PAPER • OPEN ACCESS

Quantum machine learning for image classification

To cite this article: Arsenii Senokosov *et al* 2024 *Mach. Learn.: Sci. Technol.* **5** 015040

View the article online for updates and enhancements.

MACHINE
LEARNING
Science and Technology

PAPER

Quantum machine learning for image classification

Arsenii Senokosov[*] ⓘ, Alexandr Sedykh ⓘ, Asel Sagingalieva ⓘ, Basil Kyriacou ⓘ and Alexey Melnikov[*] ⓘ

Terra Quantum AG, Kornhausstrasse 25, 9000 St. Gallen, Switzerland
[*] Authors to whom any correspondence should be addressed.

E-mail: ase@terraquantum.swiss and ame@terraquantum.swiss

Abstract

Image classification, a pivotal task in multiple industries, faces computational challenges due to the burgeoning volume of visual data. This research addresses these challenges by introducing two quantum machine learning models that leverage the principles of quantum mechanics for effective computations. Our first model, a hybrid quantum neural network with parallel quantum circuits, enables the execution of computations even in the noisy intermediate-scale quantum era, where circuits with a large number of qubits are currently infeasible. This model demonstrated a record-breaking classification accuracy of 99.21% on the full MNIST dataset, surpassing the performance of known quantum–classical models, while having eight times fewer parameters than its classical counterpart. Also, the results of testing this hybrid model on a Medical MNIST (classification accuracy over 99%), and on CIFAR-10 (classification accuracy over 82%), can serve as evidence of the generalizability of the model and highlights the efficiency of quantum layers in distinguishing common features of input data. Our second model introduces a hybrid quantum neural network with a Quanvolutional layer, reducing image resolution via a convolution process. The model matches the performance of its classical counterpart, having four times fewer trainable parameters, and outperforms a classical model with equal weight parameters. These models represent advancements in quantum machine learning research and illuminate the path towards more accurate image classification systems.

1. Introduction

Image classification is a critical task in the modern world due to its wide range of practical applications in various fields [1]. For instance, in medical imaging, image classification algorithms have been shown to significantly improve the accuracy and speed of diagnoses of many diseases [2, 3]. In the field of autonomous vehicles, image classification plays a crucial role in object detection, tracking, and classification, which is necessary for safe and efficient navigation.

Deep learning approaches [4] like deep convolutional neural networks (CNNs) have emerged as powerful tools for image classification and recognition tasks [5, 6], achieving state-of-the-art performance on various benchmark datasets [7, 8]. However, as the amount of visual data grows, modern neural networks face significant computational challenges.

Quantum technologies, on the other hand, offer the potential to overcome this computational limitation by harnessing the power of quantum mechanics to perform computations in parallel [9]. Quantum machine learning (QML) is a rapidly evolving field that combines the principles of quantum mechanics, and classical machine learning [10–13]. This field has the potential to revolutionize various areas of computing, including image classification [14–16]. It has attracted significant attention due to its potential to solve computational problems that classical computers are unable to solve efficiently [9]. This potential arises from the unique features of quantum computing, such as superposition and entanglement, which can provide an exponential speedup for specific machine learning tasks [17]. QML algorithms produce probabilistic results, which align well with classification problems [18]. They also operate in an exponentially larger search space, which has the potential to enhance their performance [19–21]. However, it is important to note that the realization of

these advantages in practical applications remains an active area of research and investigation. However, the real-world implementation of quantum algorithms faces significant challenges, such as the need for error correction and the high sensitivity of quantum systems to external disturbances [22]. Despite these challenges, QML has shown promising results in several applications [23]. In the context of image classification, QML algorithms can process large datasets of images more efficiently than classical algorithms, leading to faster and more accurate classification [24]. Recent studies have also explored hybrid quantum–classical CNNs and demonstrated the classification [18, 25–30] and generation [31–34] of images.

A promising area of research within QML for image classification is the hybrid quantum neural network (HQNN) [17, 35–37]. HQNNs combine classical deep learning architectures with QML algorithms [38–42], namely parameterized quantum circuits (PQCs), creating a hybrid system that leverages the strengths of both classical and quantum computing. This approach allows for the processing of large datasets with greater efficiency than classical deep learning architectures alone [43, 44]. HQNNs have shown promise in a variety of industrial tasks, e.g. in the healthcare [27, 30, 45–47], chemical [48, 49], financial [50], and aerospace industries [51, 52]. Further research is needed to explore the full potential of HQNNs in image classification and to develop more robust and scalable algorithms.

In this article, we propose two approaches to leverage quantum computing in the field of image recognition. The first approach involves applying parallel PQCs after classical deep convolutional layers, while the second approach involves using an HQNN with a quanvolutional layer. We evaluate the performance of these hybrid models on the MNIST dataset of hand-written digits, which is described in section 2.1.1, and demonstrate their ability to classify images.

Previous studies have primarily focused on either purely quantum solutions [53–55] for image recognition or various hybrid models [36]. Yet, the specific potential of seamlessly integrating quantum circuits with classical neural networks remains an under-explored area. In this work, we venture into this niche and present an innovative architecture for the HQNN, that is specifically designed to operate even in the NISQ era, setting a benchmark by achieving record-breaking classification accuracy with significantly fewer parameters than its classical counterpart. Importantly, we achieved these results without resorting to pretrained models or transfer learning techniques, underscoring the inherent strength of our model's design and training process. Our successful test results across multiple datasets further support the claim of the model's broad applicability and generalizability.

The first model (described in section 2.2) combines classical convolutional layers with parallel quantum layers (HQNN-Parallel). The quantum part is analogous to a classical fully connected layer. We compare the hybrid model with its most closely corresponding classical counterpart (in terms of the architecture and the number of layers) and observe that the hybrid model outperforms the classical model in accuracy (achieving 99.21% accuracy) despite having eight times fewer parameters. Moreover, we tested this model on two more datasets, on Medical MNIST, which is described in section 2.1.2, and on CIFAR-10, which is described in section 2.1.3, to ensure its generalizability.

In the second model (described in section 2.3), we introduce HQNN with a quanvolutional layer (HQNN-Quanv), which is a kernel that applies a convolution to the input image and reduces its resolution. The HQNN-Quanv achieves similar accuracy to the classical model (67% accuracy) despite having four times fewer trainable parameters in the first layer compared to the classical counterpart. Additionally, the hybrid model outperforms the classical model with the same number of weights.

We note that having fewer trainable parameters does not necessarily lead to a more efficient execution of QML models compared to classical ones. The reason for this is the much more expensive training and operation costs required for quantum models relative to classical models. However, with future advancements in quantum hardware technologies, the gap in training and operation costs between quantum and classical models might become narrower. This highlights the potential of quantum computing and QML in advancing the field of image recognition. Our results contribute to the ongoing research in this area and demonstrate the exciting possibilities for the future of QML in other fields.

2. Results

2.1. Datasets
2.1.1. MNIST
This section describes the Modified National Institute of Standards and Technology (MNIST) [56] dataset. The MNIST database consists of a large collection of grey-scale handwritten numbers, ranging from 0 to 9. Sample images from the dataset are presented in figure 1(a). Each image has a resolution of 28×28 pixels, and the main objective is to classify each image by assigning a class label using a neural network. In other words, the task is to recognize which digit is present in the image. This dataset is widely used for making the first steps in the sphere of machine learning. Nevertheless, it is worth studying as it helps to test the
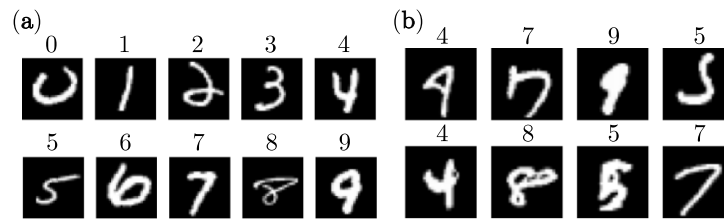
Figure 1. (a) Examples of images from the MNIST dataset. (b) Examples of ambiguous images from the MNIST dataset.
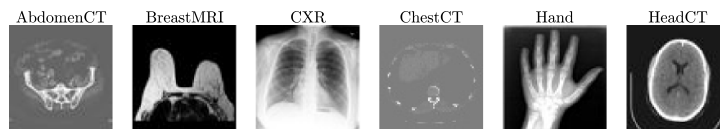

Figure 2. Examples of images from the Medical MNIST dataset.

performance of various neural network models [57, 58], especially models with PQCs [36, 53, 59]. The MNIST dataset used in this study comprises a total of 70 000 images, with 60 000 images reserved for training and 10 000 images for testing. However, in certain cases, it may be advantageous to reduce the number of images in order to expedite the training process and gain immediate insights into the model's performance.

Despite being a widely used dataset, the MNIST database contains a few images that are broken or ambiguous, posing a challenge even for human evaluators. Figure 1(b) provides examples of such images. However, our hybrid model can accurately determine the number in such images with over 99% accuracy.

2.1.2. *Medical* MNIST

The integration of QML techniques, such as HQNNs, holds immense promise in the field of medical image classification. Quantum computing's inherent capacity to handle complex and high-dimensional data, coupled with the power of neural networks, offers a unique advantage for solving intricate medical image analysis tasks. HQNNs leverage quantum computing's ability to efficiently perform certain mathematical operations required for image feature extraction and classification, potentially leading to breakthroughs in medical diagnosis and treatment [60].

To test our models in this sphere, we utilized the Medical MNIST dataset [61], which comprises a total of 58 954 medical images distributed across six distinct categories: Abdomen Computer Tomography (AbdomenCT), Breast Magnetic Resonance Imaging (BreastMRI), Chest x-ray (CXR), Chest Computer Tomography (ChestCT), Hand x-ray (Hand), and Head Computer Tomography (HeadCT). Sample images from this dataset can be observed in figure 2. Each of these images possesses a resolution of 64×64 pixels, with our primary goal being the classification of each image through the utilization of a neural network.

It is noteworthy that all images within this dataset employ 3 channels, adding an additional layer of complexity compared to the original MNIST dataset. Furthermore, we conducted various data preprocessing techniques, including random rotations of up to 10 degrees, random horizontal flips, and resizing to dimensions as large as 244×244 pixels. These steps were implemented to stabilize the training process and enhance the performance of our models.

Similar to the MNIST classification task, we divided the entire dataset into two subsets: a training set consisting of 47 163 samples and a testing set containing 11 791 samples.

2.1.3. CIFAR-*10*

The CIFAR-10 dataset is a pivotal resource in the domain of computer vision and image classification. Developed as part of the Canadian Institute for Advanced Research (CIFAR) program, this dataset serves as a fundamental benchmark for assessing the performance of machine learning algorithms in the context of image classification tasks [62]. CIFAR-10 is comprised of a total of 60 000 images, which are divided into training and testing sets with 50 000 and 10 000 samples in each respectively. Each image in the CIFAR-10 dataset is of size 32×32 pixels. These images are color images, incorporating three color channels: red, green, and blue. Consequently, every image is represented as a $32 \times 32 \times 3$ tensor. One of the distinguishing features of CIFAR-10 is its categorization into ten distinct classes, each representing a different object category. These classes are as follows: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck. This diverse set of classes ensures that the dataset is suitable for a wide range of image classification challenges,
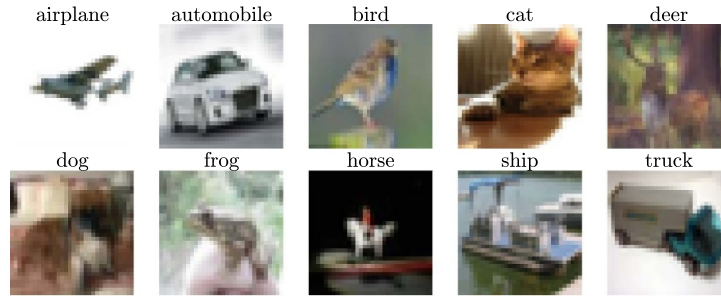
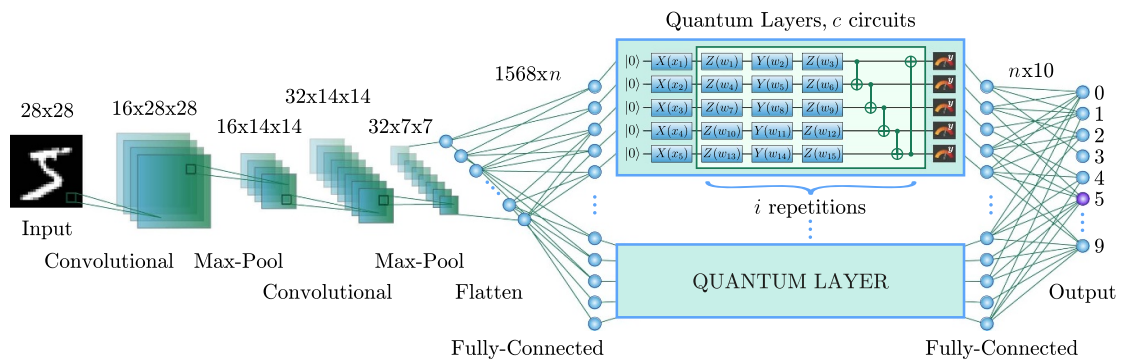**Figure 3.** Examples of images from the CIFAR-10 dataset.

Figure 4. Architecture of the proposed HQNN-Parallel. The input data samples are transformed by a series of convolutional layers, that extract relevant features and reduce the dimensionality of the input. The output channels of the convolutional layers are then flattened into a single vector before being fed into the dense part of the HQNN-Parallel. The hybrid dense part contains a combination of classical and quantum layers. The quantum layers are implemented using parallel PQCs, which allow for simultaneous execution, reducing the total computation time. Quantum layers are depicted in the figure as blue rectangles, the top rectangle is a detailed version of subsequent quantum layers in the amount of c circuits. The output of the last classical fully connected layer is a predicted digit in the range of 0–9.

spanning various domains and object types. As it has more classes than Medical MNIST and input images have bigger resolution than in MNIST, classifying images from CIFAR-10 seems to be complex task. Sample images from this dataset are shown in figure 3.

2.2. HQNN with parallel quantum dense layers, HQNN-Parallel
This section describes our first proposed model, the HQNN with parallel quantum dense layers, each of which is a PQC. Section 2.2.3 presents the results of our comparison between the hybrid model and its classical counterpart, CNN section 2.2.4. The HQNN-Parallel consists of two main components: a classical convolutional block section 2.2.1 and a combination of classical fully connected and parallel quantum layers section 2.2.2. The main purpose of the classical convolutional block is to reduce the dimensionality of the input data and prepare it for further processing. The classical fully connected as well as parallel quantum layers constitute the core of the HQNN-Parallel, and are responsible for prediction tasks of the model. Further details on the architecture and implementation of the HQNN-Parallel are presented in subsequent sections.

2.2.1. Classical convolutional layers
Figure 4 depicts the general structure of the classical convolutional part of the proposed HQNN-Parallel. The convolutional part of the network is comprised of two main blocks, followed by fully-connected layers. In this study, we utilized Rectified Linear Unit (ReLU) as the activation function [63]. Batch Normalization [64] is employed in the network as it stabilizes the training process and improves the accuracy of the model.

The first block of the convolutional part of the HQNN-Parallel comprises a convolutional layer with one input channel and 16 output channels, utilizing a square kernel of size 5×5. The layer operates with a stride of one pixel and applies a two-pixel padding to the input data. We opted for the 5×5 kernel size primarily to maintain the input image's spatial dimensions. By implementing a two-pixel padding, the 5×5 kernel facilitates the creation of 16 channels without altering the original pixel size of 28×28. Additionally, the larger kernel sizes, such as 5×5, can capture more complex and diverse features from the input image

compared to smaller kernels [65, 66]. This aligns with our objective of preserving both spatial and feature details of the image during convolution. Batch Normalization is applied to the output of the convolutional layer, followed by an activation function (ReLU) and MaxPooling [67] with a kernel size of two pixels. The resulting feature map has dimensions of $16 \times 14 \times 14$ pixels.

The second block contains a convolutional layer with 16 input channels and 32 output channels, utilizing the same kernel size and padding as the previous layer. The MaxPooling parameters remain unchanged, resulting in a feature map with dimensions of $32 \times 7 \times 7$ pixels, which will become an input for the fully connected part of the network.

2.2.2. Hybrid dense layers

Following the convolutional block, the HQNN- Parallel, continues with a hybrid dense part, as shown in figure 4. The $32 \times 7 \times 7$ feature map produced by the convolutional part serves as input for the first dense layer, which transforms the feature map from 1568 to n features. The value of n is determined by the chosen quantum part and represents the total number of encoding parameters in the quantum layers.

Each quantum layer is designed to maintain the number of input and output features, and the output of the quantum layer is fed into the second classical fully connected layer. This layer performs the final transformation and maps the n input features to 10 output features, corresponding to the number of classes into which the images can be classified. After each classical dense layer, Batch Normalization and ReLU activation are applied.

It is worth noting that the structure of the HQNN- Parallel, including the number of layers and the number of features, can be adjusted to optimize the performance on a specific task.

2.2.3. Structure of quantum layer

The quantum component of the proposed HQNN-Parallel, depicted in figure 4, consists of c parallel quantum layers, each of which is a PQC composed of three parts: embedding, variational gates, and measurement. The input data to the quantum layers are n features from the previous classical fully connected layer, divided into c parts, with each part being a vector of q values, $x = (\phi_1, \phi_2, \ldots, \phi_q) \in \mathbb{R}^q$. To encode these classical features into quantum Hilbert space, we use the 'angle embedding' method, which rotates each qubit in the ground state around the X-axis on the Bloch sphere [68] by an angle proportional to the corresponding value in the input vector: $|\psi\rangle = R_x^{\mathrm{emb}}(x)|\psi_0\rangle$, where $|\psi_0\rangle = |0\rangle^{\otimes q}$. This operation encodes the input vector into quantum space, and the resulting quantum state represents the input data from the previous classical layer. It is important to note that n is divisible by q since the input data vector is divided into $c = n/q$ parts, with each part serving as input to a PQC.

The encoding part for each PQC is followed by a variational part, which consists of two parts: rotations with trainable parameters and subsequent CNOT operations [69]. The rotations serve as quantum gates that transform the encoded input data according to the variational parameters, while the CNOT operations entangle the qubits in the PQC. The depth of the variational part, denoted as i, is a hyperparameter that determines the number of iterations of the rotations and CNOT operations in the PQC. It is important to note that the variational parameters for each PQC are different in each of the i repetitions and for each of c quantum circuits. Thus, the total number of weights in the quantum part of the HQNN-Parallel is calculated as $q \cdot 3i \cdot c$.

After performing these operations, measurement in the Pauli basis matrices is performed, resulting in

$$\nu^{(j)} = \langle 0| R_x^{\mathrm{emb}}(\phi_j)^\dagger U(\theta)^\dagger Y_j U(\theta) R_x^{\mathrm{emb}}(\phi_j) |0\rangle, \tag{1}$$

where Y_j is the Pauli-Y matrix for the jth qubit, $R_x^{\mathrm{emb}}(\phi_j)$ and $U(\theta)$ are operations, performed by the embedding and trainable parts of the PQC, respectively, and θ is a vector of trainable parameters. After this operation, we have the vector $\nu \in \mathbb{R}^q$. The outputs of all the PQCs would be concatenated to form a new vector $\hat{\nu} \in \mathbb{R}^n$ that is the input data for a subsequent classical fully-connected layer. This layer, being the final layer in the classification pipeline, produces an output in the form of a probability distribution over the set of classes. In our case, each input image is associated with one of the ten possible digits from 0 to 9, and the output of each neuron represents the probability that the image belongs to that class. The neuron with the highest output probability is selected as the predicted class for the image.

More detailed theoretical analysis, involving ZX-calculus reduction and Fourier expressivity of the HQNN-Parallel was conducted in appendix.

2.2.4. Training and results

As described above, the HQNN-Parallel was trained on MNIST dataset section 2.1.1. No preprocessing is applied, so the entire collection is used for training (60 000 images are in the training set and 10 000 are in the

Table 1. Summary of the results for the HQNN-Parallel and its classical analog, CNN.

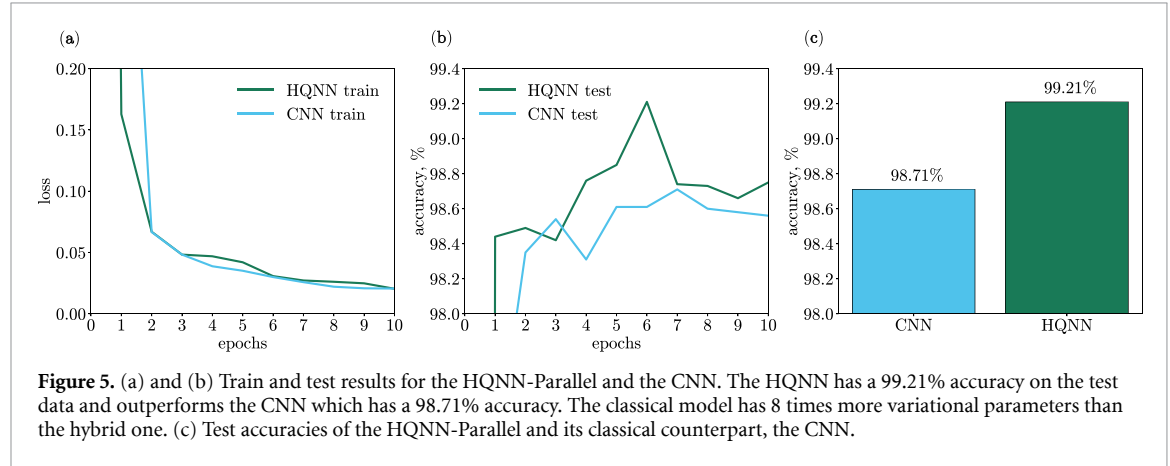| Dataset	Model	train loss	test loss	test acc	param num
MNIST	CNN	0.0205	0.0449	98.71	372 234
	HQNN	0.0204	0.0274	99.21	45 194
Medical MNIST	CNN	$0.456 \cdot 10^{-2}$	$0.396 \cdot 10^{-2}$	82.64	247 642
	HQNN	$0.429 \cdot 10^{-2}$	$0.332 \cdot 10^{-2}$	82.78	247 462
CIFAR-10	CNN	0.3659	0.5484	82.78	81 698
	HQNN	0.3851	0.5208	82.64	81 578

Figure 5. (a) and (b) Train and test results for the HQNN-Parallel and the CNN. The HQNN has a 99.21% accuracy on the test data and outperforms the CNN which has a 98.71% accuracy. The classical model has 8 times more variational parameters than the hybrid one. (c) Test accuracies of the HQNN-Parallel and its classical counterpart, the CNN.

test set). In the context of training the proposed HQNN-Parallel, the ultimate objective is to minimize the loss function during the optimization process. The cross-entropy function is employed as the loss function, given by:

$$l = -\sum_{c=1}^{k} y_c \log p_c, \tag{2}$$

where p_c is the prediction probability, y_c is either 0 or 1, determining respectively if the image belongs to the prediction class, and k is the number of classes.

The parameters of the classical layers are optimized using the backpropagation algorithm [70], which is automatically implemented in the PyTorch library [71]. The backpropagation algorithm is used to calculate the gradients of the loss function with respect to the parameters of the network, allowing for their optimization via gradient descent. However, the use of quantum layers in this task is more complex than classical methods for computing gradients. To overcome this challenge, we employ the PennyLane framework [72], which provides access to a variety of optimization techniques. We utilize the parameter shift rule [73], which is compatible with physical implementations of quantum computing [74]. This method involves evaluating the gradient of a quantum circuit by shifting the parameters in the circuit and computing the corresponding change in the circuit's output. The resulting gradient can then be used to update the circuit's parameters and iteratively minimize the loss function. By using the parameter shift rule, we are able to efficiently optimize the variational parameters in the quantum layers of the HQNN, enabling the network to learn complex patterns in the input data and achieve accurate results.

In the process of solving the problem, we tried various architectures of quantum layers. The most successful architecture for the HQNN-Parallel used a quantum layer with 5 qubits and 3 repetitions of the strongly entangling layers. The number of quantum layers is equal to 4.

The HQNN-Parallel managed to achieve a 99.21% accuracy on MNIST dataset. In order to compare the performance of the HQNN with a classical CNN, the convolutional part of the HQNN was held constant, while the quantum part was replaced with a classical dense layer containing n neurons. This modified CNN was then trained on the same MNIST dataset. A comparison of the training outcomes is depicted in figures 5(a) and (b).

The trainable parameters, as well as the primary training and testing results, for both the HQNN-Parallel and the CNN, are summarized in table 1 and illustrated in figure 5(c). From these results, it is evident that
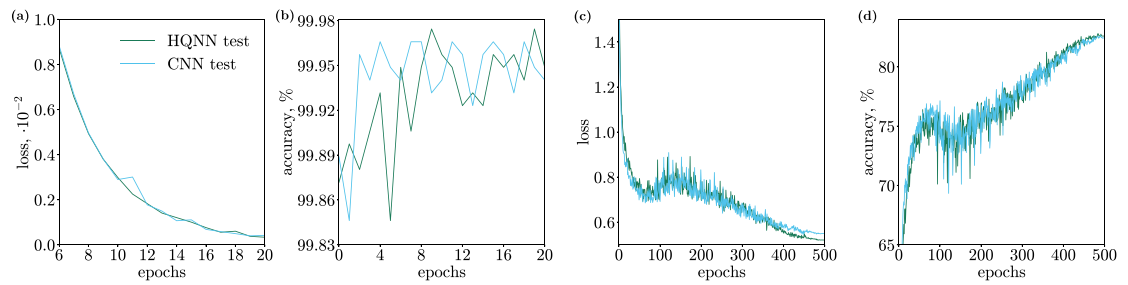
Figure 6. (a) and (b) Test results for the HQNN-Parallel and the CNN on Medical MNIST dataset. The HQNN has a 99.97% accuracy on the test data and slightly outperforms the CNN which has a 99.96% accuracy. (c) and (d) Test results for the HQNN-Parallel and the CNN on CIFAR-10 dataset. The HQNN has a 82.78% accuracy on the test data and outperforms the CNN which has a 82.64% accuracy.
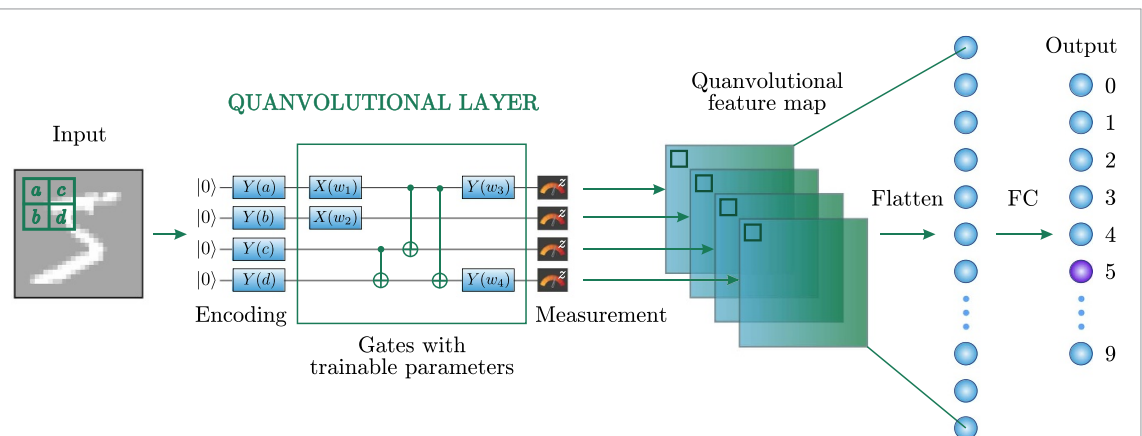

Figure 7. Architecture of the HQNN-Quanv. The Quanvolutional layer maps the input image into 4 Quanvolutional feature maps. These feature maps are then concatenated and flattened to go into the fully-connected classical layer, which gives us 10 probabilities for each class.

the most successful implementation of the HQNN-Parallel surpasses the performance of a CNN that possesses approximately eight times more parameters.

HQNN-Parallel with its classical analog were also tested on Medical MNIST dataset. The HQNN-Parallel managed to achieve a 99.97% accuracy. The classical CNN showed less accurate results with 99.96% accuracy on test data. It is worth noting that HQNN model had 247 462 trainable parameters and classical CNN had 247 642. A comparison of the training outcomes is depicted in figures 6(a) and (b).

To confirm the generalizability of hybrid architecture we tested hybrid and classical models on CIFAR-10 dataset. The HQNN-Parallel managed to achieve an 82.78% accuracy. The classical CNN showed less accurate results with 82.64% accuracy on test data. It is worth noting that HQNN model had 81 578 trainable parameters and classical CNN had 81 698. A comparison of the training outcomes is depicted in figures 6(c) and (d).

In this section, we provide a comprehensive overview of the model's architecture employed during training on the MNIST dataset. While the foundational structure remained consistent when testing on both the Medical MNIST and CIFAR-10 datasets, there were variations. Specifically, input dimensions varied based on image sizes. Additionally, the number of convolutions differed: two for Medical MNIST and three for CIFAR-10. Furthermore, the neuron count in the output layer was adjusted in line with the respective number of classification classes. The quantity of qubits and quantum layers remained unchanged across all experiments.

2.3. HQNN with quanvolutional layer, HQNN-Quanv

In this section, we give a detailed description of our second hybrid quantum approach for solving the problem of recognizing the numbers from the MNIST dataset, based on the combination of a quanvolutional layer and classical fully connected layers. The scheme of this network is presented in figure 7. We also

compare our hybrid model with its classical analog CNN and investigate the relationship between quanvolutional and convolutional layers, as well as their dependence on the number of output channels.

2.3.1. Quanvolutional layer

The general architecture of a quanvolutional layer [75] is shown in figure 7. Similar to classical convolutional layers, the quanvolutional layer comprises a kernel of size $n \times n$ pixels that convolve the input image, producing a lower-resolution output image. However, the quanvolutional layer is unique in the sense that its kernel is implemented using a quantum circuit consisting of n_q qubits. The circuit can be decomposed into three distinct parts: classical-to-quantum data encoding, variational gates, and quantum measurement. These parts work together to determine the kernel's action on the input image.

There are plenty of encoding (embedding) methods to transfer classical data into quantum states. In this section, as in the previous one, we use the 'angle embedding' technique. It is achieved by rotating the qubits from their initial $|0\rangle$ value with the $R_y(\varphi)$ unitaries, where φ is determined by the value of the corresponding pixel. After the classical data is encoded, the quantum states undergo unitary transformations, defined by the variational part.

The variational part in the quanvolutional layer usually consists of arbitrary single-qubit rotations and CNOT gates, arranged in a particular way determined by the researcher. The unitaries in the PQC are parameterized by a set of variational parameters, which are learned via training the neural network. The ultimate goal of the model training is to find a measurement basis (by tweaking variational gate parameters) that tells us the most information about a fragment of a picture confined by the quantum kernel.

Finally, for each wire, the expectation value of an arbitrary operator is calculated to obtain the classical output. As it is a real number, it represents the kernel's output pixel, while each wire yields a different image channel. For instance, a quanvolutional kernel of size 2×2 has a 4-qubit circuit, which transforms one input image into four images of reduced size.

2.3.2. Structure of HQNN-Quanv

This subsection details the architecture of the HQNN-Quanv, which is shown in figure 7. At first, a simple angle embedding of the classical data via $R_y(\varphi)$ single-qubit rotations on each wire is used, where the original pixel value $[0,1]$ is scaled to $\varphi \in [0, \pi]$. Then, we have a variational circuit part, which consists of 4 single-qubit rotations, parameterized with trainable weights, as well as three CNOT gates. At the end of the circuit, we measure the expectation value $\langle \sigma_z \rangle$ of the Pauli-Z operator on each qubit. Each channel is a picture with 4×4 pixels. After that, four output channels are flattened and fed into a fully connected layer, which yields a digit's probability.
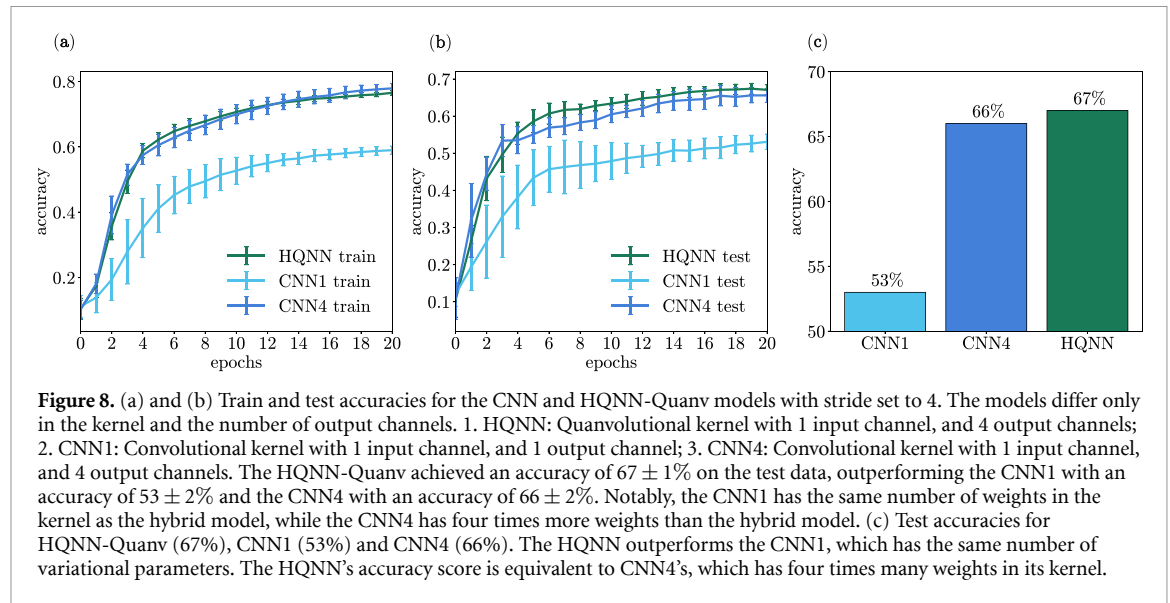
2.3.3. Training and results

In this section, we describe the training process. In order to reduce the training time of the HQNN-Quanv, only 600 images from the MNIST dataset section 2.1.1 are used with 500 of them acting as training data and 100 as test data. We also use PyTorch's resize transform with bilinear interpolation to downscale images from 28×28 to 14×14 pixels. We still use a cross-entropy loss function.

While the classical model has only one way of training weights via backpropagation, the HQNN has several options, such as the parameter-shift rule, adjoint differentiation [76] or backpropagation (which, of course, is impossible on a real quantum computer). Adjoint differentiation seems to have the most favourable scaling with both layers and wires [77], but on this particular circuit (figure 7) backpropagation proved to be quicker.

Considering everything stated above, let us see the results of the training. We trained two CNNs with different numbers of output channels and one HQNN for 20 epochs (figures 8(a) and (b)). The models were intentionally made simple and had sufficiently few parameters so as to avoid overfitting on the relatively small dataset. Test accuracies of these models are presented in figure 8(c). For each epoch, the accuracy is averaged over 10 models with random initial weights. The error bars depict one standard deviation.

At the end of the training, the HQNN-Quanv had a test accuracy of 0.67 ± 0.01, which is close enough to the CNN4 result of 0.66 ± 0.02, while CNN1 had 0.53 ± 0.02. The HQNN model has only 4 trainable weights in its quanvolutional kernel, which parameterizes rotation gates in the PQC. CNN1 and CNN4 have 4 and 16 trainable parameters in their convolutional kernels, respectively. Therefore, the HQNN's performance based on the accuracy score is equivalent to CNN4's, which has four times many weights in its kernel.

Figure 8. (a) and (b) Train and test accuracies for the CNN and HQNN-Quanv models with stride set to 4. The models differ only in the kernel and the number of output channels. 1. HQNN: Quanvolutional kernel with 1 input channel, and 4 output channels; 2. CNN1: Convolutional kernel with 1 input channel, and 1 output channel; 3. CNN4: Convolutional kernel with 1 input channel, and 4 output channels. The HQNN-Quanv achieved an accuracy of $67 \pm 1\%$ on the test data, outperforming the CNN1 with an accuracy of $53 \pm 2\%$ and the CNN4 with an accuracy of $66 \pm 2\%$. Notably, the CNN1 has the same number of weights in the kernel as the hybrid model, while the CNN4 has four times more weights than the hybrid model. (c) Test accuracies for HQNN-Quanv (67%), CNN1 (53%) and CNN4 (66%). The HQNN outperforms the CNN1, which has the same number of variational parameters. The HQNN's accuracy score is equivalent to CNN4's, which has four times many weights in its kernel.

3. Discussion

In this work, we introduced two hybrid approaches to image classification. The first approach was an HQNN-Parallel. This method allowed us to classify handwritten images of digits from the MNIST dataset with an accuracy of more than 99%. The classical model achieved a similar performance of 98.71% and has eight times more weights in a neural network. We also tested this model on the Medical MNIST, where we achieved a quality of over 99%, and on CIFAR-10, where we showed that the hybrid model classifies images with an accuracy of over 82% better than its classical counterpart with a similar number of weights. These examples confirm the generalizability of the HQNN-Parallel model. Also, the successful implementation of parallel PQCs in the hybrid model was demonstrated, which led to such remarkable results. Our proposed architecture is a unique combination of classical and quantum layers, which we believe to be a breakthrough in solving image classification problems.

The second approach we presented was an HQNN-Quanv. The quanvolutional layer uses fewer weights, four times less than the classical analog, to achieve approximately the same classification accuracy ($67 \pm 1\%$ for the hybrid model versus $66 \pm 2\%$ for the classical one on the test samples when averaged over ten models), while the classical analog with the same number of variational parameters as the hybrid model achieves an accuracy of $53 \pm 2\%$.

Hybrid quantum approaches developed in this work often had significantly fewer weights in the corresponding neural networks. However, the reduced number of weights does not imply increased efficiency of the hybrid approach due to the slower training of the hybrid model compared to the classical one with the same number of weights. In practice, to obtain a practical advantage with the fewer number of weights, more efficient quantum computers or simulators of quantum computers are required.

Our research, conducted during the NISQ era, navigated the inherent constraints of current quantum circuits, such as noise levels and qubit entanglement limitations. Relying on a hybrid quantum–classical approach, the study was designed with current quantum hardware in mind and assumed seamless integration of quantum circuits within classical layers. Despite these considerations, our models demonstrated quantum superiority across three datasets, including MNIST. However, we recognize the need for broader validation to ensure holistic generalization across diverse datasets and real-world scenarios. Further research is needed to explore the full potential of HQNNs for image classification, including testing more complex architectures. Additionally, the development of more efficient optimization techniques for training PQCs and the implementation of larger-scale quantum hardware could lead to even more significant performance improvements.

In summary, our developments provide two hybrid approaches to image classification that demonstrate the power of combining classical and quantum methods. Our proposed models show improved performance over classical models with similar architectures. We believe that these results pave the way for further research in developing hybrid models that utilize the strengths of both classical and quantum computing.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Appendix. Theoretical analysis

This section theoretically analyzes the quantum layers used in the HQNN-Parallel model in section 2.2. We focus on the methodologies of the ZX-calculus [78] to explore circuit reducibility, and Fourier accessibility [79] to examine the data embedding strategy and expressivity.

A.1. ZX-calculus reduction

ZX-calculus is a graphical language that replaces circuit diagrams with ZX-diagrams by replacing quantum tensors with so-called 'spiders', nodes on a graph with edges that connect them [78, 80, 81]. These spiders come in two flavors, a light or green-colored spider that represents tensors in the Z basis ($|0\rangle, |1\rangle$) and a dark or red-colored spider that represents tensors in the X basis ($|+\rangle, |-\rangle$). ZX-diagrams can be simplified and reduced with the language's graphical rewrite rules based on the underlying quantum operations. For example, repetitions of Pauli rotations sum together to form one Pauli rotation with an angle equal to the sum of its parts. This is translated into ZX-calculus as a specific instance of the more general rule of 'fusing' spiders, where nodes of the same color combine and sum their angles. More generally, quantum operations often possess subtle symmetries that make it difficult to implement effective circuits, and for exponentially large systems, matrix multiplication quickly becomes unwieldy. Essentially, ZX-calculus replaces tedious matrix multiplication of quantum gates with easy-to-apply graphical rules. Thus, analysis of ZX-diagrams are helpful for identifying redundancies in a quantum model.

To analyze the reduced quantum layer of figure 9(a), we first represented it as a red-green ZX-diagram. Then ZX-calculus's rewriting rules are applied to simplify the circuit and remove redundancies. Finally, the resulting new circuit is extracted. Figure 9(b) shows the simplified circuit in ZX-form. Note that the reduced ZX-form still has all the initial parameters w_i on separate nodes. This indicates that the model does not reduce away any of the inputs or trainable parameters. This verifies that all the weights and input data in the circuit actually make an impact on the final result.

A.2. Fourier expressivity

Schuld *et al* [79] showed that the output of a PQC is equivalent to a truncated Fourier series. For a feature vector of length N, the Fourier series as a function of the feature vector **x** and trainable parameters θ is:

$$f_\theta(\mathbf{x}) = \sum_{\omega_1 \in \Omega_1} \cdots \sum_{\omega_N \in \Omega_N} c_{\omega_1 \ldots \omega_N}(\theta) e^{-i\omega \cdot \mathbf{x}},$$

where $\omega_i \in \{-d_i, \ldots, 0, \ldots, d_i\}$. In other words, the number of terms in the Fourier series is one more than twice the number of times that input was placed in the circuit, d. In this analysis, we show the expressivity of the function $f_\theta(\mathbf{x})$ by sampling over a uniform distribution of random values for each θ_i from $[0, 2\pi]$ and by sampling equidistant x values with a sampling frequency of d.

For visual clarity, we display only the first two terms (associated with the first two features) of the model on the final output of the quantum circuit. If we write these inputs as x and y, the output function $f_\theta(\mathbf{x})$ becomes,

$$f_\theta(x, y) = \sum_{\omega_x = -1}^{1} \sum_{\omega_y = -1}^{1} c_{\omega_x, \omega_y}(\theta)\, e^{-i\omega_x x} e^{-i\omega_y y}.$$

Figure 10 demonstrates a violin plot of the Fourier coefficients c_{ω_x, ω_y} sampled over various θ realizations. A completely non-expressive model would have terms close to zero for all these coefficients. Instead, the figure shows that the weights of the quantum layer have a wide range of possible solutions. Additionally, the determinate of the correlation matrix of all values was equal to zero for every output, demonstrating the independent and expressive nature of all the Fourier terms in the model.
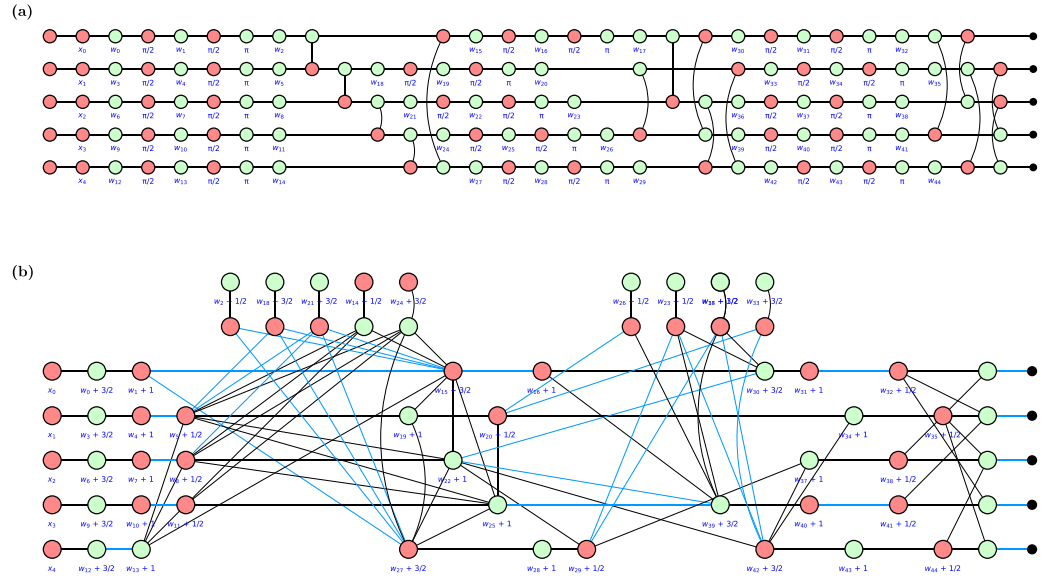
Figure 9. ZX-calculus graphs. (a) The quantum layer from HQNN-Parallel (Fig. 4) written in ZX-calculus. (b) The same quantum layer as in part (a) that has been reduced with ZX rewriting rules. All weights and inputs are on independent nodes, which suggests the model is absent of redundancy.
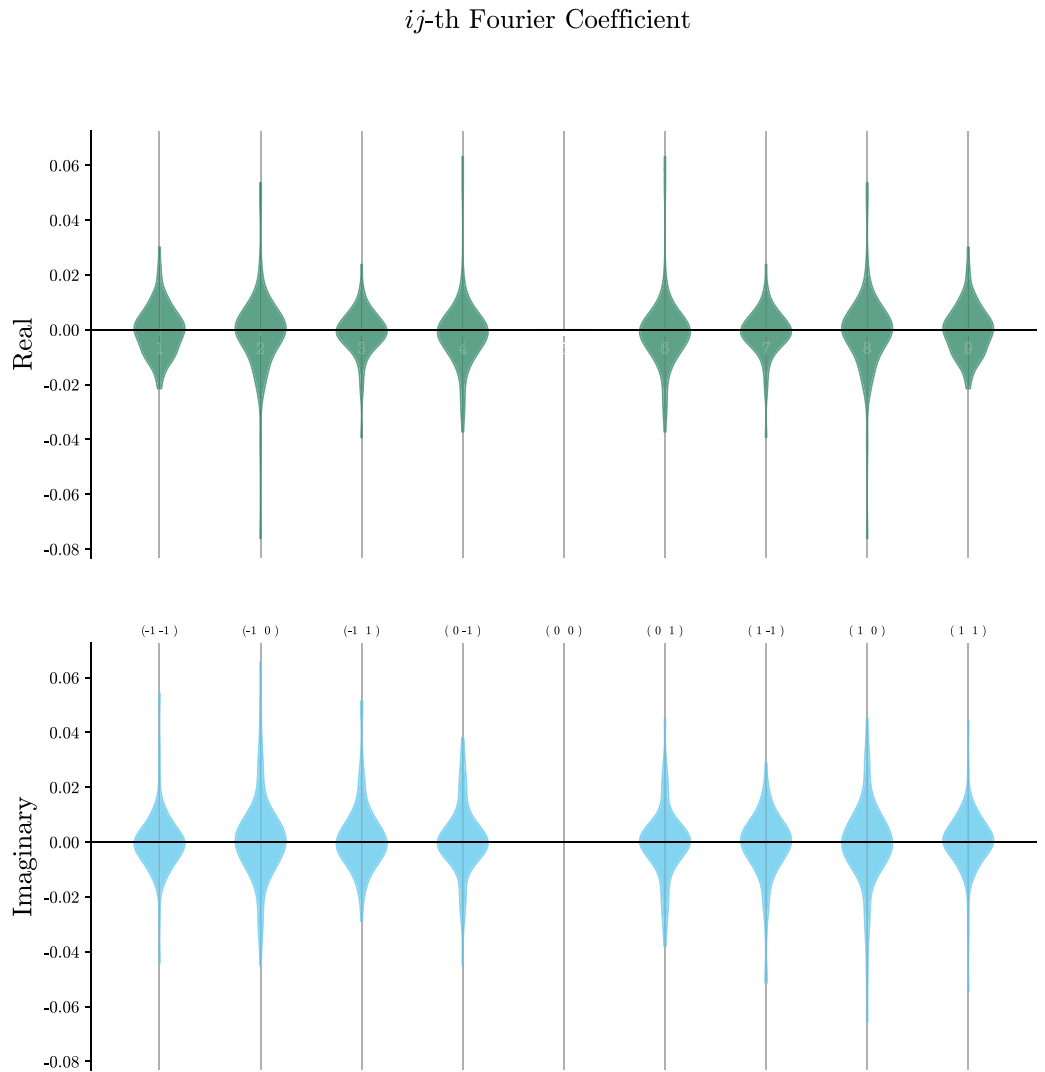

Figure 10. A violin chart of 100 samples of the values of the Fourier coefficients for the first and second input parameters of the final output measurement. The ij-th indices along the center line represent the Fourier coefficient c_{ij}. The width of the violins represents the number of samples at that magnitude. The large spread on both the real and imaginary part of every coefficient implies high expressivity in the model.

ORCID iDs

Arsenii Senokosov ● https://orcid.org/0009-0008-2383-7475
Alexandr Sedykh ● https://orcid.org/0009-0003-8431-0873
Asel Sagingalieva ● https://orcid.org/0009-0009-3931-3702
Basil Kyriacou ● https://orcid.org/0009-0006-5619-4432
Alexey Melnikov ● https://orcid.org/0000-0002-5033-4063

References

[1] Ren S, He K, Girshick R and Sun J 2015 Faster R-CNN: towards real-time object detection with region proposal networks (arXiv:1506.01497)
[2] Litjens G, Kooi T, Ehteshami Bejnordi B, Arindra Adiyoso Setio A, Ciompi F, Ghafoorian M, van der Laak J A W M, van Ginneken B and Sánchez C I 2017 A survey on deep learning in medical image analysis *Med. Image Anal.* **42** 60–88
[3] Tian L, Hunt B, Lediju Bell M A, Yi J, Smith J T, Ochoa M, Intes X and Durr N J 2021 Deep learning in biomedical optics *Lasers Surg. Med.* **53** 748–75
[4] LeCun Y, Bengio Y and Hinton G 2015 Deep learning *Nature* **521** 436–44
[5] Badrinarayanan V, Kendall A and Cipolla R 2015 SegNet: a deep convolutional encoder-decoder architecture for image segmentation (arXiv:1511.00561)
[6] Huang G, Liu Z, van der Maaten L and Weinberger K Q 2016 Densely connected convolutional networks (arXiv:1608.06993)
[7] Krizhevsky A, Sutskever I and Hinton G E 2012 Imagenet classification with deep convolutional neural networks *Advances in Neural Information Processing Systems* vol 25, ed F Pereira, C J Burges, L Bottou and K Q Weinberger (Curran Associates, Inc)
[8] Redmon J and Farhadi A 2016 YOLO9000: better, faster, stronger (arXiv:1612.08242)
[9] Ladd T D, Jelezko F, Laflamme R, Nakamura Y, Monroe C R and Lloyd O'Brien J 2010 Quantum computers *Nature* **464** 45–53
[10] Schuld M, Sinayskiy I and Petruccione F 2015 An introduction to quantum machine learning *Contemp. Phys.* **56** 172–85
[11] Biamonte J, Wittek P, Pancotti N, Rebentrost P, Wiebe N and Lloyd S 2017 Quantum machine learning *Nature* **549** 195–202
[12] Dunjko V and Briegel H J 2018 Machine learning and artificial intelligence in the quantum domain: a review of recent progress *Rep. Prog. Phys.* **81** 0074001
[13] Melnikov A, Kordzanganeh M, Alodjants A and Lee R-K 2023 Quantum machine learning: from physics to software engineering *Adv. Phys.* X **8** 2165452
[14] Cong I, Choi S and Lukin M D 2019 Quantum convolutional neural networks *Nat. Phys.* **15** 1273–8
[15] Hur T, Kim L and Park D K 2022 Quantum convolutional neural network for classical data classification *Quantum Mach. Intell.* **4** 3
[16] Herrmann J *et al* 2022 Realizing quantum convolutional neural networks on a superconducting quantum processor to recognize quantum phases *Nat. Commun.* **13** 4144
[17] Schuld M, Fingerhuth M and Petruccione F 2017 Implementing a distance-based classifier with a quantum interference circuit (arXiv:1703.10793)
[18] Bokhan D, Mastiukova A S, Boev A S, Trubnikov D N and Fedorov A K 2022 Multiclass classification using quantum convolutional neural networks with hybrid quantum-classical learning *Front. Phys.* **10** 1069985
[19] Lloyd S, Mohseni M and Rebentrost P 2013 Quantum algorithms for supervised and unsupervised machine learning (arXiv:1307.0411)
[20] Shor P W 1994 Algorithms for quantum computation: discrete logarithms and factoring *Proc. 35th Annual Symp. on Foundations of Computer Science* pp 124–34
[21] Lloyd S 1996 Universal quantum simulators *Science* **273** 1073–8
[22] Aaronson S and Chen L 2016 Complexity-theoretic foundations of quantum supremacy experiments (arXiv:1612.05903)
[23] Wiebe N, Kapoor A and Svore K M 2014 Quantum deep learning (arXiv:1412.3489)
[24] Havlíček V, Córcoles A D, Temme K, Harrow A W, Kandala A, Chow J M and Gambetta J M 2019 Supervised learning with quantum-enhanced feature spaces *Nature* **567** 209–12
[25] Liu J, Hui Lim K, Wood K L, Huang W, Guo C and Huang H-L 2021 Hybrid quantum-classical convolutional neural networks *Sci. China Phys. Mech. Astron.* **64** 290311
[26] Li W, Chu P-C, Liu G-Z, Tian Y-B, Qiu T-H and Wang S-M 2022 An image classification algorithm based on hybrid quantum classical convolutional neural network *Quantum Eng.* **2022** 5701479
[27] Houssein E H, Abohashima Z, Elhoseny M and Mohamed W M 2022 Hybrid quantum-classical convolutional neural network model for COVID-19 prediction using chest x-ray images *J. Comput. Des. Eng.* **9** 343–63
[28] Huang S-Y, An W-J, Zhang D-S and Zhou N-R 2023 Image classification and adversarial robustness analysis based on hybrid quantum–classical convolutional neural network *Opt. Commun.* **533** 129287
[29] Sagingalieva A, Kordzanganeh M, Kurkin A, Melnikov A, Kuhmistrov D, Perelshtein M, Melnikov A, Skolik A and Dollen D V 2023 Hybrid quantum ResNet for car classification and its hyperparameter optimization *Quantum Mach. Intell.* **5** 38
[30] Dong Y, Fu Y, Liu H, Che X, Sun L and Luo Y 2023 An improved hybrid quantum-classical convolutional neural network for multi-class brain tumor MRI classification *J. Appl. Phys.* **133** 064401
[31] Zhou N-R, Zhang T-F, Xie X-W and Wu J-Y 2023 Hybrid quantum–classical generative adversarial networks for image generation via learning discrete distribution *Signal Process. Image Commun.* **110** 116891
[32] Lok Tsang S, West M T, Erfani S M and Usman M 2022 Hybrid quantum-classical generative adversarial network for high resolution image generation (arXiv:2212.11614)
[33] Huang H-L *et al* 2021 Experimental quantum generative adversarial networks for image generation *Phys. Rev. Appl.* **16** 024051
[34] Rudolph M S, Bashige Toussaint N, Katabarwa A, Johri S, Peropadre B and Perdomo-Ortiz A 2022 Generation of high-resolution handwritten digits with an ion-trap quantum computer *Phys. Rev.* X **12** 031010
[35] Perelshtein M *et al* 2022 Practical application-specific advantage through hybrid quantum computing (arXiv:2205.04858)
[36] Alam M, Kundu S, Onur Topaloglu R and Ghosh S 2021 Quantum-classical hybrid machine learning for image classification (ICCAD special session paper) (arXiv:2109.02862)

[37] Trochun Y, Stirenko S, Rokovyi O, Alienin O, Pavlov E and Gordienko Y 2021 Hybrid classic-quantum neural networks for image classification *2021 11th IEEE Int. Conf. on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)* vol 2 pp 968–72

[38] Marshall S C, Gyurik C and Dunjko V 2022 High dimensional quantum learning with small quantum computers (arXiv:2203.13739)

[39] Kordzanganeh M, Sekatski P, Fedichkin L and Melnikov A 2023 An exponentially-growing family of universal quantum circuits *Mach. Learn.: Sci. Technol.* **4** 035036

[40] Pérez-Salinas A, Draškić R, Tura J and Dunjko V 2022 Reduce-and-chop: shallow circuits for deeper problems (arXiv:2212.11862)

[41] Kordzanganeh M, Kosichkina D and Melnikov A 2023 Parallel hybrid networks: an interplay between quantum and classical neural networks *Intell. Comput.* **2** 0028

[42] Jerbi S, Fiderer L J, Poulsen Nautrup H, Kübler J M, Briegel H J and Dunjko V 2023 Quantum machine learning beyond kernel methods *Nat. Commun.* **14** 517

[43] Li Y, Zhou R-G, Xu R, Luo J and Hu W 2020 A quantum deep convolutional neural network for image recognition *Quantum Sci. Technol.* **5** 044003

[44] Mitarai K, Negoro M, Kitagawa M and Fujii K 2018 Quantum circuit learning *Phys. Rev. A* **98** 032309

[45] Domingo L, Djukic M, Johnson C and Borondo F 2023 Hybrid quantum-classical convolutional neural networks to improve molecular protein binding affinity predictions (arXiv:2301.06331)

[46] Sagingalieva A, Kordzanganeh M, Kenbayev N, Kosichkina D, Tomashuk T and Melnikov A 2023 Hybrid quantum neural network for drug response prediction *Cancers* **15** 2705

[47] Jain P and Ganguly S 2022 Hybrid quantum generative adversarial networks for molecular simulation and drug discovery (arXiv:2212.07826)

[48] Sedykh A, Podapaka M, Sagingalieva A, Smertyak N, Pinto K, Pflitsch M and Melnikov A 2023 Quantum physics-informed neural networks for simulating computational fluid dynamics in complex shapes (arXiv:2304.11247)

[49] Kurkin A, Hegemann J, Kordzanganeh M and Melnikov A 2023 Forecasting the steam mass flow in a powerplant using the parallel hybrid network (arXiv:2307.09483)

[50] Paquet E and Soleymani F 2022 QuantumLeap: hybrid quantum neural network for financial predictions *Expert Syst. Appl.* **195** 116583

[51] Rainjonneau S *et al* 2023 Quantum algorithms applied to satellite mission planning for Earth observation *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **16** 7062–75

[52] Haboury N, Kordzanganeh M, Schmitt S, Joshi A, Tokarev I, Abdallah L, Kurkin A, Kyriacou B and Melnikov A 2023 A supervised hybrid quantum machine learning solution to the emergency escape routing problem (arXiv:2307.15682)

[53] Farhi E and Neven H 2018 Classification with quantum neural networks on near term processors (arXiv:1802.06002)

[54] Baek H, Joon Yun W and Kim J 2022 Scalable quantum convolutional neural networks (arXiv:2209.12372)

[55] Chen G, Chen Q, Long S, Zhu W, Yuan Z and Wu. Y 2023 Quantum convolutional neural network for image classification *Pattern Anal. Appl.* **26** 655–67

[56] Deng Li 2012 The MNIST database of handwritten digit images for machine learning research *IEEE Signal Process. Mag.* **29** 141–2

[57] Claudiu Cireşan D, Meier U, Maria Gambardella L and Schmidhuber J 2010 Deep, big, simple neural nets for handwritten digit recognition *Neural Comput.* **22** 3207–20

[58] Ciresan D, Meier U and Schmidhuber J 2012 Multi-column deep neural networks for image classification *2012 IEEE Conf. on Computer Vision and Pattern Recognition* vol 1 pp 3642–9

[59] Das M and Bolisetti T 2023 Variational Quantum Neural Networks (VQNNS) in image classification (arXiv:2303.05860)

[60] Landman J, Mathur N, Li Y Y, Strahm M, Kazdaghli S, Prakash A and Kerenidis I 2022 Quantum methods for neural networks and application to medical image classification *Quantum* **6** 881

[61] apolanco3225 2017 Medical MNIST classification (available at: https://github.com/apolanco3225/Medical-MNIST-Clas sification)

[62] Krizhevsky A *et al* 2009 Learning multiple layers of features from tiny images *Master's thesis* University of Toronto

[63] Fred Agarap A 2018 Deep learning using rectified linear units (ReLU) (arXiv:1803.08375)

[64] Ioffe S and Szegedy C 2015 Batch normalization: accelerating deep network training by reducing internal covariate shift (arXiv:1502.03167)

[65] Tan M and Le Q V 2019 Mixconv: mixed depthwise convolutional kernels (arXiv:1907.09595)

[66] Tan M, Chen B, Pang R, Vasudevan V, Sandler M, Howard A and Le Q V 2019 Mnasnet: platform-aware neural architecture search for mobile *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition* pp 2820–8

[67] LeCun Y, Bottou L, Bengio Y and Haffner P 1998 Gradient-based learning applied to document recognition *Proc. IEEE* **86** 2278–324

[68] Bloch F 1946 Nuclear induction *Phys. Rev.* **70** 460–74

[69] Barenco A, Bennett C H, Cleve R, DiVincenzo D P, Margolus N, Shor P, Sleator T, Smolin J A and Weinfurter H 1995 Elementary gates for quantum computation *Phys. Rev. A* **52** 3457–67

[70] Rumelhart D E, Hinton G E and Williams R J 1986 Learning representations by back-propagating errors *Nature* **323** 533–6

[71] Paszke A *et al* 2019 Pytorch: an imperative style, high-performance deep learning library *Advances in Neural Information Processing Systems 32* (Curran Associates, Inc) pp 8024–35

[72] Bergholm V *et al* 2018 PennyLane: automatic differentiation of hybrid quantum-classical computations (arXiv:1811.04968)

[73] Wierichs D, Izaac J, Wang C and Yen-Yu Lin C 2022 General parameter-shift rules for quantum gradients *Quantum* **6** 677

[74] Kordzanganeh M, Buchberger M, Kyriacou B, Povolotskii M, Fischer W, Kurkin A, Somogyi W, Sagingalieva A, Pflitsch M and Melnikov A 2023 Benchmarking simulated and physical quantum processing units using quantum and hybrid algorithms *Adv. Quantum Technol.* **6** 2300043

[75] Henderson M, Shakya S, Pradhan S and Cook T 2019 Quanvolutional neural networks: powering image recognition with quantum circuits (arXiv:1904.04767)

[76] Jones T and Gacon J 2020 Efficient calculation of gradients in classical simulations of variational quantum algorithms (arXiv:2009.02823)

[77] Luo X-Z, Liu J-G, Zhang P and Wang L 2020 Yao.jl: extensible, efficient framework for quantum algorithm design *Quantum* **4** 341

[78] Coecke B and Duncan R 2011 Interacting quantum observables: categorical algebra and diagrammatics *New J. Phys.* **13** 043016

[79] Schuld M, Sweke R and Jakob Meyer J 2021 Effect of data encoding on the expressive power of variational quantum-machine-learning models *Phys. Rev. A* **103** 032430

[80] van de Wetering J 2020 ZX-calculus for the working quantum computer scientist (arXiv:2012.13966)

[81] Wang Q 2023 Completeness of the ZX-calculus (arXiv:2209.14894)