SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

January 2026, Vol. 69, Iss. 1, 112501:1–112501:16
https://doi.org/10.1007/s11432-024-4475-4

AI-generated image detection algorithm based on classical-quantum hybrid neural network

Juncong XU[1,2†], Han FANG[3†], Yang YANG[1,2*], Kejiang CHEN[4], Zhaoyun CHEN[2],
Menghan DOU[6], Lei QU[1,7], Weiming ZHANG[4] & Guoping GUO[5]

[1] *School of Electronic and Information Engineering, Anhui University, Hefei 230039, China*
[2] *Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei 230088, China*
[3] *School of Computing, National University of Singapore, Singapore 119077, Singapore*
[4] *Anhui Province Key Laboratory of Digital Security, University of Science and Technology of China, Hefei 230026, China*
[5] *Laboratory of Quantum Information, School of Physics, University of Science and Technology of China, Hefei 230026, China*
[6] *Origin Quantum Computing Co., Ltd., Hefei 230026, China*
[7] *Hefei National Laboratory, University of Science and Technology of China, Hefei 230026, China*

Received 22 October 2024/Revised 31 January 2025/Accepted 4 June 2025/Published online 4 January 2026

Abstract The advance in generative artificial intelligence (AI) has led to increasingly realistic image synthesis, which from a forensic perspective, also spawned the demand for AI-generated image detections. Achieving detection generalizability across various deep generative models is crucial as different deep generative models are emerging. The key to achieving generalizability is to design better architectures for feature extraction and representation. Noteworthy, quantum neural networks (QNN), due to their larger representation space and inherent parallelism, have already shown their superior performance in feature extraction. Therefore, in this paper, we introduce a delicately designed QNN and combine it with the classical Swin Transformer V2, proposing an AI-generated image detection algorithm based on a classical-quantum hybrid neural network. In order to adapt the features of the Swin transformer and fully utilize the entanglement characteristics of QNN, we proposed an alternating layered ansatz (ALT)-based QNN, which successfully provides high trainability and rich expressibility. Extensive experiments on various generative model datasets with different training sample sizes indicate that with a small number of training samples, the hybrid neural network can achieve strong generalization, where the overall detecting accuracy is 2% higher than classical neural network. In addition, we verified the feasibility of a hybrid neural network on the "Wukong" 72-qubit superconducting quantum computer provided by OriginQ, showing similar accuracy to classical computer simulations, demonstrating that our model is feasible and effective in actual quantum computing systems.

Keywords AI-generated image detection, quantum neural network, quantum computing, generative adversarial network

Citation Xu J C, Fang H, Yang Y, et al. AI-generated image detection algorithm based on classical-quantum hybrid neural network. Sci China Inf Sci, 2026, 69(1): 112501, https://doi.org/10.1007/s11432-024-4475-4

1 Introduction

In recent years, significant progress has been made in the field of generative artificial intelligence (AI). By learning complex data distributions, the advanced generative AI models can produce high-quality images with perfect details, opening up new possibilities in numerous application areas. However, the other side of the coin creates the potential threat of disinformation. To address such issues, the passive detection of AI-generated images is widely studied [1–7].

The rapid evolution in generative AI models results in various model architectures and corresponding diverse image sources. Therefore, designing a classifier capable of extracting general discriminative features is the current focal point. Currently, AI-generated image detection algorithms can be divided into two categories based on the field of feature extraction, namely frequency domain-based feature extraction and spatial domain-based feature extraction. Among them, methods based on frequency domain feature extraction typically convert images into spectrograms, and then train classifiers based on these spectral features for detection [1, 3]. On the other hand, early methods based on spatial domain features detect by designing some manual features, such as extracting differences in saturation and color components [8, 9]. In order to cope with the increasing difficulty of detection,

[*] Corresponding author (email: sky_yang@ahu.edu.cn)
[†] These authors contributed equally to this work.

researchers have begun to turn to deep learning-based detection research. For example, Gragnaniello et al. [6] and Ju et al. [7] designed better feature extraction networks to improve the generalization ability of detection. Although the motivations and ideas of these methods are different from each other, they have a common core idea: to design a classifier that can extract universal discriminative features.

The feature extraction ability of a neuron determines the generalizability of the network. In classical neural networks, each neuron in the network exists in only one state, and its space is equivalent to the data dimension. However, in quantum neural networks (QNN), an individual neuron can exhibit more than one state due to the characteristics of quantum superposition and entanglement. Therefore, in this paper, we investigate whether introducing QNN would be beneficial in improving generalizability.

Recently, research on QNNs has revealed their unique advantages in representation space, model capacity, and generalization [10–12]. Schuld and Killoran [13] and Havlíček et al. [14] made almost simultaneous discoveries that, akin to the kernel methods in machine learning, mapping features into a higher-dimensional quantum Hilbert space can enhance classification. This effectiveness stems from the core advantage of quantum computing, which is its efficient exploration of an exponentially large quantum state space. Abbas et al. [15] employed tools from information geometry to define the expressive capacities of classical NN and QNN. Using Fisher information as a metric tool, they found that QNNs exhibit a higher effective dimension compared to classical NNs. In 2022, Caro et al. [16] provided theoretical bounds on the generalization error in variational quantum machine learning (QML) methods, showing that an efficiently implementable QML model only requires a polynomial-sized training dataset to achieve good generalization. They noted that training a QML model with N samples and T trainable parameters results in a worst-case generalization error of $\sqrt{T/N}$. This suggests that the QML model can achieve robust generalization with fewer training data.

While the current state of quantum computing is in the noisy intermediate-scale quantum (NISQ) era, where available quantum computers still exhibit noise and errors, and their scale remains relatively small. This directly limits the scale and depth of fully implemented QML. Given these constraints of QML in the NISQ era, a hybrid approach that combines QML with classical neural networks represents a more feasible choice today. In this context, neural networks and optimization methods based on the classical-quantum hybrid approach have become excellent carriers for exploiting quantum advantages [17, 18]. Among them, transfer learning from classical to quantum [19] combines the powerful feature extraction ability of classical pre-trained models with the high-dimensional feature expression space of QNN, further unlocking the application scenarios for exploiting quantum advantages. Thus, based on quantum transfer learning, we designed a classical-quantum hybrid neural network. Specifically, the algorithm first utilizes a pre-trained Swin Transformer V2 (Swin V2) [20] as a feature extractor to extract image features. Subsequently, these extracted classical features are transferred to an alternating layered ansatz [21] (ALT)-based QNN, facilitating efficient classification learning of feature data in the quantum Hilbert space. Furthermore, we conducted executions of our designed hybrid neural network on a real quantum computer, validating the feasibility of our model on near-term quantum devices.

In general, the contributions of this paper can be summarized in three main aspects.

(1) We are the first to introduce QNN in the field of AI-generated image detection. By targetedly devising the ALT-based QNN and combining it with a classical neural network, the advantages of QNN in this domain are effectively explored.

(2) To enhance the generalization of AI-generated image detection, we designed a QNN based on an alternating layered ansatz. This QNN entangles all qubits in an alternating layer-wise manner, enabling strong expressiveness with a relatively shallow circuit depth.

(3) The simulation experimental results indicate that our method can achieve strong generalization with a small number of training samples. Furthermore, the accuracy of our designed hybrid network on a real quantum computer is comparable to the simulation results, demonstrating the feasibility and effectiveness of the proposed method on actual quantum computing systems.

2 Related work

2.1 AI-generated image detection

Currently, AI-generated image detection methods can be broadly categorized into two main types based on the domain of their operation: those based on frequency features and those based on spatial features.

2.1.1 *Methods based on frequency features*

This type of methods involve extracting features in the frequency domain for detection and classification. For example, Zhang et al. [1] detected generative adversarial networks (GAN)-generated images based on the unique artifacts in the frequency spectrum produced during the upsampling process of the GAN model. Durall et al. [2] found that the transposed convolution operation of convolutional neural network (CNN) cannot replicate the frequency spectrum distribution of real images, and they detected CNN-generated images based on this spectral distortion. Frank et al. [3] conducted detection according to the fact that GAN-generated images show strong high-frequency components and grid-like patterns after undergoing the discrete Fourier transform. There are also some methods [22, 23] that extract the "fingerprints" of images in the frequency domain and carry out detection based on the differences in these "fingerprints" between real images and AI-generated images. To further enhance the cross-model generalization ability, Liu et al. [4] used a denoising network to extract the noise patterns of images and showed that real images have consistent and unique patterns in both the spatial and frequency domains, while generated images have periodic and irregular patterns.

2.1.2 *Methods based on spatial features*

This type of method involves extracting features in the spatial domain for detection and classification. For example, McCloskey and Albright [8, 9] utilized the differences in saturation and color components between AI-generated images and real images for detection. Chai et al. [24] used a patch-based classifier to identify detectable artifacts in fake images and demonstrated that even when the image generator is adversarially fine-tuned, detectable artifacts still remain in certain image patches. Wang et al. [5] achieved good cross-generative model generalization by using JPEG compression and Gaussian blur for data augmentation. And Gragnaniello et al. [6] found that downsampled images would lead to the loss of valuable features for distinguishing between real and generated images. Based on the method of Wang et al. [5], they removed the downsampling operation from ResNet-50 [25] and achieved higher classification accuracy. Ojha et al. [26] believed that deep neural networks trained on real/fake datasets will develop a tendency towards certain features of fake images in the training set. To solve this problem, they train the classifier in the CLIP:ViT [27] feature space with fixed parameters, effectively avoiding such bias and achieving impressive cross-model generalization ability.

2.2 Quantum machine learning

Recently, advancements in quantum computing have provided new approaches to solving some classical problems [28–31]. Some researchers are exploring whether current near-term quantum devices can become a new breakthrough in the field of deep learning. For classification problems, mapping features to higher-dimensional spaces often achieves better classification performance, and quantum Hilbert spaces are a good choice. Schuld et al. [13] and Havlíček et al. [14] found that mapping features into the quantum Hilbert space allows for the efficient exploration of quantum state space using quantum circuits, leading to enhanced classification performance. Mari et al. [19] integrated deep neural networks with QNN, introducing a transfer learning approach from classical to quantum. This method opens up the possibility of using any advanced deep neural network to classically pre-process large input samples, such as high-resolution images. It also enables efficient learning in the quantum Hilbert space through variational quantum circuits, providing a new tool for solving real-world problems with near-term quantum devices. Some researchers have explored the properties and advantages of QNN from a theoretical perspective. For example, Abbas et al. [15] found that QNN has a larger effective dimension compared to classical neural networks. Caro et al. [16] discovered that quantum machine learning models possess the ability to generalize well from fewer training samples. Larocca et al. [32] investigated the phenomenon of overparameterization in the QNN field. Li et al. [33] proposed a quantum self-attention neural network, exploring the advantages of quantum neural networks in text classification tasks. These studies have sparked widespread interest and discussion about quantum machine learning.

3 Method

For the task of AI-generated image detection, we devised a classical-quantum hybrid neural network (hybrid NN). The entire network can be broadly divided into two parts: a classical neural network for feature extraction and dimension reduction and a QNN for classification. In the classical model segment, we employed the Swin V2 model based on the transformer, which is known for its powerful feature extraction capability. This model effectively captures traces in the upsampling operations of the generative model caused by a lack of global information. In the

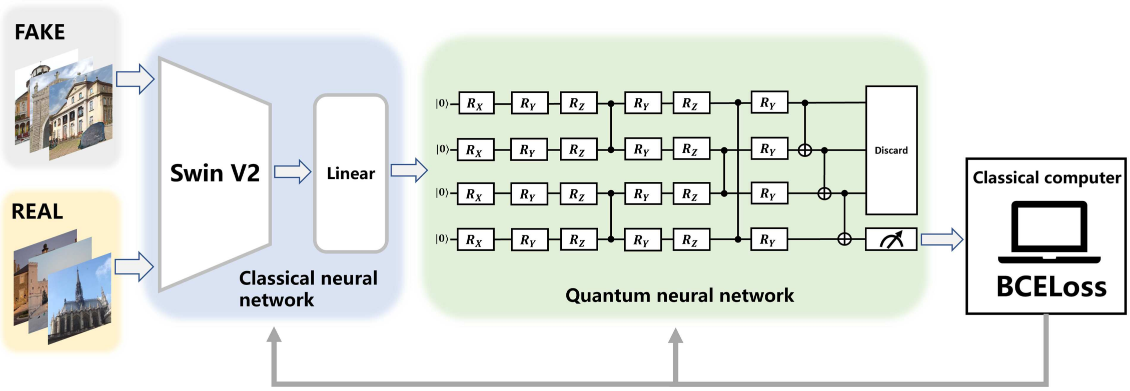

Figure 1 (Color online) The architecture of the proposed classical-quantum hybrid neural network. First, the real/false images are input into Swin V2 to extract 768-dimensional image features. Subsequently, these features are fed into a linear layer for further dimensionality reduction so as to adapt to the input dimension of the QNN. Finally, the 4-dimensional features after dimensionality reduction by the linear layer are encoded into quantum states, and the classification is completed in the quantum Hilbert space.

quantum model segment, we designed a 4-qubit QNN, efficiently leveraging the high-dimensional space of quantum states for classification learning. The framework of our designed hybrid NN is illustrated in Figure 1.

Specifically, the image data are initially fed into the pretrained Swin V2 model, which extracts 768-dimensional feature data. The linear layer then takes the feature data from Swin V2, further reducing the dimensionality from the original 768 dimensions to 4 dimensions for output. Subsequently, the QNN receives low-dimensional but information-rich feature data from the preceding layer. It encodes these features into quantum states and learns classification information in the quantum Hilbert space.

3.1 Classical neural network

The classical neural network consists mainly of two parts: the pretrained Swin V2 model and a linear layer used for dimension reduction. First, due to the upsampling operations in generative models lacking global information and leaving traces in images, we opted for the transformer-based Swin V2 pretrained network as a feature extractor. This network, based on a self-attention mechanism, possesses a larger receptive field than convolutional neural networks do, effectively extracting global features from images and capturing the global traces of image generation. Second, since the Swin V2 model outputs 768-dimensional feature data and since the current availability of qubits is limited, we employ a linear fully connected layer to further reduce the dimensionality of the feature data to accommodate the input of the QNN.

3.1.1 *Swin transformer V2 feature extraction network*

We use Swin V2 as the feature extraction network, aiming to leverage its powerful global feature extraction capabilities based on the self-attention mechanism. Its structure is illustrated in Figure 2. The goal is to extract the differences in global features between generated and real images. To extract more robust and general image features, Swin V2, built on the foundation of Swin V1 [34], increases the model capacity. Its effectiveness stems from a series of designs such as post-normalization, scaled cosine attention and log-spaced continuous position bias [20]. Based on these successful improvements, in this paper, we also applied Swin V2 as our backbone for feature extraction.

3.1.2 *Linear layer for further feature reduction*

This layer consists of a linear layer with an input dimension of 768 and an output dimension of 4, which is designed for further dimensionality reduction of features. In the current NISQ era, due to the limited scale of qubits and the existence of quantum noise, QNN allows only low dimensional inputs. However, the feature dimension extracted by Swin V2 is 768, which is still too high for the QNN input size. Therefore, we also utilize the linear layer for further feature reduction which decreases the feature dimension from 768 to 4.

3.2 Quantum neural network

The QNN is a novel neural network model based on the principles of quantum mechanics, that shares many similarities with classical NNs, such as having learnable parameters and requiring updates through gradient descent. However, there are also several differences between them. First, QNNs use quantum states to represent information

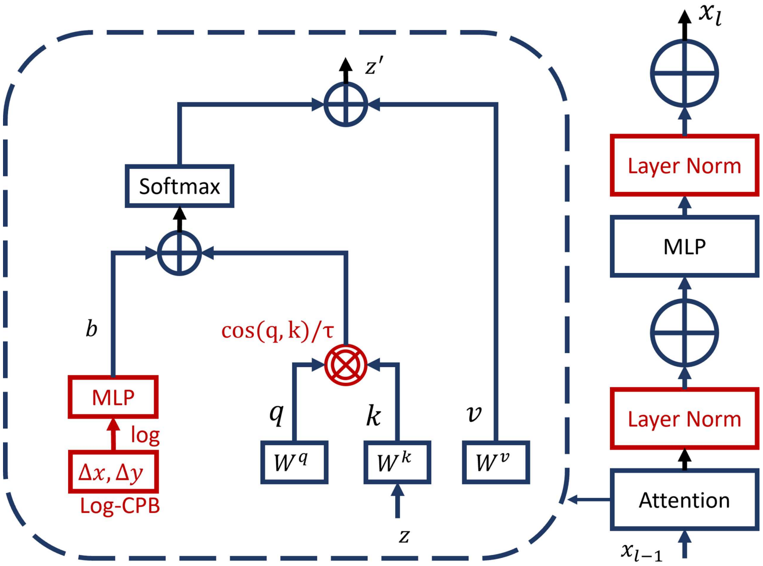

Figure 2 (Color online) The architecture of Swin Transformer V2. It has carried out a series of optimizations in comparison with Swin V1 to achieve better feature extraction. Firstly, via post-normalization, the activation amplitude becomes more moderate, enhancing the stability of model training. Secondly, by calculating pixels with scaled cosine attention, it has solved the problem that the attention maps of some modules and heads in large models are dominated by a small number of pixel pairs. Finally, the adoption of log-spaced continuous position bias (Log-CPB) makes the model more efficient when migrating across window resolutions.

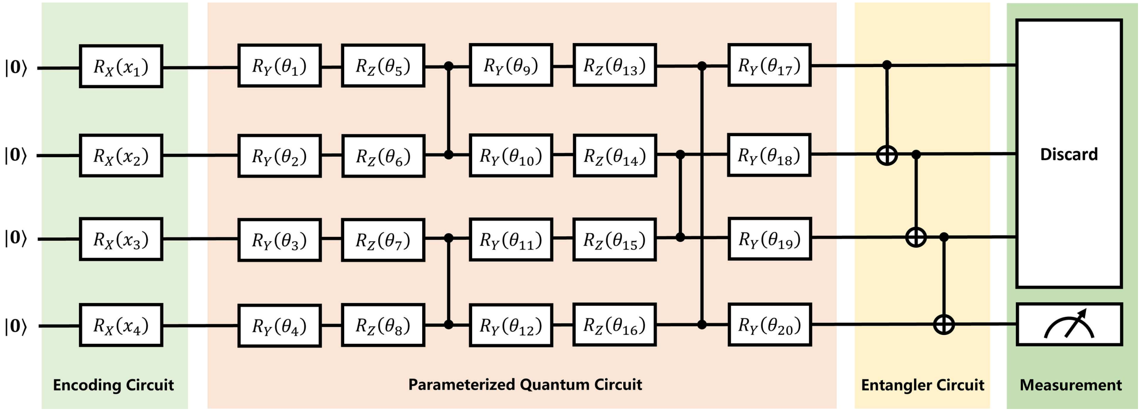

Figure 3 (Color online) The architecture of the QNN we designed. It can be divided into four parts in total, namely the encoding circuit, the parameterized quantum circuit, the entangler circuit, and the measurement module. Among them, the encoding circuit accomplishes the conversion from classical features to quantum states, and the parameterized quantum circuit is responsible for learning to map the features of different categories to separable regions in the Hilbert space. The entangler circuit is in charge of enhancing the degree of entanglement of the circuit to obtain more comprehensive measurement results. The measurement module is responsible for obtaining the final classification results from the quantum states.

in Hilbert space, rather than classical binary data. Second, the trainable parameters in a QNN are the rotation gate angles in the quantum circuit, not classical floating-point weights. Finally, the output values in a QNN are the measurement information of the qubits, not the classical neuron output values.

Based on these differences, QNNs possess a higher feature expression dimension and inherent unitary transformation properties. These characteristics make QNNs less likely to get trapped in extreme values during the training process compared to classical neural networks, thus potentially enabling better generalization. As pointed out in [16], QNN has the characteristic of obtaining good generalization from a small number of training samples. They regard QML models are regarded as parameterized quantum channels, that is, completely positive trace-preserving (CPTP) maps. By establishing bounds on the covering numbers of quantum operations and then using the chaining technique of random processes, they derive an upper bound on the generalization error of QNN. Specifically, for a QNN with T trainable parameters trained on N samples, the upper bound of the generalization error is $\sqrt{T/N}$. This theoretically proves that QNN can achieve good generalization from a small number of training samples. The QNN designed in this paper is shown in Figure 3.

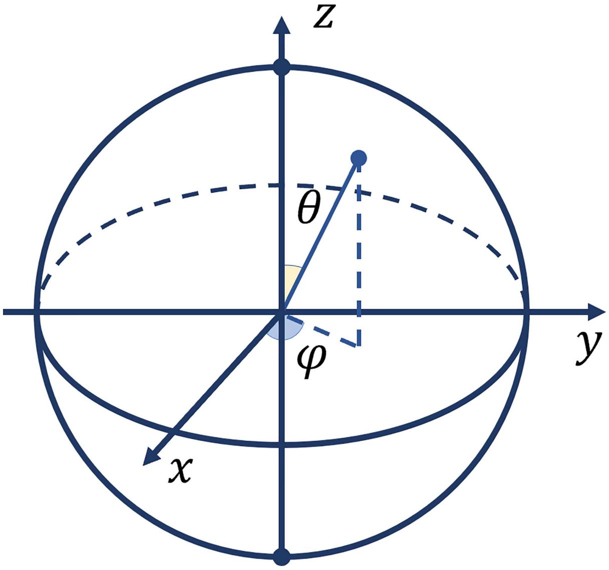

Figure 4 (Color online) The quantum Bloch sphere. It is used to visualize and describe the state of a single qubit, and it has three coordinate axes, namely x, y, and z. Meanwhile, the quantum rotation gates R_X, R_Y, and R_Z can control the quantum state to rotate around the x, y, and z axes, respectively.

3.2.1 *Encoding circuit*

The encoding circuit serves as a bridge connecting classical data and quantum states. In the field of quantum computing, prevalent encoding methods include basis encoding, amplitude encoding, and angle encoding. Considering that the QNN is positioned at the tail end of the overall model, during the training process, gradient information must pass through the QNN to reach the classical network. Given that only angle encoding among the three methods is differentiable, we opt for angle encoding to encode classical data into quantum states.

Due to the input being angle information for angle encoding, normalization is required before the encoding process. Initially, the input data $x = [x_1, x_2, x_3, \ldots, x_n]^{\mathrm{T}}$ is normalized to the range $(-1, 1)$ using the tanh function, and then multiplied by π to convert the values to angles ranging from $-\pi$ to π. This n-dimensional data serves as the angle for single-bit rotation gates, which are encoded into n qubits:

$$|x\rangle = \bigotimes_{k=1}^{n} R(x_k)\,|0^n\rangle\,, \tag{1}$$

where n represents the number of qubits, and R can be any of the Pauli matrices X, Y, and Z, i.e.,

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},\ Y = \begin{pmatrix} 0 & -\mathrm{i} \\ \mathrm{i} & 0 \end{pmatrix},\ Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{2}$$

and $R(x_k) = \mathrm{e}^{-\mathrm{i}Rx_k}$. The representation space for each qubit can be visualized using the Bloch sphere. R_X, R_Y, and R_Z enable the quantum state to rotate around the X, Y, and Z-axes of the Bloch sphere, respectively. Therefore, each feature data point is mapped to a point on the Bloch sphere, as depicted in Figure 4. In our specific implementation, we opted for the R_X gate to achieve the mapping of classical feature data to quantum states.

3.2.2 *Parameterized quantum circuit*

The encoding circuit accomplishes the transformation of classical features into quantum states, while the parameterized quantum circuit is responsible for learning to obtain the target quantum state. A typical parameterized quantum circuit consists of a series of rotation gates and control gates. The rotation angles of the quantum rotation gates are trainable parameters that are updated during training via gradient descent. Moreover, the control gates facilitate entanglement among different qubits, enhancing the expressive capability of the quantum circuit.

The parameterized quantum circuit serves as the core of QNN, and the ALT emerges as an efficient variant. Each layer of ALT contains multiple blocks, and the entangling gates within each block only act on local qubits. This local entanglement design makes the parameter updates within each block more independent, reducing the

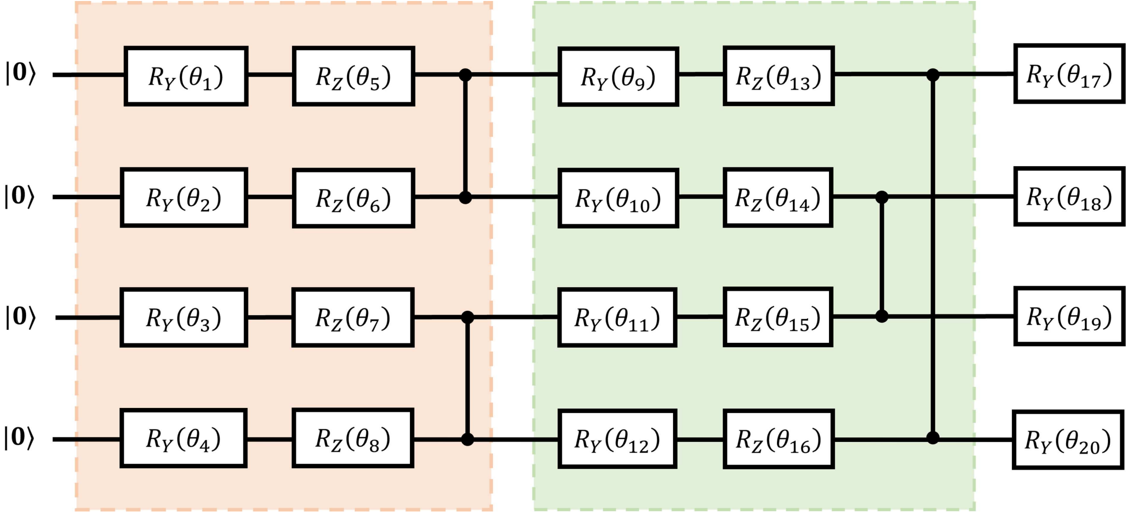

Figure 5 (Color online) The proposed parameterized quantum circuit. In this parameterized quantum circuit, the part with an orange background is the even layer, and the part with a green background is the odd layer. In each layer, the learnable parameters are composed of R_Y and R_Z gates. The even and odd layers complete the entanglement of the entire quantum system in a complementary way. Specifically, the even layer uses two CZ gates to achieve the entanglement between the 0th and 1st qubits and between the 2nd and 3rd qubits, respectively. The odd layer uses two CZ gates to achieve the entanglement between the 1st and 2nd qubits and between the 3rd and 0th qubits, respectively.

complexity and risk of gradient vanishing associated with global entanglement. Moreover, the block structures of the odd-numbered and even-numbered layers in ALT are different, entangling all qubits in an alternating manner. Such a design allows the circuit to have both strong expressiveness and high trainability. As in the research by Nakaji et al. [35], they conducted a comparative analysis of ALT and the hardware-efficient ansatz (HEA) through frame potential and Kullback-Leibler (KL) divergence and found that ALT has almost the same expressibility as HEA, but with better trainability. In addition, Cerezo et al. [21] derived a strict lower bound for the variance of the gradient of ALT under local loss functions, showing that ALT circuits with a number of layers l of $O(\text{poly}(\log n))$ can solve the problem of gradient vanishing.

Building on these theories, we targeted the challenge of AI-generated image detection by devising a parameterized quantum circuit based on ALT that combines high expressiveness with trainability, as depicted in Figure 5. In [21], the ALT circuit utilized only one type of rotation gate as the carrier for learnable parameters, limiting each qubit to rotate around a single axis on the Bloch sphere, thereby constraining the overall circuit's exploration space. To harness a larger quantum state space, we employed two types of quantum rotation gates as carriers for the quantum parameters. This approach allows each qubit to fully explore its representation space, effectively mapping the features of real and AI-generated images to distinct regions, and thus achieving better generalization.

In Figure 5, the entire circuit is segmented into odd and even layers, with the orange box representing the even layer and the green box representing the odd layer. Each even and odd layer consists of four R_Y gates, four R_Z gates, and two CZ gates, totaling eight trainable parameters. The CZ gates in odd layers and even layers entangle all the qubits in a complementary manner, ensuring that every pair of qubits is entangled across the layers. Additionally, at the end of the odd and even layers, a layer of R_Y gates is added, bringing the total trainable parameters for the entire parameterized quantum circuit to 20. Since the operations of the entire parameterized quantum circuit can be represented by a matrix $V(\theta)$ containing parameters, this part of the operation can be expressed as

$$|\varphi\rangle = V(\theta)\,|x\rangle\,. \tag{3}$$

3.2.3 *Entangler and measurement circuit*

In the context of AI-generated image detection, which is essentially a binary classification problem, the requirement is to measure the information of a specific qubit as the predicted output. It is crucial that the measured qubit captures as much information as possible from the entire quantum system, effectively representing the entire quantum system for predictive output. Quantum entanglement facilitates this objective. Therefore, before proceeding with the measurement output, a circuit entangling all qubits is employed to enhance the entanglement of the entire quantum system. As illustrated in Figure 3, the entangler circuit consists of three CNOT gates. This module achieves information transfer between adjacent qubits through CNOT, enabling the measured controlled qubit to provide a

Table 1 Composition of the test dataset.

	BigGAN	CycleGAN	GauGAN	ProGAN	StarGAN	StyleGAN	StyleGan2
Scene	1	6	1	20	1	3	4
Image source	ImageNet	Style/object transfer	COCO	LSUN	CelebA	LSUN	LSUN
# Images	4.0k	2.6k	10.0k	8.0k	4.0k	12.0k	16.0k

more comprehensive predictive output about the entire quantum system. The matrix representation of CNOT is

$$\mathrm{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{4}$$

Therefore, the entire entanglement circuit operation can be represented as

$$|\phi\rangle = (\mathrm{CNOT}_{2,3})(\mathrm{CNOT}_{1,2})(\mathrm{CNOT}_{0,1})\,|\varphi\rangle\,. \tag{5}$$

After passing through the entanglement circuit, the quantum system possesses increased entanglement and correlation. The measurement output is obtained by measuring the expectation of the final qubit with respect to the $\sigma_Z = \mathrm{diag}(1, -1)$ matrix:

$$O = \langle\phi_0|\sigma_Z|\phi_0\rangle. \tag{6}$$

Finally, it is necessary to calculate the model's loss based on the measurement output. We use binary cross-entropy as the loss function. Since the eigenvalues of the σ_Z matrix are -1 and 1, the measurement output O ranges from $[-1, 1]$. To normalize this to the range $(0, 1)$, we calculate the normalized measurement output $\tilde{O}$. The model loss is then computed using this loss function:

$$\mathcal{L}(\omega, b, \theta) = -\frac{1}{N}\sum_{i=1}^{N}[y_i \cdot \log(\tilde{O}_i) + (1 - y_i) \cdot \log(1 - \tilde{O}_i)], \tag{7}$$

where ω represents the parameters of the classical neural network, θ corresponds to the angle parameters of the QNN, N is the batch size, y_i is the true label for the ith sample in the batch, and $\tilde{O}_i$ is the normalized measurement output for the ith sample.

4 Experiments

4.1 Dataset and settings

Environment of simulation experiments: The following simulation experiments were conducted on a server equipped with Intel Xeon Gold 6230 (2.10 GHz)×2 and NVIDIA RTX A6000×4, with a total running memory of 512 GB.

Environment of experiments on real quantum computer: This part of the experiments was carried out on a superconducting quantum computer named "Wukong", manufactured by OriginQ [36]. The chip has 72 tunable transmon qubits and 126 tunable couplers, with an average single qubit gate fidelity 99.7% and two qubit gate fidelity 97.2%.

Training dataset: In our training set, we use LSUN Church as the real image dataset and images generated by StyleGAN2 [37], which was trained on LSUN Church, as the fake image dataset. Throughout all the experiments, during the training phase, all images were randomly cropped to a size of 256×256.

Test dataset: To evaluate the generalizability of the proposed method to unseen models and diverse scene images, we utilized the test dataset provided by Wang et al. [5]. This dataset includes multiscene generated images from 11 different generative models, along with corresponding real images. We specifically used test sets for seven models: ProGAN [38], StyleGAN [39], BigGAN [40], CycleGAN [41], StarGAN [42], GauGAN [43], and StyleGan2 [37]. All test images have a size of 256×256. The specific information for each model's test dataset is outlined in Table 1.

To compare with classical methods, our approach was simulated on classical computers. Finally, our model was run on a quantum computer to assess its feasibility on an actual quantum system. During the training phase,

Table 2 Experimental results (mean accuracy (%) ± standard deviation) comparing different AI-generated image detection algorithms. The best results are in bold.

Method	BigGAN	CycleGAN	GauGAN	ProGAN	StarGAN	StyleGAN	StyleGAN2	Average
Wang-0.1	73.27 ± 1.6	66.29 ± 1.94	77.94 ± 2.94	88.85 ± 2.12	88.22 ± 3.07	91.35 ± 1.21	84.09 ± 2.28	81.43 ± 1.32
Wang-0.5	72.40 ± 1.99	71.31 ± 2.24	83.53 ± 3.12	86.98 ± 1.01	79.83 ± 4.69	81.56 ± 3.99	74.77 ± 1.97	78.62 ± 1.39
No-down	81.11 ± 0.67	71.92 ± 3.06	87.63 ± 0.73	92.05 ± 0.61	78.75 ± 0.74	89.58 ± 2.03	86.02 ± 2.29	83.86 ± 0.51
CR-ResNet	58.68 ± 1.32	71.59 ± 1.84	73.35 ± 2.56	79.88 ± 1.42	80.84 ± 1.39	75.57 ± 2.74	64.20 ± 1.66	72.02 ± 1.56
CR-EfficientNet	55.85 ± 1.44	64.28 ± 1.96	57.93 ± 2.81	70.12 ± 1.63	52.20 ± 1.47	58.66 ± 2.12	65.24 ± 1.73	60.61 ± 1.46
LGrad	59.86 ± 1.43	61.92 ± 1.37	57.77 ± 0.86	76.29 ± 0.45	83.42 ± 4.42	82.44 ± 0.50	83.42 ± 0.85	72.16 ± 0.37
UniFD	$\mathbf{95.87} \pm 0.2$	$\mathbf{96.45} \pm 0.43$	$\mathbf{98.86} \pm 0.14$	96.18 ± 0.26	87.65 ± 2.61	81.50 ± 0.43	86.64 ± 0.72	91.88 ± 0.37
Classic	83.60 ± 1.56	87.97 ± 2.16	85.78 ± 1.45	92.78 ± 1.53	98.64 ± 0.61	93.94 ± 0.94	92.77 ± 1.29	90.78 ± 0.97
Hybrid	88.58 ± 2.43	92.97 ± 2.12	89.13 ± 2.87	$\mathbf{96.20} \pm 1.21$	$\mathbf{98.76} \pm 0.73$	$\mathbf{94.64} \pm 1.08$	$\mathbf{94.00} \pm 0.96$	$\mathbf{93.47} \pm 1.31$

we implemented the classical-quantum hybrid neural network using PyTorch for the classical NN and PennyLane for the quantum NN. We applied data augmentation by randomly applying Gaussian blur and JPEG compression operations to the input images, with a probability of 50% for each operation. For Gaussian blur, the parameter σ was randomly selected from $[0, 3]$, and for JPEG compression, the quality factor was randomly chosen from the set $\{30, 31, \ldots, 100\}$. All the experiments utilized the Adam optimizer with a batch size of 16. The classical neural network employed an initial learning rate of 1E−5, and the QNN used an initial learning rate of 1E−3. If the model accuracy on the validation set does not increase for six consecutive epochs, we stop the training.

4.2 Comparative analysis of classification accuracy in different AI-generated image detection algorithms

To investigate the effectiveness of our proposed method, we will compare it with the following methods.

(1) Wang-0.1 and Wang-0.5 [5] (CVPR 2020), ResNet-50 detectors trained with Gaussian blur and JPEG compression for data augmentation, with Gaussian blur and JPEG compression probabilities of 10% and 50%, respectively.

(2) No-Down [6] (ICME 2021), a detector based on Wang et al.'s method [5] but with the removal of the downsampling module from ResNet-50.

(3) CR-ResNet-50 and CR-EfficientNet-b0 [44] (ECCV 2022), detectors that extract color-robust features through data augmentation methods, using ResNet-50 and EfficientNet-b0 as their respective backbone networks.

(4) Lgrad [45] (CVPR 2023), a detector that uses image gradient maps as feature representations and trains a classifier using these gradient maps.

(5) UniFD [26] (CVPR 2023), a detector trains a classifier within the CLIP:ViT pre-trained feature space.

(6) Classic, a fully classical neural network corresponding to the hybrid neural network. It is achieved by replacing the QNN in the hybrid neural network with a linear layer that has the same input and output dimensions.

For a fair comparison, all methods will be trained on a training dataset with a sample size of 2k, including 1k StyleGAN2-generated images and 1k real images from the LSUN church.

Table 2 displays the classification accuracy of various methods on the test set. We discovered that our method achieved the highest average classification accuracy across seven test sets, demonstrating the effectiveness of our designed hybrid NN. Specifically, we achieved the highest classification accuracy in the ProGAN, StarGAN, StyleGAN, and StyleGAN2 test sets, with an overall average classification accuracy of 93.47%. This marks a 1.59% improvement over UniFD, the best-performing method among all those compared. We observed that UniFD achieves the highest detection accuracy on the BigGAN, CycleGAN, and GauGAN datasets, but its performance is lacking on the ProGAN, StarGAN, StyleGAN, and StyleGAN2 datasets, even though it was trained using a dataset composed of StyleGAN2 generated images. Moreover, our hybrid NN surpassed a fully classical NN by 2.69% in average classification accuracy, indicating that integrating a QNN can enhance detection capabilities and model generalizability.

Our hybrid NN also showed significant improvements over LGrad, especially with the BigGAN, CycleGAN, and GauGAN datasets. This may be due to differences in the gradient map representations between StyleGAN2 and those of BigGAN, CycleGAN, and GauGAN. Therefore, when the LGrad is trained using StyleGAN2's gradient maps, it may not effectively detect images from datasets like BigGAN, CycleGAN, and GauGAN. Additionally, our method outperformed other approaches such as Wang-0.1, Wang-0.5, No-down, CR-ResNet-50, and CR-EfficientNet-b0. These data-driven methods require extensive datasets for training. In our experiment, we used a single generative scenario with a limited dataset. Due to their limited ability to generalize from small training data, these methods were less effective in detection.

Table 3 Ablation experiment results (mean accuracy (%) ± standard deviation) of the effectiveness of QNN. The best results are in bold.

Model	BigGAN	CycleGAN	GauGAN	ProGAN	StarGAN	StyleGAN	StyleGAN2	Average
Linear(4-1)	83.60 ± 1.56	87.97 ± 2.16	85.78 ± 1.45	92.78 ± 1.53	98.64 ± 0.61	93.94 ± 0.94	92.77 ± 1.29	90.78 ± 0.97
Linear(4-4-1)	85.08 ± 1.50	89.14 ± 1.87	86.77 ± 1.59	93.95 ± 1.37	97.43 ± 2.03	92.69 ± 1.29	92.09 ± 0.88	91.02 ± 0.69
Linear(16-1)	82.41 ± 1.30	87.27 ± 1.11	85.97 ± 1.46	92.76 ± 1.17	98.67 ± 0.84	94.24 ± 1.18	93.02 ± 1.02	90.62 ± 0.58
Hybrid	**88.58** ± 2.43	**92.97** ± 2.12	**89.13** ± 2.87	**96.20** ± 1.21	**98.76** ± 0.73	**94.64** ± 1.08	**94.00** ± 0.96	**93.47** ± 1.31

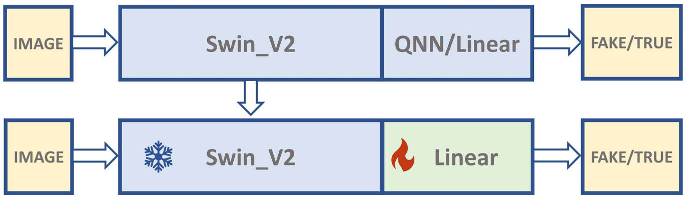

Figure 6 (Color online) A diagram depicting the experimental framework of the impact of QNN on a classical neural network. In this experiment, based on Swin V2, a classical-quantum hybrid neural network and a fully classical neural network were trained, respectively. Then, the QNN and Linear layer were removed, respectively, and the two trained Swin V2 models were retained. Finally, the parameters of the trained Swin V2 models were fixed, and a Linear layer was retrained for classification on the basis of the features extracted by these models, so as to evaluate the impact on the generalization of the features extracted by Swin V2 when jointly trained with the QNN/Linear layer.

4.3 Ablation study

4.3.1 *The effectiveness of QNN*

To further explore the effectiveness of QNN, we constructed three classical NNs for comparison. The first uses a Linear(4-1) to replace the QNN in the hybrid network, representing a model with the same input and output dimensions as the QNN. The second employs a two-layer linear structure, Linear(4-4-1), to represent a model with a similar number of parameters to the QNN. The third replaces the QNN with a Linear(16-1), representing a model with the same dimensionality as the QNN. As depicted in Table 3, the models with increased parameters and expanded representation dimensions have similar classification accuracy compared to the fully classical model in which the QNN is substituted with Linear(4-1). However, our proposed classical-quantum hybrid neural network maintains a higher classification accuracy compared to these fully classical neural networks, despite having comparable parameter counts and dimensionality. Due to the quantum principles underlying QNN, such as entanglement and superposition, which classical neural networks lack, these unique characteristics cannot be obtained simply by employing classical neural networks with a comparable number of parameters and dimensionality.

4.3.2 *Impact of QNN on feature extraction network*

To explore the impact of the presence of QNN on the feature extraction network under joint training conditions, we conducted the following experiment. Specifically, we extracted two Swin V2 models, one from the hybrid NN and the other from the fully classical NN, each trained with 2000 samples. We then fixed the weights of these two Swin V2 models and respectively retrained the classifier composed of linear layers in the feature spaces extracted by these two models, as depicted in Figure 6.

As illustrated in Figure 7, whether the retrained models come from hybrid NN or from fully classical NN, their classification accuracy tends to decrease somewhat compared to their original models. However, the retrained models from hybrid NN significantly outperform the one from fully classical NN in terms of detection accuracy, especially on the test sets of BigGAN, CycleGAN, GauGAN, and ProGAN. This indicates that the joint training of hybrid NN, due to the presence of QNN, enhances the feature extraction capability of Swin V2, thereby increasing the generalizability of the extracted features.

4.3.3 *The impact of training sample size*

To examine the impact of training sample size on the hybrid NN, we established four datasets with varying sample sizes for our experiments, specifically with 2k, 4k, 10k, and 20k training samples. Additionally, we created a fully classical NN for comparison by substituting the QNN in the hybrid NN with a linear layer. The experimental results are presented in Table 4, where each entry represents the average of 5 repeated experiments.

As shown in Table 4, the hybrid NN (labeled as “Hybrid” in the table) exhibits a noticeable improvement in classification accuracy compared to the classical NN (labeled as “Classical”), especially in scenarios with a small sample size. With a training sample size of 2k, the hybrid NN demonstrates significant accuracy improvements

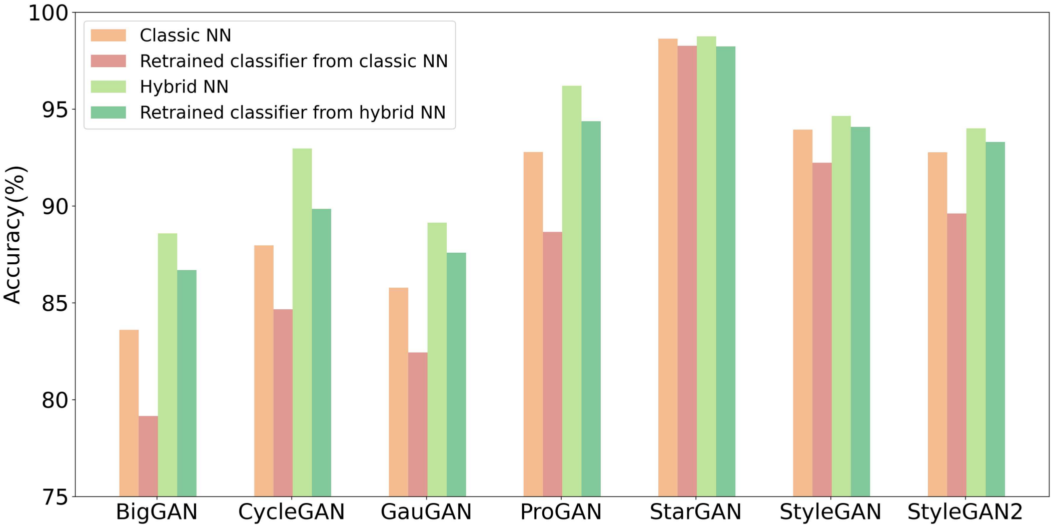

Figure 7 (Color online) The experimental results of the impact of QNN on the feature extraction network. Classic NN and Hybrid NN were first trained in a dataset with 2000 samples. Then, the final Linear layer/QNN was removed, respectively. After that, the parameters of the remaining two Swin V2 were fixed, and the Linear layer was retrained respectively for classification.

Table 4 Ablation experiment results (mean accuracy (%) ± standard deviation) in joint training under various training sample sizes. The best results are in bold. The underlined values indicate results that passed the KS test.

Model	BigGAN	CycleGAN	GauGAN	ProGAN	StarGAN	StyleGAN	StyleGAN2	Average
Classical-2k	83.60 ± 1.56	87.97 ± 2.16	85.78 ± 1.45	92.78 ± 1.53	98.64 ± 0.61	93.94 ± 0.94	92.77 ± 1.29	90.78 ± 0.97
Hybrid-2k	**88.58** ± 2.43	**92.97** ± 2.12	**89.13** ± 2.87	**96.20** ± 1.21	**98.76** ± 0.73	**94.64** ± 1.08	**94.00** ± 0.96	**93.47** ± 1.31
Classical-4k	87.49 ± 1.76	92.40 ± 0.96	88.04 ± 1.00	95.81 ± 0.69	98.02 ± 0.65	**95.17** ± 1.16	94.82 ± 0.95	93.11 ± 0.24
Hybrid-4k	**88.66** ± 1.32	**93.33** ± 1.14	**89.68** ± 0.68	**96.95** ± 0.42	**98.76** ± 0.87	95.12 ± 1.06	**95.51** ± 1.32	**94.00** ± 0.39
Classical-10k	89.88 ± 2.13	95.35 ± 0.87	90.12 ± 0.95	97.60 ± 0.53	97.74 ± 0.64	**95.95** ± 0.85	**96.00** ± 0.56	94.66 ± 0.79
Hybrid-10k	**91.28** ± 1.35	**96.20** ± 0.48	**92.97** ± 0.74	**97.83** ± 0.19	**98.26** ± 0.57	95.52 ± 0.50	94.66 ± 1.32	**95.24** ± 0.45
Classical-20k	90.98 ± 1.86	95.23 ± 1.08	90.35 ± 1.43	**98.12** ± 0.24	98.57 ± 0.33	96.32 ± 1.21	95.53 ± 1.48	95.01 ± 0.33
Hybrid-20k	**91.28** ± 1.24	**95.48** ± 1.12	**91.90** ± 1.01	98.11 ± 0.41	**98.96** ± 0.63	**96.34** ± 1.62	**96.00** ± 1.20	**95.44** ± 0.34

over the classical NN on test sets such as BigGAN, CycleGAN, GauGAN, ProGAN, and StyleGAN2. However, the classification accuracy on datasets like StarGAN and StyleGAN is comparable between the two models. Overall, the hybrid NN achieves an average improvement of 2.69% compared to the classical NN across these seven datasets. As the training sample size increases, the performance gap between the hybrid NN and the classical NN gradually diminishes. With a training sample size of 20k, the hybrid NN achieves an average accuracy improvement of 0.43% compared to the classical NN across the seven datasets. Additionally, we conducted a Kolmogorov-Smirnov (KS) test [46], a non-parametric statistical method used to determine if samples conform to a specific probability distribution. For the average classification accuracy of sample sizes 2k and 4k in Table 4, specifically the underlined data results, we took the repeated experimental results of classical NN and hybrid NN as two sample groups. Our null hypothesis (H0) was that these two sample groups came from the same distribution. After calculation, we found that the p-value for the 2k sample size is 0.0002, and for the 4k sample size, it is 0.000014. The p-value represents the significance level of the KS test. Since the p-values in both cases are less than 0.05, we reject the null hypothesis, indicating that in these cases, the distributions of repeated results from hybrid NN and classical NN do not follow the same distribution. Therefore, it can be assured that the classification accuracy of hybrid neural networks is superior to that of classical neural networks when the sample sizes are 2k and 4k. This suggests that with a small number of training samples, hybrid NN exhibits superior generalization abilities compared to classical NN, indirectly supporting the findings of Caro et al. [16].

4.4 Comparative analysis of different modules in quantum neural networks

4.4.1 *Encoding circuit*

The encoding operation serves as the first step in QNN, facilitating the conversion of feature data from the classical representation space to the quantum Hilbert space. In the method introduced in Section 3, we utilize the R_X gate

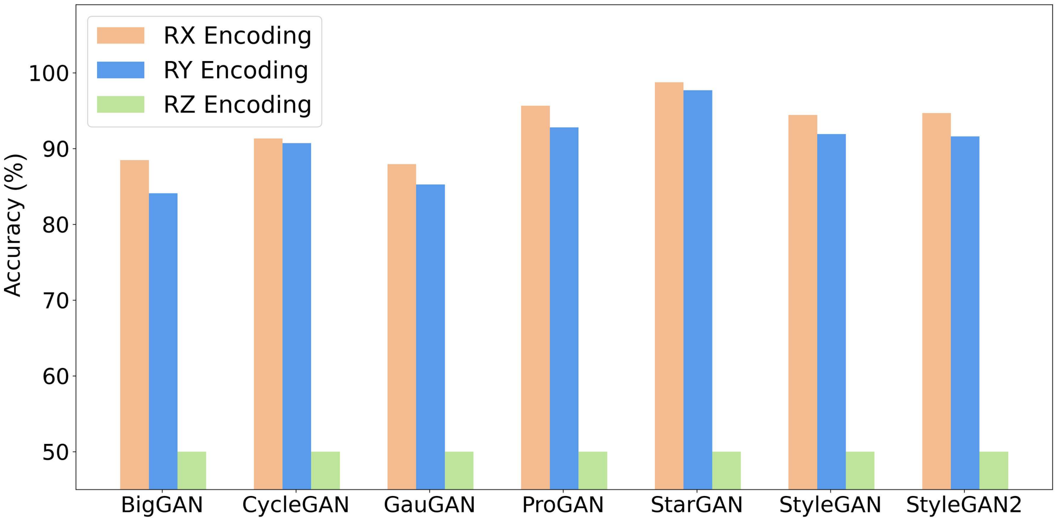

Figure 8 (Color online) Experimental results of encoding using different rotation gates. We kept the other parts of the QNN unchanged and used different rotation gates for encoding as a comparison.

Table 5 Experimental results (mean accuracy (%) ± standard deviation) comparing different parameterized quantum circuits. The best results are in bold.

Circuit	BigGAN	CycleGAN	GauGAN	ProGAN	StarGAN	StyleGAN	StyleGAN2	Average
Circuit5	84.91 ± 4.19	87.88 ± 3.29	84.68 ± 3.64	91.26 ± 4.10	98.00 ± 1.36	91.89 ± 2.10	90.89 ± 3.48	89.93 ± 2.84
Circuit6	85.66 ± 3.17	89.70 ± 2.07	85.04 ± 2.33	93.12 ± 1.92	98.62 ± 0.79	92.53 ± 1.04	92.34 ± 1.51	91.00 ± 1.25
Circuit7	85.93 ± 3.26	88.42 ± 2.04	84.56 ± 2.64	92.43 ± 1.54	97.66 ± 2.31	92.39 ± 2.35	91.77 ± 2.44	90.45 ± 1.77
Circuit8	84.25 ± 2.70	88.84 ± 2.17	83.53 ± 3.28	92.76 ± 1.60	97.85 ± 1.68	92.67 ± 1.03	92.30 ± 1.36	90.31 ± 1.55
Circuit13	**88.72** ± 0.78	92.89 ± 2.20	87.34 ± 0.81	94.81 ± 1.08	98.13 ± 1.07	93.50 ± 1.25	93.29 ± 1.28	92.67 ± 0.69
Circuit14	87.36 ± 2.31	**93.21** ± 2.14	86.53 ± 1.64	95.52 ± 0.74	**99.01** ± 0.58	93.06 ± 1.97	**94.08** ± 0.41	92.68 ± 0.86
Proposed circuit	88.58 ± 2.43	92.97 ± 2.12	**89.13** ± 2.87	**96.20** ± 1.21	98.76 ± 0.73	**94.64** ± 1.08	94.00 ± 0.96	**93.47** ± 1.31

to encode classical feature data into a quantum state. Specifically, we treat the feature data as rotation angles, using the R_X gate to control the quantum state's rotation around the X-axis of the Bloch sphere by the corresponding angle. However, there's more than one way to implement angle encoding. We conducted an experiment utilizing three rotation gates for angle encoding to investigate the differences between them. From Figure 8, it is evident that encoding with the R_X gate yields the best performance, with slightly inferior results for the R_Y gate, and the poorest performance observed with the R_Z gate, akin to random probability outcomes. The QNN designed in this study employs the R_Y gate and R_Z gate as the learnable quantum gate. When encoding with the R_X gate, the quantum state undergoes rotations around both the X-, Y-, and Z-axes, effectively exploring the entire quantum state space spanning the entire Bloch sphere. However, when encoding with the R_Y gate, the quantum state undergoes rotations around the Y- and Z-axes, and cannot explore the entire representation space of the Bloch sphere efficiently. When using R_Z gate for encoding, since the initial state of the quantum state is on the Z-axis, the quantum state will rotate in an in-place rotation, thereby failing to achieve effective encoding.

4.4.2 *Parameterized quantum circuit*

In QNN, the parameterized quantum circuit is a crucial component that significantly influences performance. To investigate the effectiveness of our designed parameterized quantum circuit, we compared it with Circuit5, Circuit6, Circuit7, Circuit8, Circuit13, and Circuit14 designed by Sim et al. [47], as shown in Figure 9. Comparative experiments were performed on a dataset with a training sample size of 2000, and the results are presented in Table 5.

From Table 5, it is evident that Circuit5 and Circuit6, despite having the highest number of parameters, did not yield the best outcomes. Notably, Circuit5 exhibited the lowest classification accuracy among all circuits. Circuit7 and Circuit8, sharing similar circuit structures, demonstrated comparable classification performance slightly surpassing Circuit5. Similarly, Circuit13 and Circuit14 also exhibited similar classification performance, but with higher accuracy compared to other circuits proposed by Sim et al. [47]. In this experiment, the circuit we proposed achieved the highest classification accuracy, outperforming the best-performing Circuit14 by 0.79%. This advantage

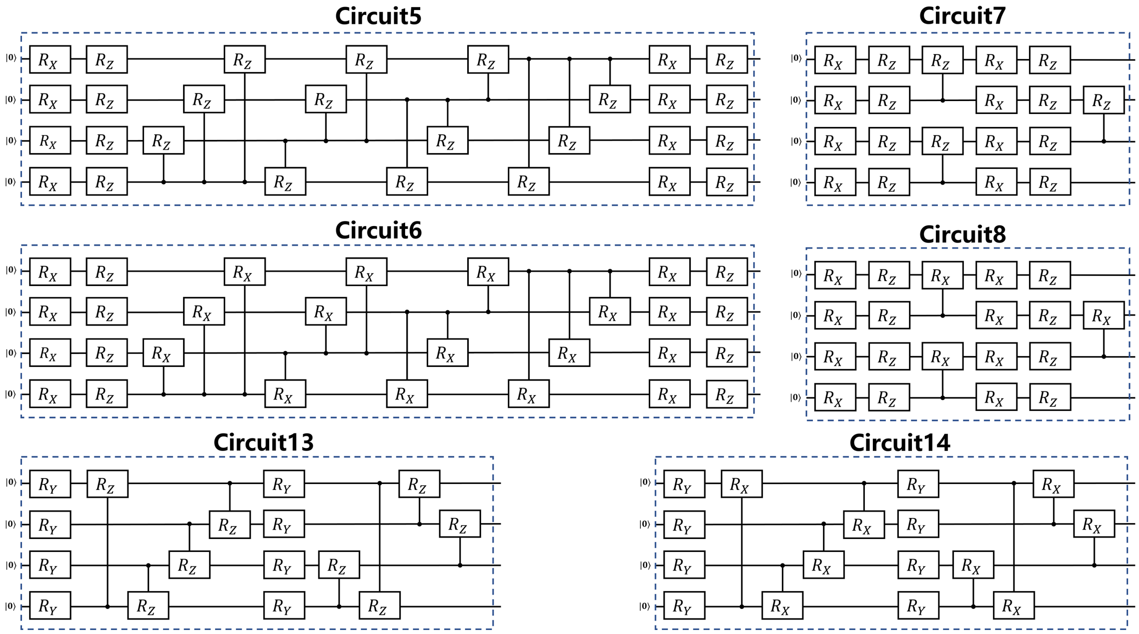

Figure 9 (Color online) The parameterized quantum circuits designed by Sim et al. [47].

stemming from the proposed circuit is based on the ALT design, which has better trainability than the HEA to which Circuit14 belongs. Based on the analysis in [35], ALT has better trainability compared with HEA, and it is derived in [21] that shallow ALT circuits will not encounter the problem of vanishing gradients. These advantages and characteristics are fit well for the training of a classical-quantum hybrid neural network, as they can better achieve gradient propagation. Therefore, the proposed circuit achieves higher classification accuracy compared to Circuit14.

4.4.3 *Entangler circuit*

At the end of the QNN, we introduced an entanglement circuit to obtain a measurement output capable of representing the entire quantum system. To investigate the effectiveness of the entanglement circuit, we removed it from the QNN while keeping other parts unchanged. We then compared this modified network with the network proposed in this study. The results are illustrated in Figure 10. It is evident that adding an entanglement circuit effectively enhances classification accuracy. This is because the entanglement layer effectively consolidates information from various qubits into a single qubit. Measuring this qubit enables us to obtain a more comprehensive understanding of the entire quantum system.

4.5 Experiments on the real quantum computer

To investigate the feasibility of our proposed method on near-term quantum devices, we conducted inference experiments on a real quantum computer. Specifically, we conducted experiments with a superconducting quantum computer named “Wukong” developed by OriginQ [36], utilizing their quantum cloud platform. The quantum computer has 72 qubits, with an average single-qubit gate fidelity of 99.7% and an average two-qubit gate fidelity of 97.2%. In experiments, we employed a quantum circuit-level parallelism strategy, which involves using qubits from different areas of a quantum chip to execute multiple circuits simultaneously. Specifically, we utilized 16 qubits to achieve four-way parallelism, reducing the number of quantum circuits needed to 25% compared to when no parallelism is used. For example, the test set contains a total of 56600 samples. Without quantum circuit parallelism, it would require executing 56600 circuits. However, with four-way parallelism, only 14150 circuits need to be executed. The topological structure of the quantum chip we used, as well as the selected qubits, is shown in Figure 11.

Table 6 presents the inference accuracy and loss of our hybrid NN on both the simulator and real quantum computer. We observed that the inference accuracy of the hybrid NN was similar on both the simulator and the real quantum computer, with less than a 1% difference in accuracy across six out of seven test sets, except for StyleGAN2. Overall, the average classification accuracy of the hybrid NN on the real quantum computer dropped

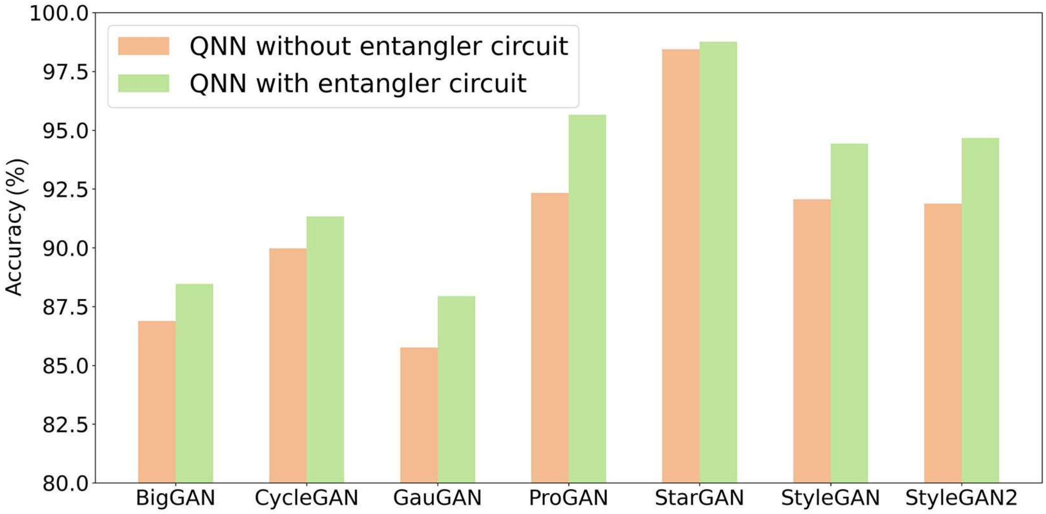

Figure 10 (Color online) Experimental results on the ablation of the entangler circuit. We kept the other parts of the QNN unchanged and compared the model without the entangler circuit with the model with the entangler circuit to evaluate the impact of the entangler circuit on performance.

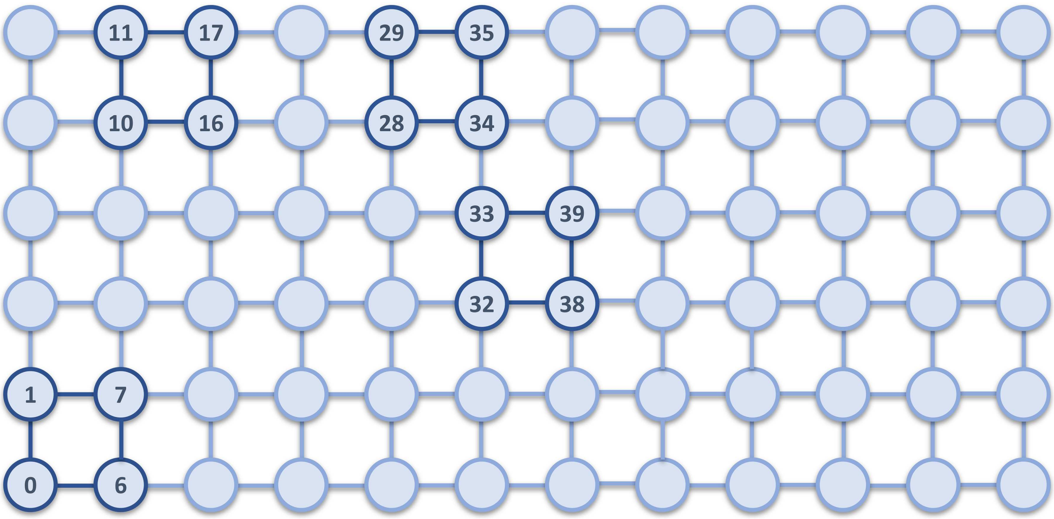

Figure 11 (Color online) The topology of the quantum computer used in this paper. We selected four regions containing four qubits from it to implement the quantum circuit-level parallelism strategy. Within the same period of time, the calculations of four quantum circuits can be carried out simultaneously.

Table 6 Inference results of the hybrid classical-quantum neural network on a real quantum computer. The best results are in bold.

	Execution platform	BigGAN	CycleGAN	GauGAN	ProGAN	StarGAN	StyleGAN	StyleGAN2	Average
Accuracy (%)	Simulator	**89.08**	**93.98**	**91.70**	96.40	**98.15**	**94.69**	**94.67**	**94.10**
	Real quantum computer	88.73	93.41	91.07	**96.79**	97.75	94.55	92.34	93.52
Loss	Simulator	**0.2739**	**0.1810**	**0.2074**	**0.1055**	**0.0839**	**0.1440**	**0.1364**	**0.1617**
	Real quantum computer	0.5045	0.5193	0.4822	0.4278	0.4176	0.4454	0.4700	0.4667

by just 0.58% compared to the simulator. It demonstrates that our proposed hybrid NN works equally well on real quantum computers, verifying the feasibility of the network on near-term quantum devices.

However, the loss on the real quantum computer was significantly higher across all datasets compared to that on the simulator. Unlike the simulator, computations in a real quantum computer are often affected by quantum noise, which leads to discrepancies between the actual outputs and the simulated results. Figure 12 illustrates the significant differences in the distribution of output logits by the hybrid NN across the two computing environments. On the simulator, the output logits are mostly concentrated in the high-confidence areas, near 0 and 1. However

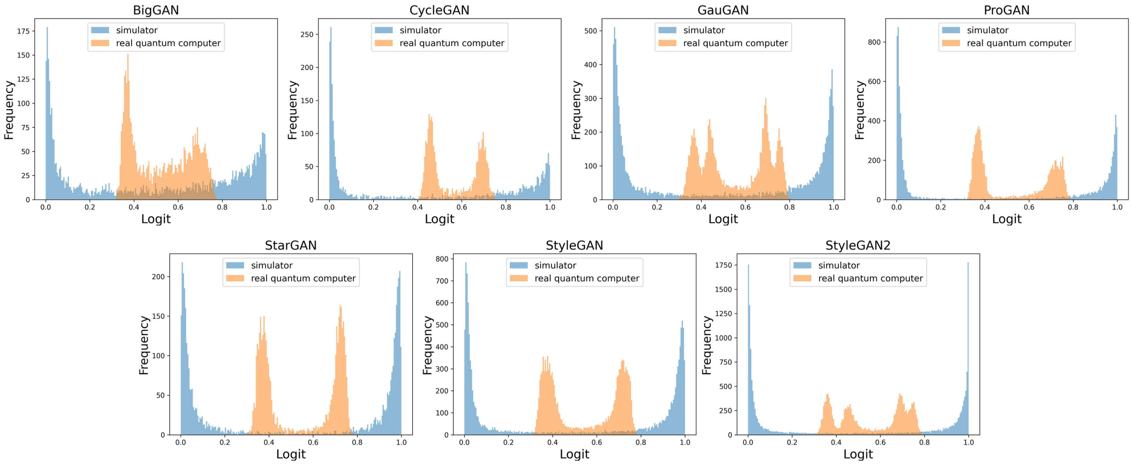

Figure 12 (Color online) Logit distribution of the hybrid NN for inference in the simulator and real quantum computer. The horizontal axis is the output logit of the model. We use 0.5 as the threshold for classification. If it is greater than 0.5, it is predicted as a real image, and if it is less than 0.5, it is predicted as an AI generated image. The vertical axis represents frequency (number of occurrences).

on the real quantum computer, the presence of quantum noise limits the range of qubit measurement values [48], causing the logit distribution to shift towards the center. Despite this, two distinct distributions corresponding to predictions for real images and AI-generated images are still observed, demonstrating that our method can effectively differentiate between real and AI-generated images even on a real quantum computer. These results indicate that our hybrid NN is feasible and effective in today's real quantum computing systems.

5 Conclusion

This paper introduces a novel classical-quantum hybrid neural network for AI-generated image detection tasks. To the best of our knowledge, this is the first instance of incorporating QNN in the field of AI-generated image detection. The introduction of QNN brings about quantum properties such as superposition, entanglement, and a larger representational space, which classical neural networks lack. The experiments indicate that our proposed classical-quantum hybrid neural network demonstrates strong generalization, especially in scenarios with limited training samples, showcasing a noticeable advantage over many state-of-the-art methods. As the first study that utilizes the generalization ability and expressibility of QNN for AI-generated image detection, this model demonstrates certain generalization advantages of QNN and has been further verified on a real quantum computer, providing a demonstration for using QNN to achieve more generalized AI-generated image detection.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 62272003), Innovation Program for Quantum Science and Technology (Grant No. 2021ZD0302300), and Science and Technology Major Project of Anhui Province (Grant No. 202423s06050001).

References

1 Zhang X, Karaman S, Chang S F. Detecting and simulating artifacts in GAN fake images. In: Proceedings of the 2019 IEEE International Workshop on Information Forensics and Security (WIFS), 2019. 1–6
2 Durall R, Keuper M, Keuper J. Watch your up-convolution: CNN based generative deep neural networks are failing to reproduce spectral distributions. In: Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020. 7887–7896
3 Frank J, Eisenhofer T, Schönherr L, et al. Leveraging frequency analysis for deep fake image recognition. In: Proceedings of the 37th International Conference on Machine Learning, 2020. 3247–3258
4 Liu B, Yang F, Bi X, et al. Detecting generated images by real images. In: Proceedings of the European Conference on Computer Vision, 2022. 95–110
5 Wang S Y, Wang O, Zhang R, et al. CNN-generated images are surprisingly easy to spot... for now. In: Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020. 8692–8701
6 Gragnaniello D, Cozzolino D, Marra F, et al. Are GAN generated images easy to detect? A critical analysis of the state-of-the-art. In: Proceedings of the 2021 IEEE International Conference on Multimedia and Expo (ICME), 2021. 1–6
7 Ju Y, Jia S, Ke L, et al. Fusing global and local features for generalized AI-synthesized image detection. In: Proceedings of the 2022 IEEE International Conference on Image Processing (ICIP), 2022. 3465–3469
8 McCloskey S, Albright M. Detecting gan-generated imagery using saturation cues. In: Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), 2019. 4584–4588
9 McCloskey S, Albright M. Detecting gan-generated imagery using color cues. 2018. ArXiv:1812.08247
10 Schuld M, Sinayskiy I, Petruccione F. An introduction to quantum machine learning. Contemp Phys, 2015, 56: 172–185
11 Biamonte J, Wittek P, Pancotti N, et al. Quantum machine learning. Nature, 2017, 549: 195–202
12 Cerezo M, Verdon G, Huang H Y, et al. Challenges and opportunities in quantum machine learning. Nat Comput Sci, 2022, 2: 567–576
13 Schuld M, Killoran N. Quantum machine learning in feature Hilbert spaces. Phys Rev Lett, 2019, 122: 040504
14 Havlíček V, Córcoles A D, Temme K, et al. Supervised learning with quantum-enhanced feature spaces. Nature, 2019, 567: 209–212

15 Abbas A, Sutter D, Zoufal C, et al. The power of quantum neural networks. Nat Comput Sci, 2021, 1: 403–409

16 Caro M C, Huang H Y, Cerezo M, et al. Generalization in quantum machine learning from few training data. Nat Commun, 2022, 13: 4919

17 Cheng T, Zhao R S, Wang S, et al. Analysis of learnability of a novel hybrid quantum-classical convolutional neural network in image classification. Chin Phys B, 2024, 33: 040303

18 Zhao R, Cheng T, Wang R, et al. Artificial intelligence warm-start approach: optimizing the generalization capability of QAOA in complex energy landscapes. New J Phys, 2024, 26: 053016

19 Mari A, Bromley T R, Izaac J, et al. Transfer learning in hybrid classical-quantum neural networks. Quantum, 2020, 4: 340

20 Liu Z, Hu H, Lin Y, et al. Swin Transformer v2: scaling up capacity and resolution. In: Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022. 11999–12009

21 Cerezo M, Sone A, Volkoff T, et al. Cost function dependent barren plateaus in shallow parametrized quantum circuits. Nat Commun, 2021, 12: 1791

22 Yu N, Davis L S, Fritz M. Attributing fake images to GANs: learning and analyzing GAN fingerprints. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019

23 Marra F, Gragnaniello D, Verdoliva L, et al. Do GANs leave artificial fingerprints? In: Proceedings of the 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), 2019. 506–511

24 Chai L, Bau D, Lim S, et al. What makes fake images detectable? Understanding properties that generalize. In: Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, 2020. 103–120

25 He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016. 770–778

26 Ojha U, Li Y, Lee Y J. Towards universal fake image detectors that generalize across generative models. In: Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023. 24480–24489

27 Radford A, Kim J W, Hallacy C, et al. Learning transferable visual models from natural language supervision. In: Proceedings of the International Conference on Machine Learning, 2021. 8748–8763

28 Chen B Y, Lei H, Shen H D, et al. A hybrid quantum-based PIO algorithm for global numerical optimization. Sci China Inf Sci, 2019, 62: 070203

29 Gao S, Pan S J, Yang Y G. Quantum algorithm for kernelized correlation filter. Sci China Inf Sci, 2022, 66: 129501

30 Li Q Y, Huang Y H, Jin S, et al. Quantum spectral clustering algorithm for unsupervised learning. Sci China Inf Sci, 2022, 65: 200504

31 He X Y, Sun X M, Zhang J L. Quantum search with prior knowledge. Sci China Inf Sci, 2024, 67: 192503

32 Larocca M, Ju N, García-Martín D, et al. Theory of overparametrization in quantum neural networks. Nat Comput Sci, 2023, 3: 542–551

33 Li G X, Zhao X Q, Wang X. Quantum self-attention neural networks for text classification. Sci China Inf Sci, 2024, 67: 142501

34 Liu Z, Lin Y, Cao Y, et al. Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021. 9992–10002

35 Nakaji K, Yamamoto N. Expressibility of the alternating layered ansatz for quantum computation. Quantum, 2021, 5: 434

36 Origin Quantum. OriginQ Cloud. https://qcloud.originqc.com.cn/en

37 Karras T, Laine S, Aittala M, et al. Analyzing and improving the image quality of StyleGAN. In: Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020. 8107–8116

38 Karras T, Aila T, Laine S, et al. Progressive growing of GANs for improved quality, stability, and variation. In: Proceedings of the International Conference on Learning Representations (ICLR), 2018. 1–26

39 Karras T, Laine S, Aila T. A style-based generator architecture for generative adversarial networks. In: Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019. 4396–4405

40 Brock A, Donahue J, Simonyan K. Large scale gan training for high fidelity natural image synthesis. In: Proceedings of the International Conference on Learning Representations (ICLR), 2019. 1–35

41 Zhu J Y, Park T, Isola P, et al. Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), 2017. 2242–2251

42 Choi Y, Choi M, Kim M, et al. StarGAN: unified generative adversarial networks for multi-domain image-to-image translation. In: Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018. 8789–8797

43 Park T, Liu M Y, Wang T C, et al. Semantic image synthesis with spatially-adaptive normalization. In: Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019. 2332–2341

44 Chandrasegaran K, Tran N T, Binder A, et al. Discovering transferable forensic features for CNN-generated images detection. In: Computer Vision—ECCV 2022. Cham: Springer Nature Switzerland, 2022. 671–689

45 Tan C, Zhao Y, Wei S, et al. Learning on gradients: generalized artifacts representation for GAN-generated images detection. In: Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023. 12105–12114

46 Massey Jr. F J. The Kolmogorov-Smirnov test for goodness of fit. J Am Stat Assoc, 1951, 46: 68–78

47 Sim S, Johnson P D, Aspuru-Guzik A. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. Adv Quantum Tech, 2019, 2: 1900070

48 van den Berg E, Minev Z K, Temme K. Model-free readout-error mitigation for quantum expectation values. Phys Rev A, 2022, 105: 032620