Tian et al. *EPJ Quantum Technology* (2025) 12:80
https://doi.org/10.1140/epjqt/s40507-025-00372-z

EPJ Quantum Technology
a SpringerOpen Journal

RESEARCH Open Access

Quantum generative adversarial network with automated noise suppression mechanism based on WGAN-GP

Yanbing Tian[1], Cewen Tian[1], Zaixu Fan[1], Minghao Fu[2] and Hongyang Ma[1*]

*Correspondence:
hongyang_ma@aliyun.com
[1]School of Information and Control Engineering, Qingdao University of Technology, Qingdao, China
Full list of author information is available at the end of the article

Abstract

Quantum Machine Learning (QML) has attracted significant attention for its potential to deliver exponential advantages over classical machine learning approaches, particularly in classification and recognition tasks. Quantum Generative Adversarial Networks (QGANs), a form of quantum machine learning, provide promising advantages in image processing and generation tasks when compared to classical technologies. However, the limitations of current quantum devices have led to suboptimal image quality and limited robustness in earlier methods. To overcome these challenges, we developed a hybrid quantum-classical approach, introducing CAQ, a quantum-classical Generative Adversarial Network (GAN) framework. Leveraging the latest WGAN-gradient penalty (GP) strategy, we trained and optimized the quantum generator, reduced the complexity of parameters, and implemented an adaptive noise input system that dynamically adjusts noise levels, thereby improving the model's robustness. Additionally, we employed a remapping technique to transform the original image's multimodal distribution into a unimodal one, thereby reducing the complexity of the learned distribution. Experiments on MNIST and Fashion-MNIST datasets show that CAQ generates grayscale images effectively, demonstrating its feasibility on near-term intermediate-scale quantum (NISQ) computers.

Keywords: Machine learning algorithms; Quantum circuit; Quantum computing

1 Introduction

Generative Adversarial Networks (GANs) are a type of neural network model used in computer vision and machine learning. They represent one of the successful applications of deep learning in generative modeling [1]. A classical GAN consists of two components: the generator and the discriminator. These two components compete against each other in a game-theoretic manner, enabling the generator to produce increasingly realistic data. The generator attempts to produce realistic data to "deceive" the discriminator, while the discriminator seeks to distinguish between real data and data generated by the generator, acting as a "quality inspector," thereby creating an adversarial competition between the two. Due to its outstanding performance in areas such as image generation [2], data aug-

mentation [3], and future visual prediction [4], classical GANs have garnered widespread attention since their inception.

Consequently, this has sparked significant interest in the quantum computing community in developing a quantum-enhanced GAN. Quantum GANs have already made some groundbreaking progress, indicating that efforts to develop quantum GANs are worthwhile. However, we have identified issues in early quantum GANs, such as training instability [5] and mode collapse [6], indicating significant room for improvement. This is particularly relevant in the context of generating high-resolution images on current noisy intermediate-scale quantum (NISQ) devices.

1.1 For the current state of research on quantum generative adversarial networks

The emergence of Variational Quantum Circuits (VQCs) provides a bridge, combining classical machine learning methods with quantum computing. VQCs have already been applied to various tasks, including quantum machine learning [7], optimization problems [8], pattern recognition [9], classification, and regression. This suggests that quantum generative models may have stronger expressive power than classical generative models. Consequently, in 2018, the Quantum Generative Adversarial Network (QGAN) [10] was first proposed. Dallaire-Demers et al. [11] successfully trained a QGAN based on Parameterized Quantum Circuits (PQC) and conducted basic data analysis experiments. Maria Schuld et al. [12] combined the concepts of quantum variational inference and autoencoders, proposing the idea of Quantum Variational Autoencoders (QVAEs). Benedetti et al. [13] used two PQCs, leveraging quantum feature space to produce approximately pure states. Situ et al. [14] utilized Quantum Hybrid Generative Adversarial Networks to address the generation of discrete distributions, a problem that classical GANs cannot directly solve. Huang et al. [15] implemented Quantum Generative Adversarial Learning in superconducting quantum circuits, with the quantum generator achieving high fidelity (average 98.8) output. In 2020, Huang et al. [16] proposed a QGAN based on a patch strategy and conducted experiments in superconducting quantum circuits. In 2022, Kiani et al. [17] proposed a Quantum Wasserstein Generative Adversarial Network (QWGAN) based on the Quantum Earth Mover (EM) and effectively learned quantum data. Agliardi et al. [18] optimized the QGAN model by adjusting parameters to generate multivariate distributions, further enhancing the performance of Quantum Generative Adversarial Networks.

2 Related work

2.1 Wasserstein generative adversarial networks (WGAN)

During the training process of classical GANs, minimizing JS and KL divergences often leads to instability and issues like mode collapse [20–24]. To address these problems, researchers introduced alternative objectives to redefine GAN training strategies. WGAN [25], for instance, utilizes the Wasserstein distance to optimize the network by minimizing it. The objective function is:

$$\min_{G} \max_{D \in \mathcal{D}} \mathbb{E}_{\mathbf{x} \sim P_{\text{data}}}[D(x)] - \mathbb{E}_{z \sim P_z}[D(G(z))] \tag{1}$$

Here, D represents the set of all 1-Lipschitz functions. Instead of outputting a binary classification probability, D provides a score reflecting the distance between the generated

distribution P_G and the true distribution P_{data}, making it a critic. Minimizing the Wasserstein distance offers advantages over JS and KL divergences, which are only defined when the two distributions share the same support. In contrast, Wasserstein distance can measure the distance between any two distributions and remains valid even when JS and KL divergences fail. It also helps prevent vanishing gradients, improving training stability [26].

However, WGAN has limitations. Arjovsky et al. implemented Lipschitz constraints by clipping the weights of the critic's parameters within a fixed range, but this approach leads to issues like slow convergence and vanishing gradients [27]. To address these drawbacks, Gulrajani et al. proposed WGAN-GP [28, 29], which adds a gradient penalty (GP) term to enforce 1-Lipschitz continuity. The modified objective function is:

$$min \max_{G} \mathbb{E}_{\mathbf{x}\sim P_{data}}[D(x)] - \mathbb{E}_{\mathbf{z}\sim P_{\mathbf{z}}}[D\big(G(z)\big)] + \lambda \mathbb{E}_{\hat{x}}[(\parallel \nabla_{\hat{x}} D(\hat{x}) \parallel_2 -1)^2]^{\leftarrow} \tag{2}$$

The gradient penalty term ensures that the gradient at the interpolated point $\hat{x}$ remains within a reasonable range. In WGAN-GP, the critic continuously applies the condition in the equation, preventing vanishing gradients and ensuring smoother training. The use of Wasserstein distance leads to more stable gradients, reducing mode collapse and improving both generator and critic training. WGAN-GP has been widely applied to tasks such as image generation and synthetic data augmentation for deep learning models.

2.2 Gate-model quantum neural networks

Gate-based quantum computers are built upon a sequence of quantum logic gates, each represented by a unitary matrix $U \in \mathbb{C}^{2^n \times 2^n}$, acting on quantum states in a Hilbert space. The general evolution of a quantum state $|\psi_{\text{in}}\rangle$ through a gate-based circuit is expressed as:

$$|\psi_{\text{out}}\rangle = U_k(\theta_k) \cdots U_2(\theta_2) U_1(\theta_1) |\psi_{\text{in}}\rangle$$

where $U_i(\theta_i)$ denotes a parameterized quantum gate (often a rotation gate such as R_x, R_y, R_z) with a tunable parameter θ_i. These parameters serve as trainable weights in the quantum neural network.

The Gate-Model Quantum Neural Network (Gate-QNN) adopts this principle to construct a learnable model [35]. A QNN is defined as a sequence of parameterized unitaries followed by a measurement, producing a classical label prediction $\hat{y}$. The goal is to minimize the deviation from the true label y using a loss function L:

$$L(\boldsymbol{\theta}) = \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\ell \left(y, \hat{y}(\boldsymbol{\theta}, x) \right) \right]$$

where $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_k)$ are the gate parameters and ℓ is typically a cross-entropy or mean square loss depending on the encoding and task.

The measurement step applies a positive operator-valued measure (POVM) or a projective measurement in the computational basis, giving outcome probabilities $p(y|\boldsymbol{\theta}, x)$. In binary classification tasks, the prediction can be based on the expectation value of a Hermitian observable M, such as:

$$\langle M \rangle = \langle \psi_{\text{out}} | M | \psi_{\text{out}} \rangle$$

This value is post-processed into classical prediction labels. Furthermore, a Recurrent Gate-QNN (RQNN) extends this model by allowing quantum or classical side information to propagate backward, effectively encoding temporal dependencies. The RQNN can be modeled as a sequence of parameterized unitaries $U_t(\theta_t, h_{t-1})$, where h_{t-1} denotes information from the previous time step, giving the recurrence:

$$|\psi_t\rangle = U_t(\theta_t, h_{t-1})|\psi_{t-1}\rangle$$

This structure enables the modeling of sequential or time-dependent quantum data, similar to classical recurrent neural networks (RNNs), but implemented within the quantum framework.

The cited work proposes a constraint-based optimization framework for training both non-recurrent and recurrent Gate-QNNs [36], taking into account the availability and direction of side information, and proving that optimal learning strategies vary significantly across these architectures. These studies provide a new theoretical foundation for training quantum neural networks, especially in addressing the challenges of quantum data processing and temporal dependencies in quantum computing.

2.3 Parametric quantum circuit (PQC)

The variational quantum circuit (PQC) [30] is usually organized as a quantum neural network, which is a circuit consisting of a series of parameterized quantum gates, where the parameters of the quantum gates in the circuit are controlled by θ, and these parameters can be adjusted by classical optimization algorithms in order to minimize the objective function. The PQC generally consists of an initial state, quantum gates with the parameter θ, and a measurement operation. The initial state is $|0\rangle^{\otimes n}$, and this initial quantum state will be used as the input of the PQC. The initial quantum state can be transformed into a new state by continuously adjusting the parameter θ of the quantum gate through the optimization algorithm, until the convergence condition is satisfied. Finally, the expectation value of the quantum circuit, obtained through the measurement operation, can be expressed as:

$$E(\theta) = \langle 0 \mid U^{\dagger}(\theta)\hat{M}U(\theta) \mid 0\rangle \tag{3}$$

Where U represents the youngest positive transformation of the gate, M represents the measurement operation of the expectation value, and θ represents the parameter set of a series of quantum gates. The optimization objective of a quantum circuit is often to solve a specific task by finding the optimal set of gate parameters θ such that the expectation value $E(\theta)$ is minimized or a specific objective condition is satisfied. The optimization process of PQC is shown in Fig. 1.

3 Design of the CAQ algorithm model

In this section, we introduce the CAQ framework for generating high-quality images in NISQ. CAQ consists of a generator and a critic, and the method of WGAN-GP [19] was chosen for training because of better convergence compared to weight clipping. To address scalability and quantum resource bottlenecks in generating high-resolution images, CAQ utilizes a remap to transform the image dataset, simplifying the complex generation

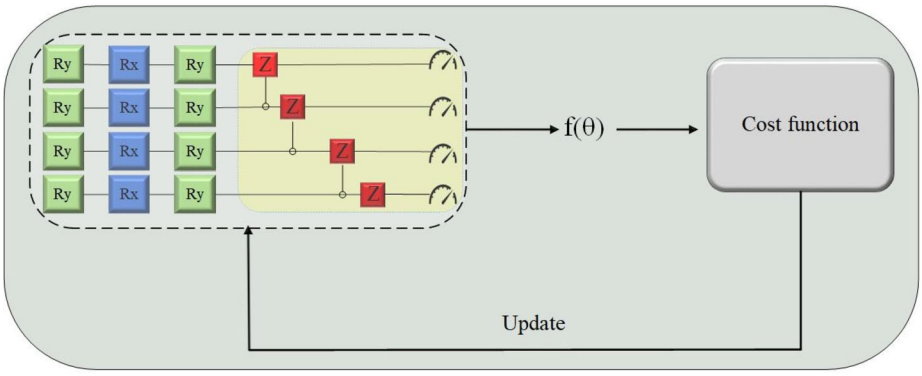

Figure 1 Optimization process of PQC

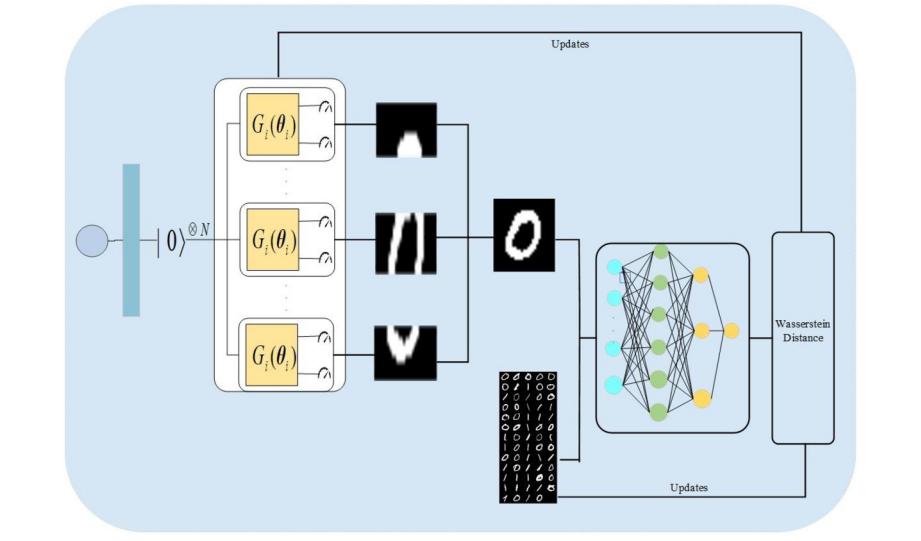

Figure 2 CAQ's overall model architecture

task. Meanwhile, CAQ adds an automatic noise input system to mitigate the risk of pattern collapse. To reduce the hardware cost and circuit depth, CAQ uses a simplified version of the PQC circuit with greatly reduced parameters and no performance degradation. Unlike classical GAN, the variational quantum generator does not need to map a priori noise distributions into a high-dimensional space, but instead obtains different results through the randomness of quantum measurements.CAQ introduces the design and implementation of a quantum-classical hybrid Generative Adversarial Network (GAN) for image generation on real quantum computers, as illustrated in Fig. 2.

3.1 Automatic noise reloading system

In a generator composed of classical neural networks, a GAN requires the input of a noise vector "z" to obtain different samples $G(z)$. Similarly, in quantum circuits, noise vectors are crucial for the training of quantum generators, which cannot be successfully trained for stable performance by the randomness of quantum circuits alone. In fact, there are no experiments showing that quantum generators can be trained to converge adversarial

networks to optimal distributions without noisy inputs. In addition, ineffective utilization of noisy inputs can cause the generator to always omit some training samples and assign too high a probability to others, thus failing to generate a variety of images in the same class, a problem also known as "pattern collapse".

In order to solve the above problem, in the QPatch framework [16], a noise input with a fixed range in the interval of $[0, \frac{\pi}{2}]$ is used to a certain extent, the noise is utilized to train the generator and increase the stability of learning the correct distribution, L_{D_1} and L_{G_1} due to the fact that its noise is always distributed in the range, it can not cope with the various conditions in the training process to make the adjustment. A noise input system is proposed in MasaiQ [41] to automatically adjust the noise range. Within the adaptive noise regulation framework, CAQ builds upon the foundational principles of MasaiQ, leveraging dynamic noise range adjustment to enhance the stability of QGAN training. However, CAQ further refines this approach by optimizing both the noise parameter range and the choice of activation functions, ensuring improved adaptability and training efficiency.

In contrast to MasaiQ, which employs a fixed noise adjustment strategy, CAQ leverages $\frac{L_G}{L_C}$ as the core noise adjustment metric and introduces an adaptive noise control mechanism wherein the minimum noise level is elevated to $\frac{\pi}{6}$ and the maximum is dynamically modulated, initially set at $\frac{2\pi}{3}$. Unlike MasaiQ's fixed noise range, CAQ enables smooth, data-driven variation through a Sigmoid + Leaky ReLU transformation, facilitating more effective noise adaptation across different learning phases. Additionally, CAQ replaces MasaiQ's Tanh + ReLU activation scheme, which is susceptible to gradient vanishing, with Sigmoid for smoother transitions and Leaky ReLU to mitigate gradient stagnation. Crucially, CAQ integrates a training-aware noise adaptation strategy that varies noise magnitudes across different learning phases—allowing greater noise levels for early-stage diversity exploration and progressively refining the generator's precision in later training iterations. We denote the auto-tuned noise system based on the maximum threshold as:

$$\text{Noise}_{max} = \frac{2\pi}{3} * [LeakyReLU\left(Sigmoid\left(\frac{L_C}{L_G} - \frac{L_{G_1}}{L_{C_1}}\right)\right)] \tag{4}$$

Where L_{G_1} and L_{C_1} are the values of the generator and critic's loss functions after one iteration of training. As shown in the above equation, the auto-adjustment range will not exceed the threshold of $\frac{2\pi}{3}$. Although the lower bound is fixed not to go below $\frac{\pi}{6}$, the upper bound will adaptively adjust the noise range according to the conditions in time and efficiently to adapt to different training conditions.

CAQ introduces the Sigmoid function, which is a smooth, continuous function, implying that it is differentiable throughout its domain of definition, with the functional expression $Sigmoid(x) = \frac{1}{1+e^{-x}}$. As x approaches positive infinity, the function value tends to 1; as x approaches negative infinity, the function value tends to 0. Thus, it can restrict the output to between the range $[0, 1]$. Since the output of the Sigmoid function is not zero-centered, it means that the input data are not normalized, and the output may deviate in a certain direction to a value less than zero. Further, CAQ introduces the Leaky ReLU function which is defined as Leaky ReLU$(x) = max(\alpha x, x)$, where α is a constant [32, 33]. It provides a non-linear activation by normalizing all negative values to zero, which allows a small, non-zero output for negative input values compared to the standard ReLU function, preventing the activation function from being completely inactive for negative inputs. The activation functions used by CAQ, the Sigmoid function and the ReLU function, mainly

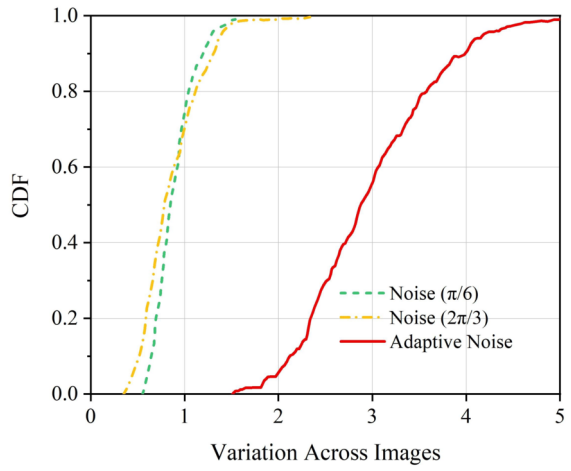

Figure 3 The scoring of CDF scores proves that Adaptive noise systems in CAQ play an effective role

perform their functions of normalizing and scaling the data so that the outputs are distributed within a reasonable range and the noise is reduced [37]. A reasonable range so that the noise is more adapted to the training conditions of the generator.

In order to verify the effectiveness of the automatic noise suppression mechanism, we designed simulation experiments to evaluate its adaptability and stability under different training conditions. In the experiments, we set different upper and lower noise limits respectively, and observe the generator's training performance and the diversity of generated samples under different conditions [34, 38]. As shown in the Fig. 3, we use the CDF score well to verify its effectiveness. And the experimental results show that: When the noise input can be adaptively adjusted, the generator can maintain high image generation quality at different training stages and avoid the occurrence of pattern collapse. Dynamically adjusting the upper and lower limits of the noise enables the generator to generate more diverse images and the quality of the generated samples is significantly higher than that of the generator with fixed noise input.

3.2 Remapping of processed images

CAQ introduces a method to remap the gray values of pixels in the original image [39], which greatly reduces the difficulty of the image generation task by transforming the multi-peaked distribution in the original image into the easier-to-learn single-peaked distribution in the task of generating an adversarial network to learn a discrete distribution.

First, the gray values of the pixels in the original image are sorted from largest to smallest, and the distribution positions of the original gray values are adjusted to form a new gray image. In the new image, the first pixel corresponds to the position of the lowest gray value in the original image, which is sequentially distributed upwards, and the last pixel corresponds to the position of the highest gray value in the original image. This results in a new grayscale image where the order of the pixels depends on the gray value sorted position of the pixels in the original image. The process of remapping is shown schematically in Fig. 4.

The grayscale values of the input image are converted into a state representation suitable for quantum computation, after which Amplitude Encoding is applied to load the data into

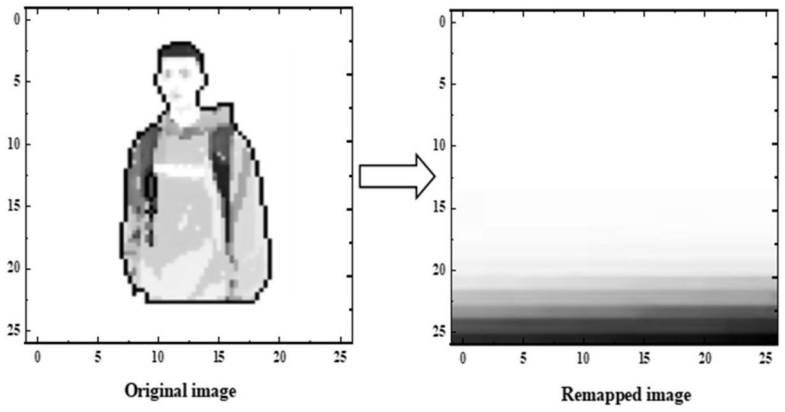

Figure 4 Remapping rearranges the image process pixel by pixel

the quantum state as it can efficiently represent the normalized pixel data. We can map the normalized pixel gray values to the quantum state. Given a N-pixel grayscale image with each pixel grayscale value x_i belonging to $[0, 255]$, we first perform normalization:

$$\tilde{x}_i = \frac{x_i}{\sum_{i=0}^{N-1} x_i} \tag{5}$$

The normalized gray values are then mapped to quantum states:

$$|\psi\rangle = \sum_{i=0}^{N-1} \alpha_i |i\rangle, \quad \text{where} \quad \alpha_i = \frac{\tilde{x}_i}{\sqrt{\sum_{j=0}^{N-1} \tilde{x}_j^2}} \tag{6}$$

The $|\psi\rangle$ is trained by a quantum circuit: the quantum generator produces a new distribution of pixel gray values (still in the quantum state), the pixel values are obtained and the image is reconstructed by measuring the quantum state, the classical discriminator evaluates the similarity between the generated image and the real image and calculates the loss, and the loss function is applied to continuously adjust the parameterized quantum gates and perform the reverse mapping to recover the original data features. The remapping method can be considered a simplified alternative to Principal Component Analysis (PCA), as both techniques share the fundamental objective of reducing data complexity, optimizing data distribution, and facilitating the training of Quantum Generative Adversarial Networks (QGANs) or other generative models. However, their underlying mechanisms and effects on image generation differ significantly.

MosaiQ [41] utilizes PCA-based dimensionality reduction to preprocess input images, effectively reducing data dimensionality and computational overhead. However, this transformation may inadvertently attenuate high-frequency components, potentially leading to a loss of fine details in the generated images. In contrast, the remapping method eschews linear transformations and instead reorders pixel intensity values, reshaping the statistical distribution into a more learnable form. This approach enhances the training efficiency of QGANs while mitigating quantum circuit depth requirements, making it particularly well-suited for near-term quantum computing architectures. To quantitatively compare these methods, we adopt the Fréchet Inception Distance (FID), where lower FID

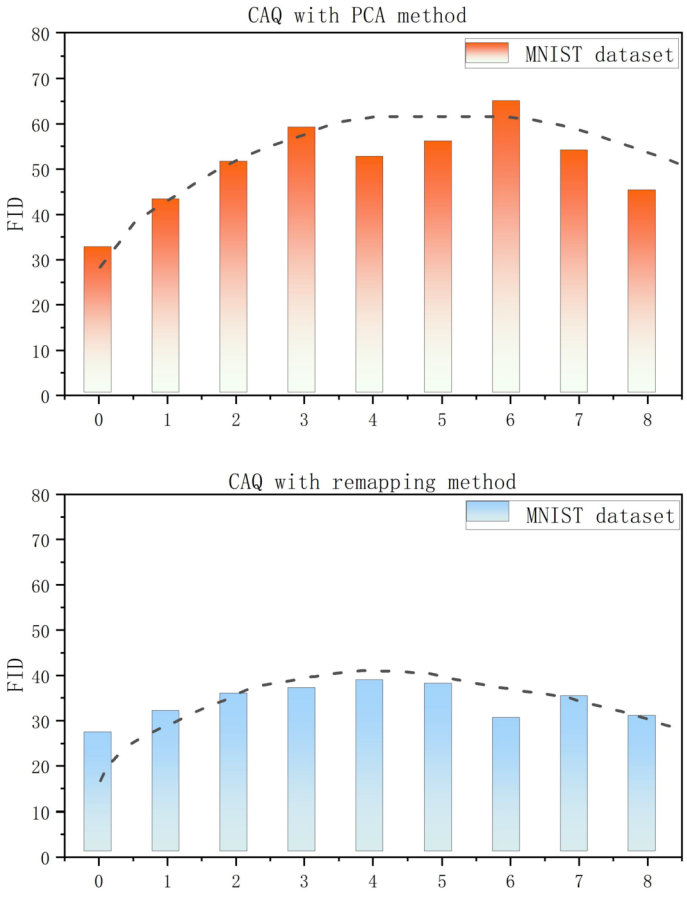

Figure 5 In the MNIST handwriting dataset for digits 0-8, CAQ uses a remapping method to consistently generate lower FID scores, corresponding to higher quality images, compared to the PCA method

values indicate higher fidelity to real data. As shown in Fig. 5, the remapping method (blue) achieves a lower FID than PCA (orange), confirming its advantage in preserving the distributional characteristics of real images while improving training efficiency. These findings highlight the potential of the remapping method as a lightweight yet effective preprocessing technique, making it particularly beneficial for quantum generative models operating within the limitations of Noisy Intermediate-Scale Quantum (NISQ) devices.

3.3 Structure of simolidied generator

In the classical generative task, the generator samples random probabilities to perform quantum measurements of the ground state. Given that the initial input state is $|0\rangle^{\otimes n}$, only a rotational layer is required, primarily composed of R_x and R_y gates. In our simplified quantum generator, after encoding via R_y gates, we apply an additional R_x gate to each qubit, i.e., $U_R = \prod_{i=1}^{n} R_x(\theta_l^i)$. This reduces the number of parameters and resource usage significantly compared to more complex rotational layers.

To explore the effect of two-qubit gates, we tested CZ, CNOT, and ISWAP gates. Notably, CNOT performed worse than the other two. The best results were achieved with the CZ gate, yielding a fidelity of 0.946, suggesting better trainability of entanglement.

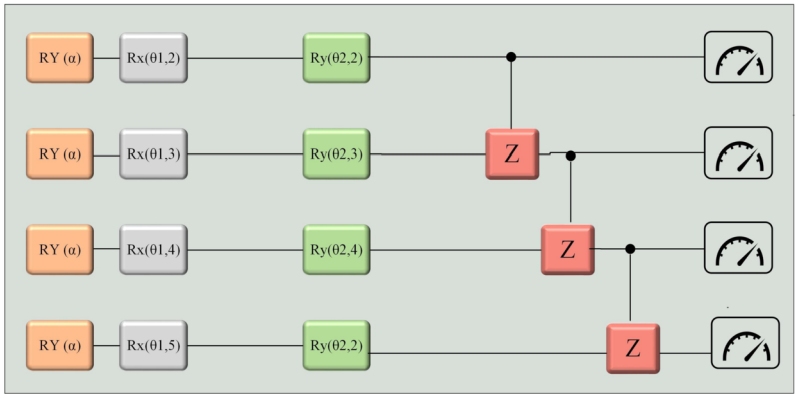

Figure 6 The structure of each sub-generator of CAQ. The orange blocks represent $RY(\boldsymbol{\alpha})$ gates, which initialize the quantum state and introduce superposition. The grey blocks correspond to $RX(\theta_1)$ rotations along the X-axis, modulating both amplitude and phase distributions. The green blocks denote additional $RY(\theta_2)$ operations, enhancing the expressivity of the quantum state

Given the lower hardware overhead, we propose using duplicated CZ gates in the CAQ for entanglement.

All sub-generators in CAQ follow the same structure, with parameterized quantum circuits (PQC) on five qubits. Each sub-generator's circuit, shown in Fig. 6, noisily encodes data into angles via R_y and R_x gates, with adjacent qubits entangled through CZ gates. After several layers, parametric weights are optimized, and measurements on ancilla qubits perform nonlinear transformations. This design reduces PQC depth, mitigating hardware errors, minimizing parameter counts, and still generating high-quality images. Further experimental validation is provided in Sect. 5.

The generator samples an N-dimensional noise vector $L = (l_1, l_2, \ldots, l_N)$ from some prior distribution P_z (e.g., Gaussian distribution [22]). Immediately after, the components of the noise vector L are input to the R_y layer rotational encoding that is parameterized in the subgenerator. From the input $|0\rangle\widehat{\otimes}n$ to the generator, we obtain the state by encoding the circuit:

$$|L\rangle = R_Y^1(l_1)R_Y^2(l_2)\ldots R_Y^N(l_N)|0\rangle^{\otimes N} \tag{7}$$

Where $R(\lambda,\theta,\phi)$ can be denoted as $R_z(\lambda)R_y(\theta)R_z(\phi)$, the aim is to change a quantum bit into an arbitrary quantum state on the Bloch sphere using the Mississippi transformation. This youngest-positive transformation we denote by a $U(\lambda,\theta,\phi)$, and the quantum state of the ith subgenerator after passing through the youngest-positive operation evolves as: $|0\rangle$

$$|\psi_i\rangle = U_L(\lambda_i,\theta_i,\phi_i)|L\rangle \tag{8}$$

In general, here we choose the measurement operator $M = (|0\rangle\langle 0|)^{\otimes A}$ for the projection measurements, so that the resultant quantum state $|\psi_v\rangle$ after tracking the Ancilla quantum bit is:

$$|\psi_U\rangle = \mathrm{Tr}_A\left(\frac{(|0\rangle\langle 0|)^{\otimes A}\otimes \mathrm{I}|\psi_i\rangle\langle\psi_i|}{\langle\psi_i|(|0\rangle\langle 0|)^{\otimes A}\otimes \mathrm{I}|\psi_i\rangle}\right) \tag{9}$$

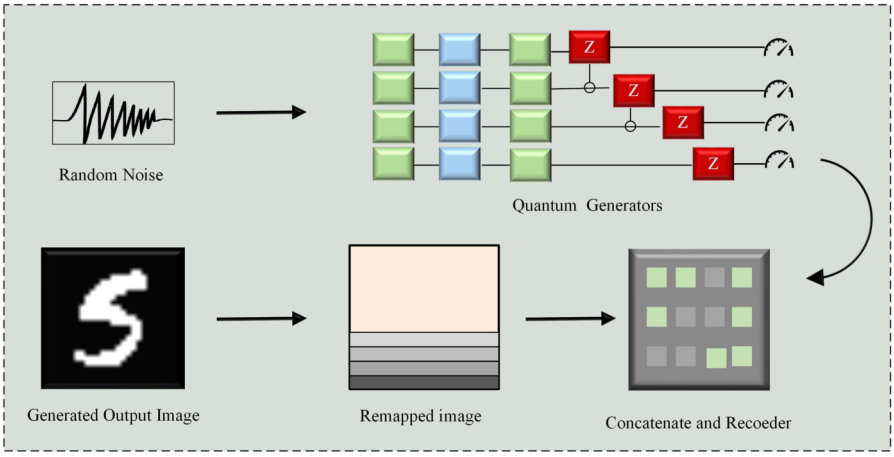

Figure 7 The working process of the generator of CAQ

This probability Pi is interpreted as the pixel value of the target image, and we form the target image by listing all the outputs of all the sub-generators together. Its working process is shown in Fig. 7.

3.4 Structure of discriminator

In generative adversarial learning, which is usually the case when NISQ devices and classical computers are working together, the critic we chose for classical neural networks for CAQ meets the requirements and is more suitable for NISQ devices. The critic here is the same as in the classical WGAN-GP; the critic receives the image output from the generator and produces an estimate of the Wasserstein distance measuring the difference between the generated distribution and the true distribution, contributing to a more stable training process. In addition, the gradient penalty (GP) introduces an additional loss that limits the gradient paradigm of the criteric, specifically, for each generator output, its gradient is computed and its paradigm is forced to approximate 1. This helps prevent gradient explosion and gradient vanishing [6], and motivates smoother learning in the criteric. However, quantum circuits still have some limitations; loading high-dimensional data into quantum circuits consumes a lot of quantum resources, which is difficult to do in practice. Secondly, the learning process of quantum circuits is more difficult to understand compared to classical neural networks, and there are problems such as barren plateaus [30, 31] in quantum circuits, for which there is still no complete set of solutions. Unlike other classical discriminators, the critic in CAQ has a nonlinear terminal layer with multiple activation functions (e.g., RELU, Sigmoid function, etc.), which enables the network to learn more complex distributions. At the critic end as a classifier outputs its discriminator, i.e., real data or false data, represented by a value. The whole working process of generator and discriminator is shown in Fig. 8.

4 CAQ algorithm model implementation and training

4.1 Parameters and training details

The CAQ employs parameters aligned with the classical WGAN-GP learning process, optimized using the ADAM optimizer, with both generator and critic learning rates set

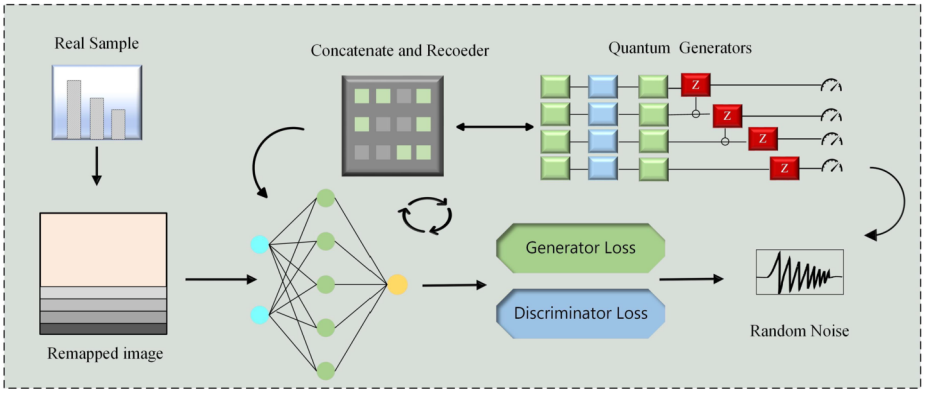

Figure 8 The whole working process of generator and discriminator of CAQ

Table 1 Summary of CAQ Model Parameters and Settings

Parameter/Setting	Description
Learning Rate (Generator)	0.01
Learning Rate (Critic)	0.0002
Regularization Parameter (λ)	10
Number of Classes (n_C)	5
Optimization Algorithm	Adam , $\beta_1 = 0$, $\beta_2 = 0.9$
Datasets Used	MNIST and Fashion MNIST
Image Resolution	28×28 grayscale images
Sample Size for Training	800 samples per category
Batch Size	25
Total Training Iterations	600 (equivalent to 3000 batches)
Epochs (2-Class)	37.5
Epochs (3-Class)	25
Implementation Libraries	PyTorch (classical ML), PennyLane (quantum ML)
Discriminator Architecture	Fully connected neural network with 2 hidden layers (ReLU)
Output Layer Neurons	Number of neurons equal to activated pixels
Quantum Bits Utilized	4 for MNIST (including 1 Ancilla); 5 for Fashion MNIST
Prior Distribution	Uniform prior in [0, 1) for enhanced learning
Adaptive Noise Reloading	Automatically adjusts noise input during training
Quantum Circuit Parameters	
Parameter Initialization	Random initialization within specified ranges
Optimization Algorithm	ADAM optimizer for efficient convergence
Hyperparameter Settings	Fine-tuned based on performance metrics

to 0.0002, chosen after extensive training sessions. Each generator utilizes measured Ancilla quantum bits to produce multiple patches from several sub-generators, each representing a row of pixels in the generated image. We find that the Gaussian prior distribution effectively restricts the range to $[-\pi, \pi)$, enhancing the quantum generators' learning capability; thus, we adopt it for our experiments, with plans to explore alternative prior distributions in future work. An adaptive noise reloading system adjusts the noise input range automatically based on generator and critic losses during each training iteration. A detailed description about the initialization of quantum circuit parameters, optimization algorithm selection and hyperparameter setting is made and a table is listed in Table 1.

4.2 Training strategies and resource consumption

To train the CAQ, we implement the WGAN-GP strategy, where the difference between generated and true distributions is quantified by the Wasserstein distance. Unlike tradi-

tional WGAN-GP, our generator output stems from multiple sub-generators, necessitating distinct parameter calculations for each sub-generator relative to a given loss function. We denote the parameters of the generator and critic as θ and λ, respectively, and express the combined loss function as $f(\theta, \lambda)$.

For parameter updates, we utilize the gradient descent algorithm, traditionally used in neural networks, to derive gradients of the loss function. However, in quantum circuits, the gradient information is complex and often treated as a black box. To compute partial derivatives within variational quantum circuits, we apply the parameter shift rule, modifying the parameters of the PQC twice. The partial derivative is approximated by:

$$\frac{\partial L}{\partial \theta} \approx \frac{L(\theta + \delta) - L(\theta - \delta)}{c} \tag{10}$$

Here, δ represents a small offset (bounded by $\frac{\pi}{2}$) applied to circuit parameters, while c depends on the specific quantum gate. The change in loss function values before and after the shift allows us to estimate the gradient for the PQC within the same quantum circuit. Denoting the total parameters of our generator as n, we express the gradient of the calculated parameter j of the i-th sub-generator with respect to the loss function $f(\theta)$ accordingly.

$$\begin{aligned} \frac{\partial f(\theta, \lambda)}{\partial \theta_{i,j}} = & \frac{1}{2} \langle f(\left[\theta_{1,1}, \ldots, \theta_{i,j} + \frac{\pi}{2}, \ldots, \theta_n\right], \lambda) \rangle \\ & - \langle f(\left[\theta_{1,1}, \ldots, \theta_{i,j} - \frac{\pi}{2}, \ldots, \theta_n\right], \lambda) \rangle \end{aligned} \tag{11}$$

In the CAQ model, the use of quantum resources was successfully balanced by optimizing the number of quantum bits, the depth of quantum circuits and the number of operations. In the experiments on the MNIST dataset, four quantum bits (including one Ancilla quantum bit) were used to assist in error correction and optimization of the generated images. In the more complex Fashion-MNIST dataset, the number of quantum bits was increased to 5 to better cope with the complexity of the dataset. While increasing quantum bits enhances the expressive power of the model, it also increases quantum resource consumption and operational complexity. Therefore, the choice of quantum bits needs to be a trade-off between computational power and resource consumption.

CAQ experiments show that 4-5 quantum bits are the best choice for generating 28×28 pixel grayscale images, which can effectively generate high-quality images. To reduce the circuit depth and ensure the generation quality, the CAQ model uses a simplified rotating layer and entanglement operation. In the simplified circuit, only the necessary R_x and R_y operations are retained, reducing the number of parameters and circuit depth. The CZ gate is chosen as the main entanglement operation, as it exhibits higher entanglement capability and lower hardware overhead during training, and is superior in performance and resource consumption compared to the classical CNOT gate. Ultimately, the use of five quantum bits and CZ operations in CAQ significantly reduces the number of operations and resource consumption while ensuring the generation quality, improves the maneuverability of the model on Noise-Intermediate Scale Quantum (NISQ) devices, and provides a reliable solution for quantum resource management of complex tasks in the future.

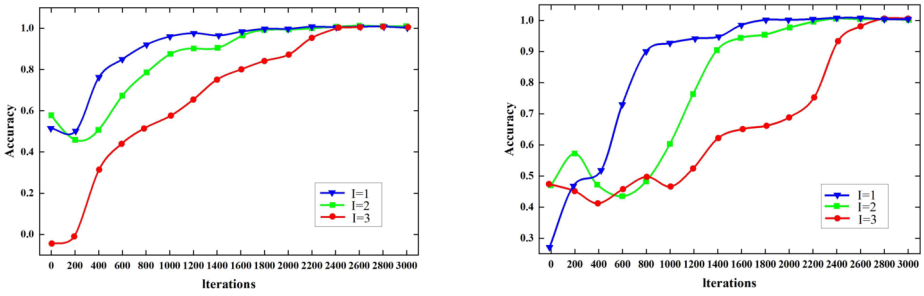

Figure 9 Accuracy of different layers of the generator

5 Simulation analysis

5.1 Comparison of simplified generators

In order to validate the advantage of simplified parameterized quantum circuits in the ability to generate images, we compared them with classical quantum circuits without simplification under two datasets (MNIST, FMNIST). Here the 28×28 generated image size scheme trained and used for each of the two different quantum circuit models.

In the MNIST handwritten digit generation task we use the subgenerator consisting of eight layers, each layer consisting of five quantum bits, one of which is an Ancilla quantum bit. In the apparel category generation task from FMNIST, due to the increased complexity of the dataset, the subgenerator consists of ten layers, each consisting of seven quantum bits, one of which is an Ancilla quantum bit.

We set the rotational layer of the unsimplified quantum circuit as $U_R = \prod_{i=1}^{n} R_y(\theta_{l,3}^i) \times R_x(\theta_{l,2}^i) R_y(\theta_{l,1}^i)$, and the rotational layer of the simplified circuit as $U_R = \prod_{i=1}^{n} R_y(\theta_{l,3}^i) R_x(\theta_{l,2}^i)$. The parameters are reduced by 3/1 compared to the previous ones, and for the two-bit quantum gate for the entanglement function, the classical CNOT gate is used in the unsimplified circuit, and as a comparison, the CZ gate is used in our circuit. We compared the results of the two sets of experiments in terms of two metrics, accuracy and KL (Kullback-Leibler) scatter [24]. The results produced are shown in Fig. 9,10 as a function of the number of iterative training sessions.

As expected, the accuracy of the simplified quantum circuit components used by CAQ is also maximized more quickly, and the convergence maxima of the accuracy of the two quantum circuits are not significantly different, indicating that the simplified circuits do not underperform the complex circuits in generating images in the generator. Moreover, due to the reduction of the rotational layer, the cost of the hardware used is greatly reduced.

5.2 Comparison with SOTA's quantum GANS

In this subsection, we compare images generated by CAQ with those from state-of-the-art (SOTA) quantum GANs, specifically evaluating their visual quality and detail on the MNIST and FMNIST datasets. The results are presented in Fig. 11.

First, we simulated HQCGAN [40] to generate grayscale images. However, for complex image shapes, particularly in the FMNIST dataset, HQCGAN's output often falls short, making it challenging to distinguish between garments and shoes. We then compared CAQ with QGPatch [16], the first quantum adversarial network to implement a patch

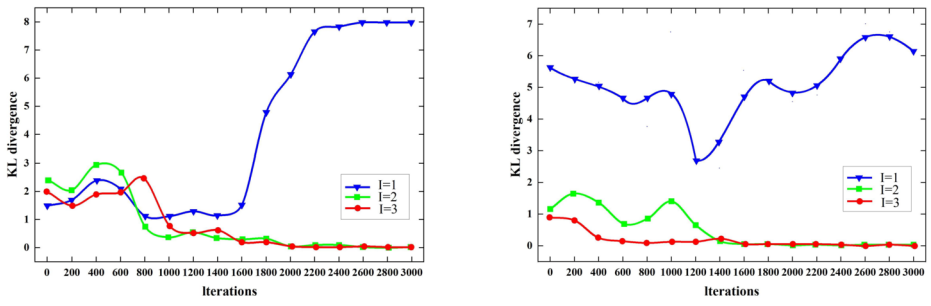

Figure 10 KL of different layers of the generator

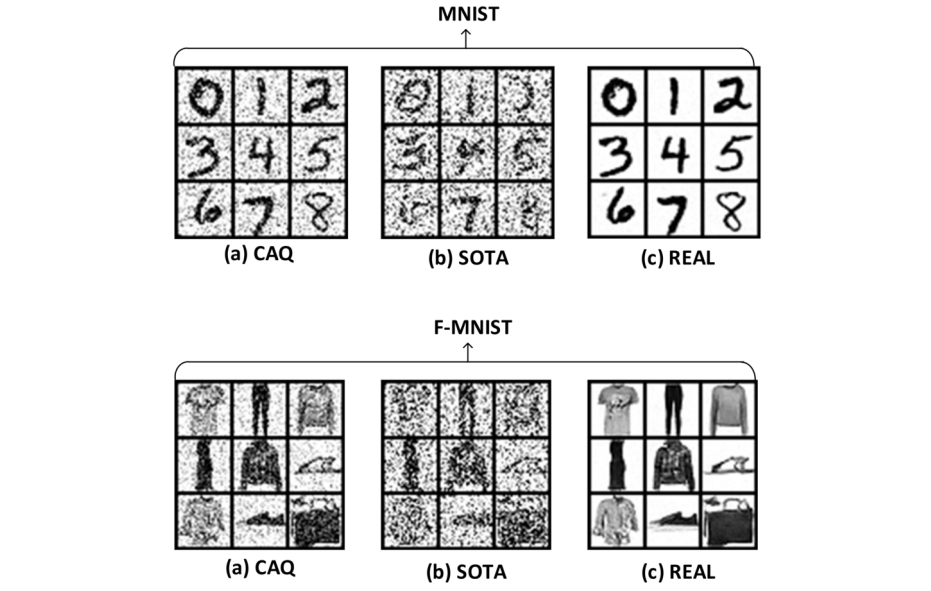

Figure 11 CAQ can generate high quality images on MNIST and FMNIST compared to real samples and SOTA's

strategy. While QGPatch effectively generates clear handwritten digits, its performance on FMNIST remains inadequate, with indistinct images.

Both HQCGAN and QGPatch suffer from quantum gate errors due to circuit depth, which diminishes their generation capabilities. In contrast, CAQ's simplified circuit design yields a notable improvement in image quality. CAQ-generated handwritten digits closely resemble real samples, and although dresses and shoes in the FMNIST dataset are clearly distinguishable, a gap with real images persists, highlighting the challenges quantum generators face with complex samples.

We attribute the high quality of images generated by CAQ to two key designs: first, we implemented an auto-adaptive noise reloading system to enhance the training of the generator; second, we pre-processed the target image using a remapping method. In contrast to models like HQCGAN and QGPatch, which rely on a fixed noise range, our approach dynamically adjusts the noise levels in response to environmental conditions. This adaptability effectively addresses the issue of mode collapse commonly encountered in generative adversarial networks.

The use of an automatically adapted noise range allows the generator to better accommodate environmental influences, thereby enhancing the stability of the generated images. This method enables the model to fully learn the details of the image feature distribution and to recognize variations across different distributions, resulting in greater diversity in the generated outputs. The adjustable noise range facilitates training in more optimal conditions, ultimately improving the quality and stability of the generated images.

However, quantum generators may still produce errors or defects in the generation of certain complex images, especially when dealing with high complexity datasets such as FMNIST. The generator may respond to complex contours or details with blurred or inaccurate images due to the accumulation of noise caused by the increased depth of quantum gates, which affects the clarity of the image. For example, the boundaries of footwear and apparel may not be clearly distinguishable, indicating that the optimization of quantum circuit parameters for depth and entanglement operations has not yet reached an ideal state.

To further improve these generation defects in the future, firstly, the effect of noise accumulation can be reduced by optimizing the depth and gate operation of quantum circuits. Second, the tuning of the number of quantum bits and entanglement operations can also help the generator to better capture image features, especially in higher dimensional image space. In addition, the introduction of more post-processing techniques, e.g., through further optimization of quantum states, can reduce the computational burden of the generator, resulting in clearer and more accurate images.

5.3 Comparison with classic WGAN

In this subsection, CAQ uses a simplified generator structure and an automatic noise reloading system, as well as adding a pre-processing session for remapping, and as a comparison, the classical WGAN-GP model is chosen, and still the generative task is evaluated on two datasets, MNIST and FMNIST, and the generated images are shown in Fig. 12. For both classical and quantum versions the target image was successfully generated, but the images generated in both cases still have some defects.

For MNIST, the images generated by CAQ exhibited vignetting and occasional superposition of digits, attributed to the limited training cycles due to quantum bit constraints. While CAQ trained for only a few hundred iterations, classical WGAN-GP underwent tens of thousands of iterations, resulting in clearer outputs. Despite this, CAQ demonstrated potential in learning key features with significantly fewer parameters. However, the quantum-generated images showed increased blurriness due to non-deterministic pixel values derived from Ancilla qubits, whereas the classical model produced sharper images overall. The Wasserstein distance curve indicated that as the number of iterations increased, both models showed gradual image quality improvement, with classical WGAN-GP achieving better clarity.

For FMNIST, the more complex dataset, both models were able to generate images, though defects such as blurriness and vignetting were more pronounced due to the higher complexity. Interestingly, CAQ showed superior detail representation for certain categories, such as pants and dresses, hinting at quantum advantages in capturing underlying feature spaces. The Wasserstein distance progression was slower for the FMNIST dataset, especially for CAQ, indicating that more iterations and training are needed for better results in complex tasks.

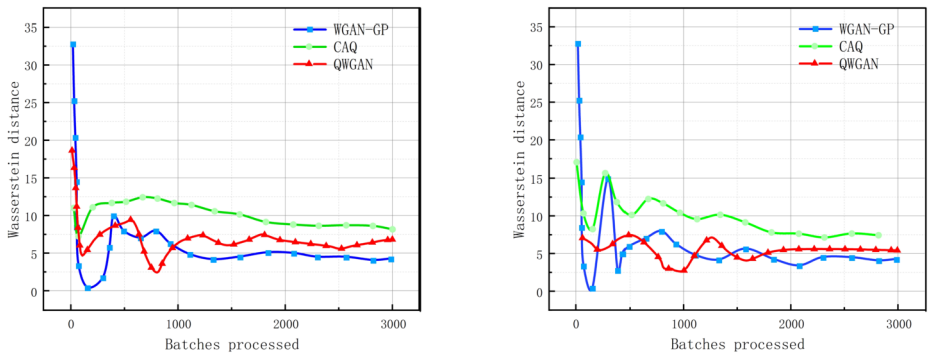

Figure 12 The WGAN-GP, CAQ, and QGAN models show a gradual smoothing of Wasserstein's distance as the number of trainings increases. CAQ achieves a significantly higher threshold than the other models

6 Conclusions and outlook

The advancement of quantum Generative Adversarial Networks (GANs) is remarkable. While they currently lag behind classical GANs, quantum computing harbors substantial computational potential. Our proposed CAQ represents a significant step in this field. We introduce a simplified quantum generator with fewer layers and parameters, maintaining performance by employing multiple sub-generators alongside a patching strategy for collaborative image generation. To enhance generator stability, we implement a tunable adaptive noise reloading system, improving interactions between the quantum generator and its elements.

Recent progress in quantum networking and quantum Internet infrastructure provides a promising foundation for distributed quantum machine learning systems, enabling low-latency, entangled communication between quantum devices across distances [42, 43]. This technological leap may soon support scalable training and deployment of quantum GANs in networked quantum environments. Inspired by the classical Wasserstein GAN (WGAN), our model, comprising a generator and a critic, utilizes the WGAN-GP strategy to optimize the quantum generator through Wasserstein distance while remapping images to learn distributions. We successfully trained on the MNIST and FMNIST datasets, demonstrating that our CAQ outperforms existing state-of-the-art methods.

Our current critic is based on classical neural networks, but future work could explore quantum discriminators to establish a fully quantum GAN framework. Such a framework may be more adept at modeling complex data and capturing intrinsic features. As quantum machine learning theory progresses, we aim to develop more suitable circuit designs and training methodologies, enhancing image generation capabilities. Additionally, advancements in quantum resources will facilitate the processing of larger datasets, enabling exploration of color images and intricate details, with the goal of generalization to real-world applications.

Author contributions

Yanbing Tian and Cewen Tian wrote the main text of the manuscript, while Zaixu Fan and Minghao Fu prepared all the information and data Hongyang Ma coordinated the whole work. All authors reviewed the manuscript.

Funding information

This work is supported by the Key Program of Shandong Province, China, Project (Grant Nos.2023CXGC010901), Joint Fund of Natural Science Foundation of Shandong Province (Grant Nos.ZR2021LLZ001).

Data availability
No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare no competing interests.

Author details
[1]School of Information and Control Engineering, Qingdao University of Technology, Qingdao, China. [2]School of Science, Qingdao University of Technology, Qingdao, China.

Received: 19 October 2024 Accepted: 21 May 2025 Published online: 23 June 2025

References

1. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets. Adv Neural Inf Process Syst. 2014;27. https://doi.org/10.48550/arXiv.1406.2661.
2. Arjovsky M, Bottou L. Towards principled methods for training generative adversarial networks. 2017. arXiv preprint. arXiv:1701.04862. https://doi.org/10.48550/arXiv.1701.04862.
3. Wang Y, Xu S, Liu J, et al. MFIF-GAN: a new generative adversarial network for multi-focus image fusion. Signal Process Image Commun. 2021;96:116295. https://doi.org/10.1016/j.image.2021.116295.
4. Akata Z, Reed S, Walter D, et al. Evaluation of output embeddings for fine-grained image classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2015. p. 2927–2936. https://doi.org/10.1109/cvpr.2015.7298911.
5. Castrejon L, Aytar Y, Vondrick C, et al. Learning aligned cross-modal representations from weakly aligned data. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. p. 2940–2949. https://doi.org/10.1109/cvpr.2016.321.
6. Huang HL, Du Y, Gong M, et al. Experimental quantum generative adversarial networks for image generation. Phys Rev Appl. 2021;16(2):024051. https://doi.org/10.1103/PhysRevApplied.16.024051.
7. Stein SA, Baheri B, Chen D, et al. Qugan: a quantum state fidelity based generative adversarial network. In: 2021 IEEE international conference on quantum computing and engineering (QCE). New York: IEEE Press; 2021. p. 71–81. https://doi.org/10.1109/qce52317.2021.00023.
8. Chu C, Skipper G, Swany M, et al. Iqgan: robust quantum generative adversarial network for image synthesis on nisq devices. In: ICASSP 2023-2023 IEEE international conference on acoustics, speech and signal processing (ICASSP). New York: IEEE Press; 2023. p. 1–5. https://doi.org/10.1109/icassp49357.2023.10096772.
9. Biamonte J, Wittek P, Pancotti N, et al. Quantum machine learning. Nature. 2017;549(7671):195–202. https://doi.org/10.1038/nature23474.
10. Bennett KP, Parrado-Hernández E. The interplay of optimization and machine learning research. J Mach Learn Res. 2006;7:1265–81. https://doi.org/10.1287/orms.2023.02.02.
11. Gyongyosi L, Imre S. Training optimization for gate-model quantum neural networks. Sci Rep. 2019;9(1):12679. https://doi.org/10.1038/s41598-019-48892-w.
12. Nielsen MA, Chuang IL. Quantum computation and quantum information. Cambridge: Cambridge University Press; 2000.
13. Ferrie C. Quantum model averaging. Nat Phys. 2014;10(5):259–61. https://doi.org/10.1038/nphys2929.
14. Kingma DP, Adam BJ. A method for stochastic optimization. 2014. arXiv preprint. arXiv:1412.6980. https://doi.org/10.48550/arXiv.1412.6980.
15. Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks. 2015. arXiv preprint. arXiv:1511.06434. https://doi.org/10.48550/arXiv.1511.06434.
16. Schuld M, Sinayskiy I, Petruccione F. An introduction to quantum machine learning. Contemp Phys. 2015;56(2):172–85. https://doi.org/10.1080/00107514.2014.964942.
17. Briegel HJ, De las Cuevas G. Projective simulation for artificial intelligence. Sci Rep. 2012;2(1):1–7. https://doi.org/10.1038/srep00400.
18. Grant E, Benedetti M, Cao S, et al. Hierarchical quantum classifiers. npj Quantum Inf. 2018;4(1):1–8. https://doi.org/10.1038/s41534-018-0116-9.
19. Havlíček V, Córcoles AD, Temme K, et al. Supervised learning with quantum-enhanced feature spaces. Nature. 2019;567(7747):209–12. https://doi.org/10.1038/s41586-019-0980-2.
20. Zeng W, Johri S, Landsman KA, et al. Learning and inference via quantum generative models. Phys Rev Lett. 2020;125(17):170501. https://doi.org/10.1103/PhysRevLett.125.170501.
21. Lloyd S, Mohseni M, Rebentrost P. Quantum algorithms for supervised and unsupervised machine learning. 2013. arXiv preprint. arXiv:1307.0411. https://doi.org/10.48550/arXiv.1307.0411.
22. Verma M, Singh AK. Quantum Generative Adversarial Networks (QGANs): a Tutorial. 2020. arXiv preprint. arXiv:2009.10102. https://doi.org/10.48550/arXiv.2009.10102.
23. Cao Y, Zhu Y, Li C, et al. QGAN: quantum generative adversarial networks. Quantum Inf Process. 2020;19(4):1–14. https://doi.org/10.1007/s11128-020-2628-6.

24. Hoffman R, Boixo S, et al. Quantum generative adversarial networks. Quantum Sci Technol. 2020;5(3):034001. https://doi.org/10.1088/2058-9565/ab8c81.
25. Pang S, Yang Y, et al. Quantum GAN for generative modeling. J Quantum Inf Sci. 2020;10(4):123–36. https://doi.org/10.3390/jqis10040007.
26. Zhang S, Wu J, et al. Quantum adversarial networks. Quantum Mach Learn. 2022. https://doi.org/10.48550/arXiv.2210.07287.
27. Khan S, Bashir A, et al. Hybrid Quantum-Classical Generative Adversarial Network. 2022. arXiv preprint. arXiv:2211.04674. https://doi.org/10.48550/arXiv.2211.04674.
28. Zhou Y, Zhu C, et al. Quantum image generation using QGAN. Quantum Inf Process. 2023;22(1):1–15. https://doi.org/10.1007/s11128-023-3385-0.
29. Sullivan M, Wang C, et al. Improving QGAN Performance with Classical Strategies. 2023. arXiv preprint. arXiv:2302.02315. https://doi.org/10.48550/arXiv.2302.02315.
30. He Y, Wu D, et al. A hybrid quantum-classical framework for QGANs. IEEE Trans Quantum Eng. 2023. https://doi.org/10.1109/TQE.2023.3242783.
31. Li Y, Zhao X, et al. QGANs for Multi-modal Data Generation. 2023. arXiv preprint. arXiv:2305.01234. https://doi.org/10.48550/arXiv.2305.01234.
32. Chen G, Wang Y, et al. Quantum Generative Adversarial Networks with Informed Feedback. Quantum Sci Technol. 2024. https://doi.org/10.1088/2058-9565/acad1d.
33. Zhang L, Liu Y, et al. Adversarial Learning on Quantum States. 2023. arXiv preprint. arXiv:2309.05599. https://doi.org/10.48550/arXiv.2309.05599.
34. Khan S, Chang Y, et al. Variational quantum generative adversarial networks. Quantum Inf Process. 2024;23(3):1–15. https://doi.org/10.1007/s11128-024-01478-3.
35. Gyongyosi L, Imre S. Scalable distributed gate-model quantum computers. Sci Rep. 2021;11(1):5172. https://doi.org/10.1038/s41598-020-76728-5.
36. Gyongyosi L, Imre S. Circuit depth reduction for gate-model quantum computers. Sci Rep. 2020;10(1):11229. https://doi.org/10.1038/s41598-020-67014-5.
37. Xiao Y, Li W, et al. Quantum image processing using GANs. Quantum Inf Process. 2024;23(5):1–17. https://doi.org/10.1007/s11128-024-01501-5.
38. Yang S, Wang Z, et al. Enhancing Quantum GANs through Classical Post-processing. 2023. arXiv preprint. arXiv:2307.03456. https://doi.org/10.48550/arXiv.2307.03456.
39. Feng Y, Li P, et al. A review of quantum GANs: challenges and solutions. Quantum Mach Learn. 2024;2(1):1–20. https://doi.org/10.3390/qml2010001.
40. Zhou X, Chen L, et al. Quantum generative adversarial networks for text generation. Quantum Inf Process. 2024;23(4):1–25. https://doi.org/10.1007/s11128-024-01540-0.
41. Silver D, Patel T, Cutler W, et al. Mosaiq: quantum generative adversarial networks for image generation on nisq computers. In: Proceedings of the IEEE/CVF international conference on computer vision. 2023. p. 7030–9. https://doi.org/10.1109/iccv51070.2023.00647.
42. Gyongyosi L, Imre S. Advances in the quantum Internet. Commun ACM. 2022;65(8):52–63. https://doi.org/10.1145/3524455.
43. Gyongyosi L, Imre S, Nguyen HV. A survey on quantum channel capacities. IEEE Commun Surv Tutor. 2018;20(2):1149–205. https://doi.org/10.1109/comst.2017.2786748.

Publisher's note