show that QGANs, utilizing quantum parallelism and superposition, can achieve exponential speedups in specific tasks [17, 18]. Research confirms

QGANs' ability to model complex quantum distributions, highlighting the advantages of quantum superposition and entanglement in generative tasks [19–21]. QGAN research spans several areas: at the data processing level, QGANs efficiently learn both classical and quantum data [22–24]; in algorithmic optimization, quantum entanglement helps address challenges like non-convex optimization and mode collapse seen in classical GANs [25]; and in applications, QGANs show potential in fields like small-molecule drug discovery [26] and anomaly detection [27]. Unlike classical GANs, which struggle with quantum data distributions, QGANs are naturally compatible with quantum environments, making them ideal for augmenting datasets for hybrid quantum-classical neural networks (HQCNNs) [28]. Despite significant theoretical advancements and proof-of-concept validations, practical QGAN applications for HQCNN data augmentation remain underdeveloped. Existing studies primarily focus on theoretical model development, with limited progress in scalable, engineering-oriented solutions for real-world applications. Bridging the gap between the theoretical promise of QGANs and their practical implementation as data augmentation strategies remains a critical challenge in quantum machine learning.

This study presents a framework that integrates hybrid quantum-classical neural networks (HQCNNs) with quantum generative adversarial networks (QGANs) to address the data scarcity issue in quantum machine learning. The research designs two augmentation strategies: a general strategy applicable to various HQCNN architectures for performance optimization, and a customized strategy that dynamically generates samples based on the model's performance on specific data categories, enhancing robustness on challenging classes. Simulation experiments on the MNIST dataset compare QGAN's effectiveness with traditional data augmentation methods and classical GANs, confirming the framework's efficacy. Results show QGAN achieves performance similar to deep convolutional GANs (DCGANs) with fewer parameters, highlighting quantum computing's efficiency in generative modeling. These findings provide a new approach to overcoming limitations of conventional data augmentation and lay the foundation for practical quantum data augmentation in real-world applications.

The rest of the framework of this paper is as follows: In Sect. 2, we provide a comprehensive review of the theoretical foundations of qubits, quantum circuits, variational quantum circuits, and QGANs. In Sect. 3, we elaborate on the proposed architecture design of HQCNN and QGAN, alongside detailed technical descriptions of the two data augmentation strategies. In Sect. 4,

we present the experimental results, comparing the proposed approach with classical benchmark models and providing a thorough analysis. In Sect. 5, we conclude the study, summarizing key findings, discussing practical implications, and outlining directions for future research.

2. Preliminary

2.1. Qubits, Quantum Gates, Quantum Circuits and Measurements

Unlike classical bits, which are limited to 0 or 1, quantum bits (qubits) can exist in a superposition state of $|0\rangle$ and $|1\rangle$. Their state can be described as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where α, β are complex coefficients satisfying $|\alpha|^2 + |\beta|^2 = 1$. This superposition enables quantum computing to process multiple states simultaneously.

Quantum gates, the fundamental units of quantum computation, operate on qubits in superposition, analogous to classical logic gates. Single quantum bit gates, including the I, Pauli-X, Pauli-Y, Pauli-Z, and Hadamard gates, transform the state of a single qubit, as shown in Eq. 1. Phase (rotation) gates are another common class of quantum gates. They keep $|0\rangle$ unchanged, but rotate $|1\rangle$ around the Z-axis by a specified angle θ. Commonly used rotation gates in quantum computing and quantum machine learning are mainly R_x, R_y, R_z gates, and the matrix form is shown in Eq. 2. Multi-qubit gates, like the controlled-NOT (CNOT) gate, create entanglement and enable multi-qubit operations, with their matrix form given in Eq. 3. The combination of these quantum gates constitutes a quantum circuit, and various complex quantum algorithms can be implemented by designing specific gate sequences.

Measurement is a critical step in quantum computing for extracting results. Upon measurement, a qubit's superposition collapses to $|0\rangle$ or $|1\rangle$ state with the probability of $|\alpha|^2$ and $|\beta|^2$, respectively, irreversibly altering the qubit's original state—a fundamental feature of quantum mechanics [29]. In entangled systems, measuring one qubit instantly influences others—a principle exploited in protocols like quantum key distribution [30] and quantum teleportation [31]. Measurement not only converts quantum states into classical information but also bridges quantum and classical computation.

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$
$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{1}$$
$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$R_\theta = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}, R_x = \begin{bmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix},$$
$$R_y = \begin{bmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix}, R_z = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}. \tag{2}$$

$$\text{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{3}$$

2.2. Variational Quantum Circuits(VQCs) and HQCNNs

Variational quantum circuits (VQCs), also known as parameterized quantum circuits (PQCs), comprise gates with tunable parameters. These gates are configured in specific combinations tailored to the task and available quantum resources. The parameter optimization process integrates both classical and quantum systems for effective training. These circuits are remarkably resistant to noise, making them ideal for noisy intermediate-scale quantum (NISQ) devices.

The workflow of the hybrid quantum-classical neural networks (HQC-NNs) reflects the close collaboration between classical and quantum. First, the classical data $\boldsymbol{x} \in \mathbb{R}^d$ is subjected to preliminary feature extraction and processing by the classical neural network module to extract more representative features. Then, these features are mapped to quantum states $\rho(\boldsymbol{x})$ through a specific encoding method, which is achieved by using the quantum circuit $U(\boldsymbol{x})$ to act on the initial state ρ_0. Subsequently, a parameterized quantum circuit $V(\boldsymbol{\theta})$ with trainable parameters acts on $\rho(\boldsymbol{x})$ to extract deep features using the superposition and entanglement properties of quantum states, this process combines the powerful parallel computing capabilities of

quantum systems and the flexible optimization and adjustment capabilities of classical systems. After that, the resulting state $\rho(\boldsymbol{x}) = U(\boldsymbol{x})\rho_0 U^\dagger(\boldsymbol{x})$ is measured for a carefully selected observable O to produce an expected value: $E(\boldsymbol{\theta}) = \mathrm{Tr}\left[V(\boldsymbol{\theta})\rho(\boldsymbol{x})V^\dagger(\boldsymbol{\theta})O\right]$. This expected value $E(\boldsymbol{\theta})$ is used to calculate the loss function $L(\boldsymbol{\theta})$, which measures the difference between the HQCNNs predicted output and the expected target. To optimize the model, the parameter $\boldsymbol{\theta}$ is iteratively adjusted on a classical computer according to the loss function to ensure seamless interaction between quantum computing and classical optimization, thereby continuously improving the performance of the model. The complete schematic diagram is shown in Fig. 1.

Numerous experiments show that hybrid variational quantum circuits combined with deep neural networks (DNNs) outperform traditional DNNs in efficiency and convergence [32, 33]. Recent studies further show that quantum circuits achieve faster learning and higher accuracy in applications such as quantum convolutional networks [34], quantum LSTM networks [35], and natural language processing [36], underscoring their potential in advancing machine learning.
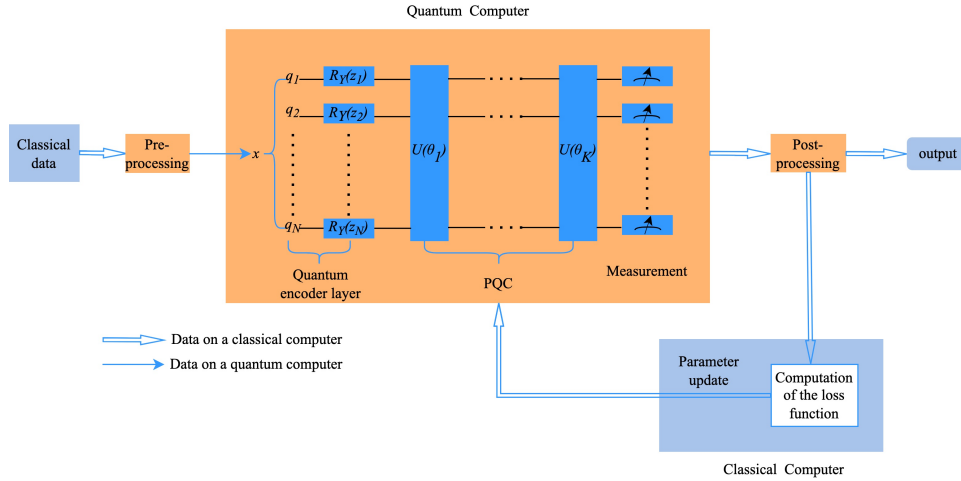

Figure 1: Flowchart of Hybrid Quantum-Classical Neural Networks Framework

2.3. Quantum Generative Adversarial Networks(QGANs)

Generative adversarial networks (GANs), a breakthrough in unsupervised learning, operate based on the principle of adversarial training. GANs perform well in image generation [37], video synthesis, speech generation [38],

and text generation [39], with broad applications. GANs were first proposed by Goodfellow et al. in 2014 [40]. A GAN consists of two key components: a generator (G) and a discriminator (D). The generator takes random samples from a latent space to produce outputs resembling real training data, while the discriminator learns to distinguish real from generated samples. Through adversarial training, the generator learns to fool the discriminator, and both networks iteratively adjust parameters until the discriminator assigns equal (50%) probability to real and generated samples, indicating a Nash equilibrium. Quantum GANs (QGANs) also comprise a generator and a discriminator, but the generator uses parameterized quantum circuits instead of classical networks. The core adversarial mechanism remains unchanged. The loss function for training QGANs is defined in Eq. 4. Training alternates between optimizing the generator's parameters λ and the discriminator's parameters μ, similar to classical GANs. Quantum generator parameters are optimized using automatic differentiation [41]. When training the discriminator, λ is fixed and μ is updated using backpropagation. When training the generator, μ is fixed and gradients are computed using the parameter shift rule [42]. The training process is illustrated in Algorithm 1.

$$\min_{G} \max_{D} V(D, G) = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))] \quad (4)$$

3. Methodology

In this study, we deeply explore the hybrid quantum-classical neural network (HQCNN) and quantum generative adversarial network (QGAN), aiming to tap the huge potential of quantum mechanics and quantum machine learning in data augmentation, and focus on the impact of different data augmentation strategies on HQCNN model optimization. The study consists of two parts. The first proposes a general data augmentation strategy that is broadly applicable and compatible with various HQCNN architectures. It enhances data processing and classification performance through standardized procedures; The second part introduces a customized augmentation scheme tailored to a specific HQCNN. This scheme analyzes the performance of the model in different categories and constructs a dynamic sample generation mechanism to address overfitting in data-scarce settings and improve learning on complex distributions. The study highlights the unique advantages of

7

Algorithm 1 Quantum Generative Adversarial Network Algorithm

Input: Generator G: qubit N, Depth d, Trainable parameters λ; Discriminator D: Classical fully-connected neural network, Trainable parameters μ; Learning rate η, Training set T

Output: Parameters λ of the converged quantum generator G_λ and parameters μ of the classical discriminator D_μ

 1: **while** λ, μ have not converged **do**
 2:     Train the discriminator D
 3:     Sample a data point $x \sim \kappa$ from the training set T
 4:     Generate a sample $x_G \sim p_\lambda = |\langle x|\psi_\lambda\rangle|^2$ from the quantum circuit.
 5:     Compute $\nabla_\mu \ell(D_\mu)$ using the backpropagation algorithm
 6:     Update μ: $\mu \leftarrow \mu - \eta\nabla_\mu \ell(D_\mu)$
 7:     Train the generator G
 8:     Adjust the parameters λ of the quantum generator to $\lambda + \pi/2$, and sample $x \sim$ $p_{\lambda+\pi/2}(x) = |\langle x|\psi_{\lambda+\pi/2}\rangle|^2$
 9:     Adjust the parameters λ of the quantum generator to $\lambda - \pi/2$, and sample $x \sim$ $p_{\lambda-\pi/2}(x) = |\langle x|\psi_{\lambda-\pi/2}\rangle|^2$
10:     Compute $\nabla_\lambda \ell(G_\lambda)$
11:     Update λ: $\lambda \leftarrow \lambda - \eta\nabla_\lambda \ell(G_\lambda)$
12: **end while**
13: **Return** Parameters λ of the converged quantum generator G_λ and parameters μ of the classical discriminator D_μ

---

quantum computing in data augmentation and offers innovative approaches for enhancing quantum machine learning performance in practical applications.

3.1. Model Architecture of HQCNN

The proposed HQCNN adopts a cascaded architecture combining a classical convolutional neural network (CNN) with a variational quantum circuit (VQC). The CNN module includes a 3×3 convolutional layer, a max-pooling layer, and a ReLU activation. This module effectively extracts local features and reduces the dimensionality of the feature map. The ReLU introduces nonlinearity to enhance model expressiveness. The extracted feature vector is then encoded into a quantum state using angle encoding. This method enables efficient conversion of classical features into quantum states for further quantum processing. The encoded quantum state is fed into a VQC composed of R_y, R_z rotation gates, and CNOT gates. The full HQCNN architecture is illustrated in Fig. 2. This design combines the efficiency of classical CNNs in feature extraction and reduction with the strengths of

quantum circuits in deep feature processing and parallelism, offering strong support for classification tasks.
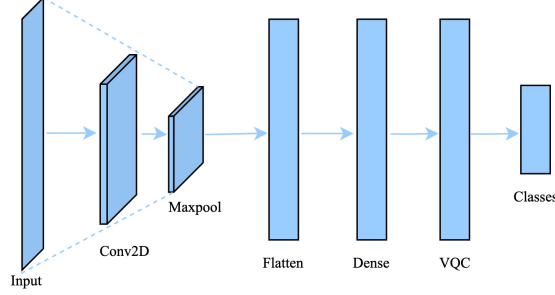

Figure 2: Architecture Diagram of Hybrid Quantum-Classical Neural Network

3.2. Model Architecture of QGAN

Due to the limited quantum resources of Noisy Intermediate-Scale Quantum (NISQ) devices, generating large-scale data directly on quantum hardware is impractical. Moreover, QGANs use random noise vectors as input, requiring lower computational accuracy and offering advantages over other quantum machine learning algorithms. To address these constraints, we adopt a quantum-classical hybrid architecture tailored for QGANs. The generator consists of a quantum variational circuit (VQC) and a classical neural network. The VQC leverages quantum gates and tunable parameters to exploit superposition and entanglement to generate quantum states with data features and complex correlations. The classical neural network post-processes the VQC output, mapping quantum states to data samples aligned with the training distribution via linear transformations and nonlinear activations. The discriminator is a classical neural network with multiple fully connected layers, capable of efficiently distinguishing generated from real samples. The architecture is illustrated in Fig. 3.

3.3. Algorithm of the General-purpose Data Augmentation Strategy

To enhance the performance of HQCNNs, we propose a general data augmentation scheme. This scheme enhances data processing and classification performance through standardized procedures. It uses a quantum generative adversarial network (QGAN) to generate high-quality samples, which are merged with the original dataset to expand data volume and improve generalization. Implementation details are shown in Algorithm 2.
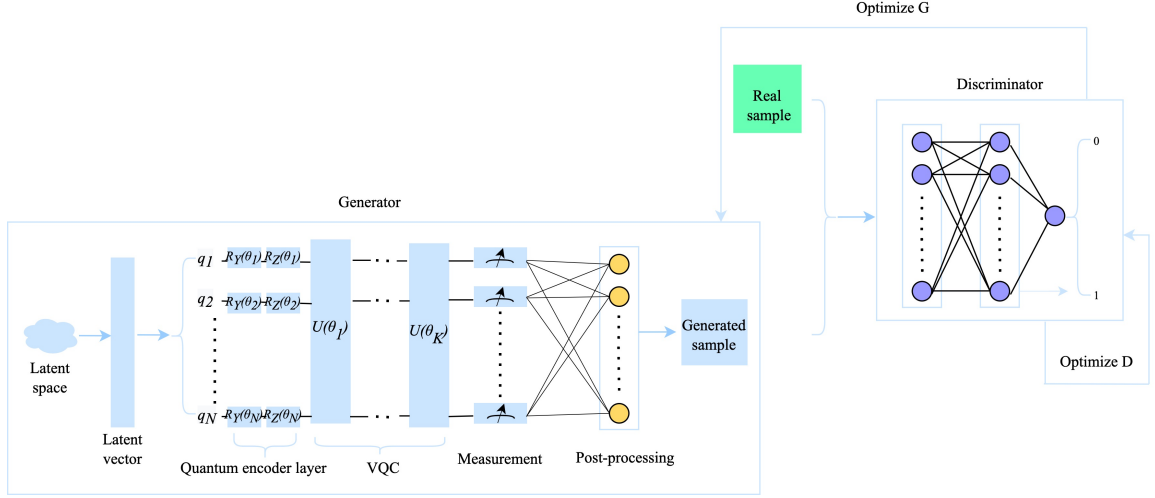
9

Figure 3: Architecture Diagram of Quantum Generative Adversarial Network

---

Algorithm 2 General Data Augmentation for HQCNNs

Input: HQCNN model M, training set D_{train}, test set D_{test}
Output: Retrained HQCNN model M'

1: Initialize QGAN: generator G, discriminator D
2: Initialize augmented sample set $D_{aug} = []$
3: Train G and D until QGAN converges
4: Generate N_{gen} samples and add to D_{aug}
5: Combine datasets: $D_{combined} \leftarrow D_{train} \cup D_{aug}$
6: Retrain M on $D_{combined}$
7: **Return** M'

3.4. Algorithm of the Customized Data Augmentation Strategy

To enhance a specific HQCNN's performance in low-accuracy categories without compromising high-accuracy ones, we propose a customized data augmentation method. The method consists of four components: classifier model, sampling module, generation model, and dataset augmentation module.

Classifier model: computes initial accuracy and confidence for each class and filters generated samples.

Sampling module: estimates the proportion of misclassified samples per class to guide sample generation and allocation.

Generation model: produces samples to expand the training set.

Dataset augmentation module: merges generated and original data to build the enhanced dataset. The complete process is illustrated in Algorithm 3.

4. Experimental Results and Analysis

This study conducts simulations on PyTorch and Qiskit using the MNIST dataset, a widely used benchmark in deep learning and quantum machine learning. MNIST consists of 28×28 grayscale digit images and serves as a standard benchmark for image classification [43]. Due to the high computational cost of simulating large-scale quantum systems, the MNIST task is simplified to a three-class problem with digits "0", "1", and "2". This approach simplifies quantum circuit design and fits current hardware limitations. To simulate real-world data scarcity, only 100 samples per class are used for training and testing, significantly reducing data size. This setting allows in-depth evaluation of HQCNN data augmentation using QGANs under resource constraints. We next analyze the results of applying QGANs, classical GANs, and conventional data augmentation (CDA) to HQCNNs. By comparing training efficiency and generation quality under identical datasets and hyperparameters, we quantify QGAN's advantages in parameter efficiency, iteration count, and resource usage. These findings offer theoretical and practical insights for advancing quantum data augmentation.

4.1. Comparison of GANs with Different Architectures

This section outlines the QGAN and classical GAN models used for comparison. The comparison includes the number of generator parameters and training iterations, as shown in Table 1.

Algorithm 3 Customized Data Augmentation for HQCNNs

Input: HQCNN model M, training set D_{train}, test set D_{test}, number of classes C, basic confidence threshold τ
Output: Retrained HQCNN model M'

1: Initialize: error counter E, generation proportion R, augmented sample set $D_{aug} = []$
2: **for** each $y \in D_{test}$ **do**
3:      Predict y' with M; if $y' \neq y$, then $E[y] \leftarrow E[y] + 1$
4: **end for**
5: $E_{total} \leftarrow \sum_i E[i]$; for each class i, compute $R[i] \leftarrow E[i]/E_{total}$
6: Adjust the QGAN training strategy based on error proportion R:

       a. Increase weights of high-error classes i in G
       b. Set training distribution of D based on R

7: Train generator G and discriminator D to generate samples
8: **for** each class i **do**
9:      $N_i \leftarrow N_{gen} \cdot R[i]$, generate $3N_i$ samples S_i
10:      Initialize filtered set $S'_i \leftarrow []$
11:      Compute the specific confidence threshold τ for the class:

       a. If $E[i]$ is large: $\tau = \tau - \alpha * R[i]$
       b. If $E[i]$ is small: $\tau = \tau + \beta * R[i]$

12:      **for** each sample s in S_i **do**
13:         Predict with M to get class c and confidence p, if $c = i$ and $p \geq \tau$ add s to S'_i
14:      **end for**
15:      **while** $|S'_i| < N_i$ **do**
16:         Regenerate until requirements met or max attempts reached
17:      **end while**
18:      Select first N_i samples from S'_i and add to D_{aug}
19: **end for**
20: Combine datasets $D_{combined} = D_{train} \cup D_{aug}$, Retrain model M
21: **Return** M'

Table 1: Parameters and Training Epochs of Different GANs

Model	Parameters	Epochs
QGAN	1,510,212	200
DCGAN	3,346,476	200 / 500
GAN-1	1,510,425	200
GAN-2	4,414,537	500

12

4.2. Performance of the General-purpose Data Augmentation Strategy

An effective data augmentation method should be broadly applicable. Therefore, in this section, we evaluate the performance of QGAN and classic GANs of various architectures on HQCNN and compare them with traditional augmentation methods (random rotation, translation, and contrast adjustment) to assess the general strategy's effectiveness. Following Algorithm 2, the strategy generates 300 augmented samples, evenly distributing 100 per class across digits "0", "1", and "2" to ensure class balance. The detailed experimental comparison results are shown in Fig. 4.

An effective data augmentation method should be broadly applicable.
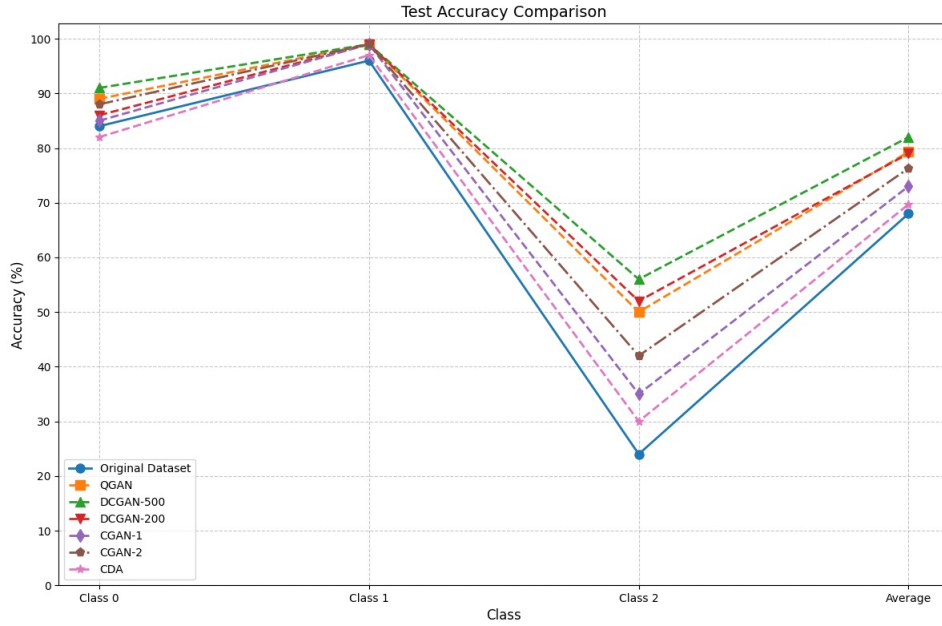

Figure 4: Comparison of Different Data Augmentation Methods

The experimental results yield two key findings, offering empirical support for applying quantum generative adversarial networks (QGANs) to data augmentation in hybrid quantum-classical neural networks (HQCNNs).

First, the accuracy curve shows that GAN-based augmentation consistently outperforms conventional methods. However, contrary to Ref. [44], while conventional augmentation offers limited improvement for HQCNNs, it does not reduce classification accuracy. This may be because our HQCNNs

use angle encoding and a hybrid quantum-classical architecture. This design helps preserve original features during quantum encoding, avoiding accuracy loss often caused by traditional augmentation. It also shows that the data diversity of traditional methods is limited, and it is difficult to overcome the performance bottleneck of HQCNN.

Secondly, QGAN demonstrated distinct advantages in the GAN-based augmentation comparison. Compared with GAN-1 (similar structure and parameters) and GAN-2 (more complex architecture), QGAN improved classification accuracy on MNIST by 6.3 and 3 percentage points, respectively, highlighting the superiority of quantum architectures in data generation. It is worth noting that QGAN achieves similar performance to DCGAN [45] with much fewer parameters and iterations, balancing efficiency and effectiveness. Additionally, quantum entanglement may enhance sample quality and diversity by implicitly modeling complex data correlations.

In summary, through multiple controlled experiments, this study confirms the effectiveness of the QGAN framework in enhancing HQCNN accuracy and performance, offering theoretical and practical guidance for quantum data augmentation.

4.3. Performance of the Customized Data Augmentation Strategy

This section presents experimental results of data augmentation strategies tailored for hybrid quantum-classical neural networks (HQCNNs). We first report initial performance metrics. We then compare results across three aspects: dynamic vs. non-dynamic sample generation, the performance of different GAN variants, and the trade-off between sample quality and quantity. The performance indicators of the initial classifier are shown in Table 2.

Table 2: Classification Accuracy and Average Prediction Confidence

Class	0	1	2	Average
Accuracy (%)	84	96	24	69
Average Prediction Confidence	0.485	0.505	0.315	0.435

4.3.1. Customized Augmentation *vs.* General Augmentation

Based on Algorithm 3 and Table 2, we generated 300 samples. We then determined the number of samples assigned to each class. For well-performing

14

categories, we retained images with confidence no lower than the initial value. These high-quality samples not only improved model performance for those classes but also helped balance the dataset. For underperforming categories, we lowered the threshold to include more samples, enhancing the model's learning capacity. Details are provided in Table 3. adversarial network (QGAN) as an example: This experiment uses samples generated by the quantum generative adversarial network (QGAN) as an example:

Table 3: Statistics for Different Classes

Class	0	1	2
Number of Generated Samples	45	11	244
Number of Augmented Dataset	145	111	344
Confidence	0.50	0.48	0.45

To establish a performance baseline, we implemented a general data augmentation strategy as a control. This strategy generates 300 augmented samples, with 100 per class to ensure balance. This strategy includes all generated samples in the training set without screening by confidence or class characteristics. This simulates the uniform expansion strategy often used in traditional augmentation, without considering the category-specific model performance. This control design highlights the value of the proposed customized strategy in boosting performance, while revealing the limitations of general augmentation in data-scarce settings. The comparison results are shown in Fig. 5.

4.3.2. The Enhancement Effects of Various GANs

This study systematically compares the data augmentation effects of four GAN architectures (Sec. 4.1) applied to HQCNNs. To ensure reliability and comparability, all four methods use a unified, customized augmentation strategy, applying dynamic sample generation and adjustable confidence screening thresholds for category-specific enhancement. The experiment controls key variables—including total generated data, sample quality, and HQCNN architecture—focusing on performance differences among classical GANs, DCGANs, and QGANs in enhancing HQCNN classification. Comparative results are shown in Fig. 6.

The experimental results show that the four types of GANs are effective in enhancing the performance of the specific HQCNN. Among them, DC-
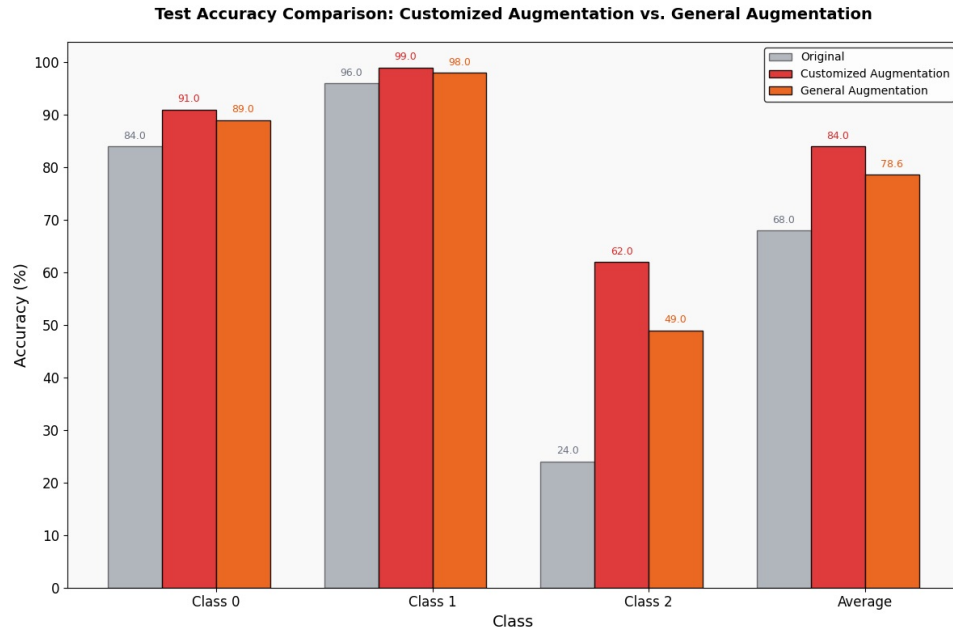
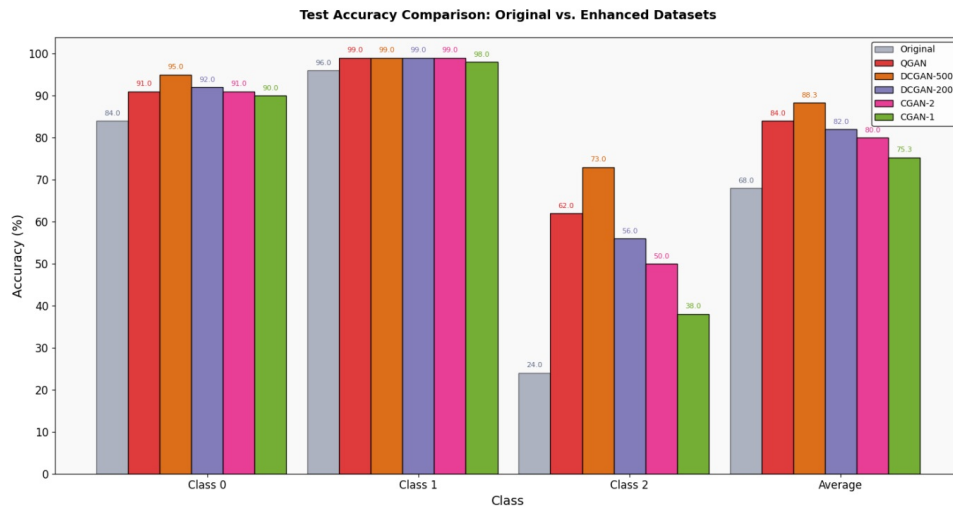Figure 5: Comparison between Customized Augmentation and General Augmentation

Figure 6: Comparison of Various GANs Enhancement Effects

GAN with 500 epochs achieved the best test accuracy (88.3%), confirming its effectiveness in image generation and feature extraction. Notably, the quantum generative adversarial network (QGAN) significantly outperforms its classical counterpart. Under identical experimental conditions, QGAN reduces parameters by two-thirds and training iterations by 60% compared to classical GAN-2, yet achieves higher classification accuracy. This highlights the unique strengths of quantum computing in data augmentation. Comparison with DCGAN further reveals QGAN's technical advantages. Although QGAN slightly underperforms DCGAN at 500 iterations, it surpasses DCGAN at 200 iterations (84% vs. 82%) using a shallower network and half the parameters. These results suggest that QGAN performs efficiently under limited resources and holds promise as a substitute for traditional deep models in constrained environments (e.g., quantum hardware), offering both theoretical and practical value for quantum data augmentation. As quantum computing advances, QGAN is poised to drive innovation in machine learning.

4.3.3. Sample Quality **vs.** Sample Quantity

This study investigates how the quality and quantity of generated samples affect data augmentation. Using systematic experimental design, we conducted multiple controlled experiments to analyze HQCNN performance under different sample quality-quantity combinations. We controlled variables such as model architecture and training hyperparameters, and adjusted the confidence screening threshold to vary sample quality and quantity. Table 4 details the experimental settings, and Fig. 7 shows HQCNN classification accuracy under different sample quality-quantity conditions.

Table 4: Statistics of Generated Samples and Confidence

Class	0	1		2	
Generated Samples	45	11	9	16	43
Augmented Dataset	145	111	109	116	143
Confidence	0.50	0.48	[0.45, 0.9)	[0.42, 0.45)	[0.38, 0.42)

Experimental results show that under constraints on sample quantity and generator performance, increasing the number of augmented samples in underperforming categories is more effective than improving sample quality.
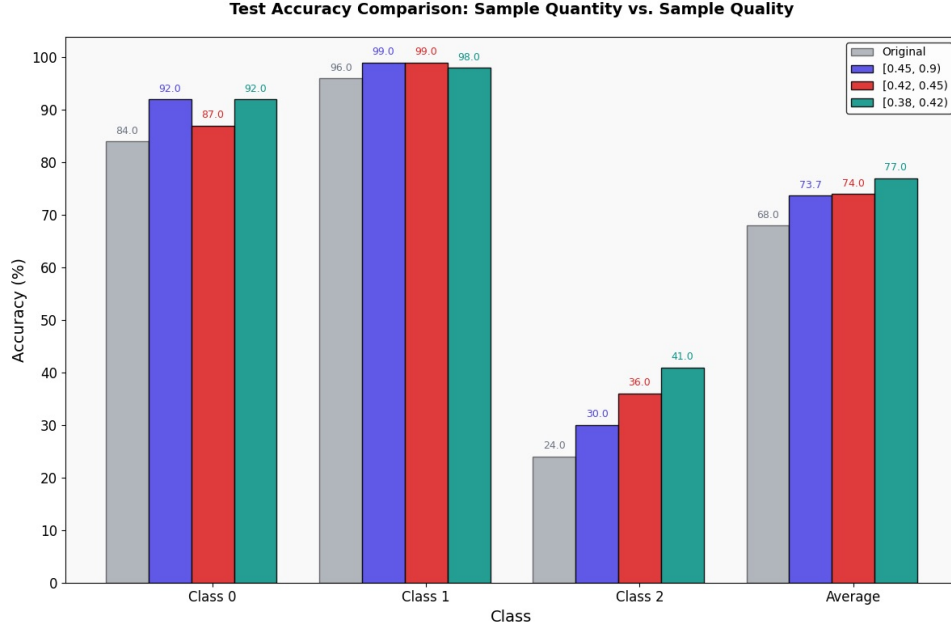
17

Figure 7: Comparison of Sample Quality and Sample Quantity

This suggests that in resource-constrained settings, prioritizing sample quantity in key categories more efficiently addresses model shortcomings, offering practical guidance for optimizing data augmentation strategies.

5. Conclusion

To tackle data scarcity in quantum machine learning (QML), we propose a data augmentation framework that integrates hybrid quantum-classical neural networks (HQCNNs) with quantum generative adversarial networks (QGANs). The framework includes two complementary strategies: a general strategy adaptable to various HQCNN architectures, and a customized strategy tailored to specific model weaknesses. Together, these strategies substantially improve model performance. Experiments on the MNIST dataset demonstrate the framework's effectiveness. Compared to traditional data augmentation techniques and classical GAN models, the QGAN-based augmentation strategy exhibits exceptional performance. Remarkably, with far fewer parameters than deep convolutional GANs (DCGANs), our approach achieves similar results, highlighting quantum computing's efficiency in gen-

erative modeling. These findings break through the limitations of conventional data augmentation methods, paving a transformative path for advancements in QML.

Future work should enhance QGAN's ability to generate diverse, high-quality samples at scale to meet complex task demands. Additionally, deeper integration of HQCNNs and QGANs is also needed to improve efficiency and accuracy in high-dimensional data processing. Progress in quantum hardware will be key to bridging the gap between theory and practice, enabling real-world deployment of quantum-enhanced machine learning.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

Data availability

No data was used for the research described in the article.

References

[1] R.P. Feynman, Simulating physics with computers, CRc Press, 2018.

[2] J. Biamonte, P. Wittek, N. Pancotti, Quantum machine learning, Nat. 549 (2017) 195–202, `https://dx.doi.org/10.1038/nature23474`.

[3] M. Schuld, I. Sinayskiy, F. Petruccione, An introduction to quantum machine learning, Contemp. Phys. 56 (2) (2015) 172-185, `https://dx.doi.org/10.1080/00107514.2014.964942`.

[4] V. Havlíček, A.D. Sinayskiy, K. Temme, Supervised learning with quantum-enhanced feature spaces, Nat. 567 (2019) 209-212, `https://dx.doi.org/10.1038/s41586-019-0980-2`.

[5] M. Sajjan, J. Li, R. Selvarajan, Quantum machine learning for chemistry and physics, Chem. Soc. Rev. 51 (15) (2022) 6475-6573, `https://dx.doi.org/10.1039/D2CS00203E`.

[6] J. Liu, K.H. Lim, K.L. Wood, Hybrid quantum-classical convolutional neural networks, Sci China Phys Mech. 64 (9) (2021) 290311, `https://dx.doi.org/10.1007/s11433-021-1734-3`.

[7] D. Arthur, P. Date, Hybrid quantum-classical neural networks, 2022 IEEE International Conference on Quantum Computing and Engineering (QCE). IEEE, 2022: 49–55, `https://dx.doi.org/10.1109/QCE53715.2022.00023`.

[8] F. Fan, Y. Shi, T. Guggemos, Hybrid quantum-classical convolutional neural network model for image classification, IEEE Trans. Neural Netw. Learn. Syst. 2023, `https://dx.doi.org/10.1109/TNNLS.2023.3312170`.

[9] Y. Liang, W. Peng, Z.J. Zheng, A hybrid quantum–classical neural network with deep residual learning, Neural Networks. 143 (2021) 133-147, `https://dx.doi.org/10.1016/j.neunet.2021.05.028`.

[10] E. Paquet, F. Soleymani, QuantumLeap: Hybrid quantum neural network for financial predictions, Expert Syst. Appl. 195 (2022) 116583, `https://dx.doi.org/10.1016/j.eswa.2022.116583`.

[11] A.I. Gircha, A.S. Boev, K. Avchaciov, Hybrid quantum-classical machine learning for generative chemistry and drug design, Sci. Rep. 13 (1) (2023) 8250, `https://dx.doi.org/10.1038/s41598-023-32703-4`.

[12] J. Preskill, Quantum computing in the NISQ era and beyond, Quantum. 2 (2018) 79, `https://dx.doi.org/10.22331/q-2018-08-06-79`.

[13] K. Bharti, A. Cervera-Lierta, T.H. Kyaw, Noisy intermediate-scale quantum algorithms, Rev. Mod. Phys. 94 (1) (2022) 015004, `https://dx.doi.org/10.1103/RevModPhys.94.015004`.

[14] C. Shorten, T.M. Khoshgoftaar, A survey on image data augmentation for deep learning, J. Big Data. 6 (1) (2019) 1-48, `https://dx.doi.org/10.1186/s40537-019-0197-0`.

[15] M.A. Nielsen, I.L. Chuang, Quantum computation and quantum information, Cambridge University Press, 2010.

[16] S. Nokhwal, S. Nokhwal, S. Pahune, Quantum generative adversarial networks: Bridging classical and quantum realms, Proceedings of the 2024 8th International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence, 2024: 105-109, `https://dx.doi.org/10.1145/3665065.3665082`.

[17] P.L. Dallaire-Demers, N. Killoran, Quantum generative adversarial networks, Phys. Rev. A. 98 (1) (2018) 012324, `https://dx.doi.org/10.1103/PhysRevA.98.012324`.

[18] S. Lloyd, C. Weedbrook, Quantum generative adversarial learning, Phys. Rev. Lett. 121 (4) (2018) 040502, `https://dx.doi.org/10.1103/PhysRevLett.121.040502`.

[19] C. Zoufal, A. Lucchi, S. Woerner, Quantum generative adversarial networks for learning and loading random distributions, npj Quantum Inf. 5 (1) (2019) 103, `https://dx.doi.org/10.1038/s41534-019-0223-2`.

[20] C.A. Riofrio, O. Mitevski, C. Jones, A characterization of quantum generative models, ACM Trans. Quantum Comput. 5 (2) (2024) 1-34, `https://dx.doi.org/10.1145/3655027`.

[21] M. Hibat-Allah, M. Mauri, J. Carrasquilla, A framework for demonstrating practical quantum advantage: comparing quantum against classical generative models, Commun. Phys. 7 (1) (2024) 68, `https://dx.doi.org/10.1038/s42005-024-01552-6`.

[22] M. Benedetti, E. Grant, L. Wossnig, Adversarial quantum circuit learning for pure state approximation, New J. Phys. 21 (4) (2019) 043023, `https://dx.doi.org/10.1088/1367-2630/ab14b5`.

[23] J. Zeng, Y. Wu, J.G. Liu, Learning and inference on generative adversarial quantum circuits, Phys. Rev. A. 99 (5) (2019) 052306, `https:/dx.doi.org/10.1103/PhysRevA.99.052306`.

[24] S. Ahmed, C. Sánchez Muñoz, F. Nori, Quantum state tomography with conditional generative adversarial networks, Phys.

Rev. Lett. 127 (14) (2021) 140502, `https://dx.doi.org/10.1103/PhysRevLett.127.140502`.

[25] M.Y. Niu, A.Zlokapa, M. Broughton, Entangling quantum generative adversarial networks, Phys. Rev. Lett. 128 (22) (2022) 220505, `https://dx.doi.org/10.1103/PhysRevLett.128.220505`.

[26] J. Li, R.O. Topaloglu, S. Ghosh, Quantum generative models for small molecule drug discovery, IEEE Trans. Quant. Eng. 2 (2021) 1-8, `https://dx.doi.org/10.1109/TQE.2021.3104804`.

[27] D. Herr, B. Obert, M. Rosenkranz, Anomaly detection with variational quantum generative adversarial networks, Quantum Sci. Technol. 6 (4) (2021) 045004, `https://dx.doi.org/10.1088/2058-9565/ac0d4d`.

[28] H. Situ, Z. He, Y. Wang, Quantum generative adversarial network for generating discrete distribution, Inform. Sciences. 538 (2020) 193-208, `https://dx.doi.org/10.1016/j.ins.2020.05.127`.

[29] A. Broadbent, E. Kashefi, Parallelizing quantum circuits, Theor. Comput. Sci. 410 (26) (2009) 2489-2510, `https://dx.doi.org/10.1016/j.tcs.2008.12.046`.

[30] C.H. Bennett, G. Brassard, Quantum cryptography: Public key distribution and coin tossing, Theor. Comput. Sci. 560 (2014) 7–11, `https://dx.doi.org/10.1016/j.tcs.2014.05.025`.

[31] D. Bouwmeester, J.W. Pan, Experimental quantum teleportation, Nat. 390 (6660) (1997) 575-579, `https://dx.doi.org/10.1038/37539`.

[32] S. Sim, P.D. Johnson, A. Aspuru-Guzik, Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms, Adv. Quantum Technol. 2 (12) (2019) 1900070, `https://dx.doi.org/10.1002/qute.201900070`.

[33] A. Abbas, D. Sutter, C. Zoufal, The power of quantum neural networks, Adv. Nat. Comput. Sci. 1 (6) (2021) 403-409, `https://dx.doi.org/10.1038/s43588-021-00084-1`.

[34] S.Y.C. Chen, T.C. Wei, C. Zhang, Quantum convolutional neural networks for high energy physics data analysis, Phys. Rev. Res. 4 (1)

(2022) 013231, `https://dx.doi.org/10.1103/PhysRevResearch.4.013231`.

[35] S.Y.C. Chen, S. Yoo, Y.L.L. Fang, Quantum long short-term memory, ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2022: 8622-8626, `https://dx.doi.org/10.1109/ICASSP43922.2022.9747369`.

[36] C.H.H Yang, J. Qi, S.Y.C. Chen, Decentralizing feature extraction with quantum convolutional neural network for automatic speech recognition, ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2021: 6523-6527, `https://dx.doi.org/10.1109/ICASSP39728.2021.9413453`.

[37] Zhu J-Y, Krähenbühl P, Shechtman E, et al. Generative visual manipulation on the natural image manifold[C]. European conference on computer vision, 2016: 597-613. J.Y. Zhu, P. Krähenbühl, E. Shechtman, Generative visual manipulation on the natural image manifold, Computer vision–ECCV 2016: 14th European conference, 2016: 597-613, `https://dx.doi.org/10.1007/978-3-319-46454-1_36`.

[38] A.V.D. Oord, S. Dieleman, H. Zen, Wavenet: A generative model for raw audio, 12 (2016), `https://dx.doi.org/10.48550/arXiv.1609.03499`.

[39] I. Gulrajani, F. Ahmed, M. Arjovsky, Improved training of wasserstein gans, Adv. Neural Inf. Process. Syst. 30 (2017).

[40] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, Generative adversarial nets, Adv. Neural Inf. Process. Syst. 27 (2014).

[41] H.L. Huang, Y. Du, M. Gong, Experimental quantum generative adversarial networks for image generation, Phys. Rev. Appl. 16 (2) (2021) 024051, `https://dx.doi.org/10.1103/PhysRevApplied.16.024051`.

[42] K. Mitarai, M. Negoro, M. Kitagawa, Quantum circuit learning, Phys. Rev. A. 98 (3) (2018) 032309, `https://dx.doi.org/10.1103/PhysRevA.98.032309`.

[43] D. Li, The mnist database of handwritten digit images for machine learning research, IEEE Signal Process. Mag. 29 (6) (2012) 141–142, `https://dx.doi.org/10.1109/MSP.2012.2211477`.

[44] L. Zhouli, P. Wang, U. Parampalli, Impact of Data Augmentation on QCNNs, 2023, `https://dx.doi.org/10.48550/arXiv.2312.00358`.

[45] A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, 2015, `https://dx.doi.org/10.48550/arXiv.1511.06434`.