Quantum Machine Intelligence (2025) 7:35
https://doi.org/10.1007/s42484-025-00247-7

RESEARCH ARTICLE

Quantum Hamiltonian embedding of images for data reuploading classifiers

Peiyong Wang[1] · **Casey R. Myers**[1,2] · **Lloyd C. L. Hollenberg**[3] · **Udaya Parampalli**[1]

Received: 22 July 2024 / Accepted: 30 January 2025

Abstract

When applying quantum computing to machine learning tasks, one of the first considerations is the design of the quantum machine learning model itself. Conventionally, the design of quantum machine learning algorithms relies on the "quantisation" of classical learning algorithms, such as using quantum linear algebra to implement important subroutines of classical algorithms, if not the entire algorithm, seeking to achieve a quantum advantage through possible run-time accelerations brought by quantum computing. However, recent research has started questioning whether quantum advantage via speedup is the right goal for quantum machine learning (Schuld and Killoran 2022 PRX Quantum 3(3):030101.). Research also has been undertaken to exploit properties that are unique to quantum systems, such as quantum contextuality, to better design quantum machine learning models (Bowles et al. 2023). In this paper, we take an alternative approach by incorporating the heuristics and empirical evidences from the design of classical deep learning algorithms to the design of quantum neural networks. We first construct a model based on the data reuploading circuit (Pérez-Salinas et al. 2020 Quantum 4(226):226) with the quantum Hamiltonian data embedding unitary (Schuld and Petruccione 2021). Through numerical experiments on image datasets, including the famous MNIST and FashionMNIST datasets, we demonstrate that our model outperforms the quantum convolutional neural network (QCNN) (Cong et al. 2019 Nat Phys 15(12):1273–1278) by a large margin (up to over 40% on MNIST test set). Based on the model design process and numerical results, we then laid out six principles for designing quantum machine learning models, especially quantum neural networks.

Keywords Quantum machine learning · Image classification · Quantum neural networks · MNIST · FashionMNIST

1 Introduction

As a key application area of quantum computing, quantum machine learning (Biamonte et al. 2017) has received considerable attention as an area that may achieve a potential quantum advantage compared to classical machine learning/deep learning algorithms through runtime acceleration. The quest for achieving such an acceleration has become a standard motivation in the development of quantum machine learning algorithms. The effectiveness of such motivation has been demonstrated by the use of efficient quantum subroutines that could accelerate linear algebra calculations, such as the quantum principal component analysis algorithm (qPCA), which involves calculations of the eigenvalues and eigenvectors of a covariance matrix by quantum phase estimation (Lloyd et al. 2014).

✉ Peiyong Wang
peiyongw@student.unimelb.edu.au

✉ Udaya Parampalli
udaya@unimelb.edu.au

Casey R. Myers
casey.myers@uwa.edu.au

Lloyd C. L. Hollenberg
lloydch@unimelb.edu.au

[1] School of Computing and Information Systems, Faculty of Engineering and Information Technology, The University of Melbourne, Melbourne, Australia

[2] School of Physics, Mathematics and Computing, The University of Western Australia, Perth, Australia

[3] School of Physics, The University of Melbourne, Melbourne, Australia

Published online: 04 March 2025

Unlike principal component analysis and kernel methods, which are often referred to as statistical learning algorithms, neural networks, with their ability to discover hidden patterns in large-scale unstructured datasets such as images and natural language, have gained popularity since the invention of AlexNet (Krizhevsky et al. 2024) and have become the foundation of modern artificial intelligence applications such as ChatGPT-4 (Bubeck et al. 2023). However, since time complexity is rarely the first priority during the design of novel deep neural network architectures, which often rely on intuition and even inspirations from biological neural networks, it becomes less obvious that quantum computing should find any advantage or utility in deep learning and AI.

Although recent research has attempted to integrate properties that are unique to quantum systems, such as contextuality, into the design of quantum machine learning models for specific types of tasks that could lead to quantum advantage (Bowles et al. 2023), few studies have taken into account the intuition behind successful deep learning models and how to integrate them into quantum machine learning models. This serves as the main motivation for our research. In this paper, we aim to bridge this gap by bringing this intuition to the design of quantum machine learning models, especially quantum neural networks, via numerical experiments for the design of a quantum machine learning model for benchmarking image processing tasks.

Our main contributions in this paper are as follows.

- A quantum classifier based on the quantum Hamiltonian embedding approach and the data reuploading circuit for image classification tasks that could outperform the baseline quantum convolutional neural network model (Cong et al. 2019).
- Based on the model design process and the numerical experiments, we lay out a set of guiding principles for future quantum machine learning (QML) model design.

The results of our paper further emphasise the importance of heuristics during the design of quantum machine learning models, especially heuristics and empirical knowledge found in the extensive classical deep learning literature.

This paper is organised as follows: In the rest of this section, we briefly introduce the relevant research in applying quantum machine learning to image processing, as well as common quantum data embedding approaches. In Section 2, we propose a quantum classification model based on the data reuploading circuit (Pérez-Salinas et al. 2020) and quantum Hamiltonian embedding method (Schuld and Petruccione 2021). In Section 3, we demonstrate the effectiveness of this model by evaluating the classification performance on different datasets, including the famous MNIST (LeCun et al. 2010) and FashionMNIST (Xiao et al. 2017) datasets. In Section 4, we discuss the results obtained through numerical experiments which inform our proposed six basic principles for the design of quantum neural networks.

1.1 QML for image processing

As an illustration of quantum neural network (QNN) design, we consider one of the most important tasks in modern artificial intelligence—image processing, including image classification, segmentation, and generation. Since the success of AlexNet (Krizhevsky et al. 2024) at the ImageNet Large-Scale Visual Recognition Challenge 2012 (ILSVRC 2012), deep neural networks, especially deep convolutional neural networks (CNN), have dominated image-related tasks. Recently, the vision transformer (ViT) (Dosovitskiy et al. 2021) and its variants are trending in image-related tasks due to its structural compatibility with large language models and the potential to build a single unified multi-modal model. An important step in the vision transformer is to cut the image into patches, following the same inductive bias as convolutional neural networks. From the history of image processing with deep neural networks, we can see that there is a central principle in the network architectures designed throughout the years, which is the locality of information and translation invariance. This is reflected in both the convolution kernels in CNNs and image patches in ViTs.

In the quantum context, several approaches have been developed as a direct analogue to the classical CNN, namely the quantum convolutional neural network (QCNN) (Cong et al. 2019), which borrows the idea of localised operators, shared parameters, and downsampling from its classical counterpart. It has been benchmarked for binary classification with classical image data (Hur et al. 2022; Gong et al. 2024) and its effectiveness demonstrated through experiments. Variational circuits other than QCNNs can also be applied to image processing, but often require classical methods to reduce the dimension of the input data (Jaderberg et al. 2022; Khatun and Usman 2024). A popular choice is to use a pre-trained classical neural network, such as ResNet (He et al. 2015), to preprocess the original images and extract features (Zaman et al. 2024; Khatun and Usman 2024). This approach often involves a quantum-classical hybrid neural network, where the output layer of the classical neural network is replaced with a parameterised quantum circuit. However, the necessity of such an approach remains unclear, as the last layer of a classical neural network has a great similarity to logistic regression, which, by itself, is a simple machine learning model. When using a classical neural network for dimension reduction, the "heavy lifting" of feature extraction is off-loaded to the classical neural network, and the extracted features are often classified by a simple machine learning model. There is also research that involves the implementation or mimicking of classical convolutional operations via

quantum circuits, such as Kerenidis et al. (2019), in which the quantum version of convolution is achieved through local unitary operators with parameter sharing, and the pooling is achieved by tracing out a subset of qubits. Other research aims to replace only the convolution operation in a classical neural network, such as the quanvolutional neural network (Henderson et al. 2020) and its variants (Riaz et al. 2023). The data reuploading classifier (Pérez-Salinas et al. 2020) has also been adapted for image classification, such as Easom-Mccaldin et al. (2021), in which the pixel data in the image are encoded as rotation parameters together with trainable parameters that are shared among different patches of the same image. From this, we can see that, when applying quantum machine learning to high-dimensional data, especially image data, it is a common practice to

- use a classical model to reduce the dimension of the original image data and extract task-related features, such as in Khatun and Usman (2024) and Zaman et al. (2024). It should be noted that non-machine learning-based models could also be adopted to extract features and reduce the difficulty of learning for the downstream quantum model, such as the remapping method proposed in Zhou et al. (2023), which simplifies the multi-modal distribution of the input image.
- use amplitude embedding to reduce the number of qubits required, such as in West et al. (2023).
- use the data reuploading circuit or use a small circuit that only operates on localised patches of the image to reduce the number of qubits required when angle embedding is involved, such as in Easom-Mccaldin et al. (2021) and Riaz et al. (2023).

For the remainder of this section, we will give a brief introduction to common quantum data embedding methods.

1.2 Quantum data embedding

One of the most important steps when applying quantum machine learning to classical data is loading the data into the quantum computer. For example, it is difficult to find the classical counterpart of quantum data embedding for the loading of images, where classically images can easily be stored as rank-3 tensors and matrices in computer memory. Appropriate data embeddings are crucial to the success of both classical and quantum machine learning models. Similar data embedding processes, where the original data structure is not suitable to be directly processed by the machine learning model (mostly neural networks), could be found in research and applications that involve graph and natural language data. In each case, the data embedding method, as well as the machine learning model that follows data embedding, needs to reflect the intrinsic properties of the data. Such intrinsic properties can sometimes be described simply as translation, rotation, and permutation symmetry (Heredge et al. 2024). However, most of the time, it lies more on a semantic level and is hard to describe via mathematical relations.

There are three widely adopted data embedding methods for quantum machine learning (Schuld and Petruccione 2021):

- BASIS EMBEDDING: In basis embedding, a length-n binary string is directly embedded as one of the basis states of an n-qubit quantum system by applying Pauli-X operators to the quantum bits that are supposed to encode the classical bit "1". For example, to encode the binary bit string "0101" as a quantum state, one only needs to apply Pauli-X gates to the initial state $|0000\rangle$ on the second and fourth qubits:

$$0101_2 \mapsto |0101\rangle = X_2 X_4 |0000\rangle; \tag{1}$$

- ANGLE EMBEDDING: In angle embedding, classical floating-point data is embedded as rotation angles of parameterized quantum gates, such as the Pauli rotation gates R_X, R_Y, and R_Z. For $\mathbf{x} = (x_1, x_2)^T$, one could use angle embedding to encode $\mathbf{x}$ as follows:

$$\mathbf{x} \mapsto R_X(x_1) R_Z(x_2) |0\rangle; \tag{2}$$

- AMPLITUDE EMBEDDING: In amplitude embedding, the normalised padded data vector $\mathbf{x} = (x_0, x_1, \cdots, x_{2^N-1})^T$ is embedded as a quantum state of an N-qubit system with real amplitudes:

$$\mathbf{x} \mapsto |\mathbf{x}\rangle = \sum_{i=0}^{2^N-1} x_i |i\rangle. \tag{3}$$

There are also special embedding methods designed for image data, such as the flexible representation of quantum images (FRQI) (Le et al. 2011; Yan et al. 2016), which embeds the data in a quantum state that takes spatial information into account. Since these methods still require specific amplitudes for the embedded quantum states, they could be viewed as an extension of the amplitude embedding method. To make amplitude embedding feasible with current quantum hardware, several approximate heuristic-based state preparation methods have been proposed, such as the GASP algorithm (Creevey et al. 2023), in which the authors applied genetic algorithms to discover relatively low-depth quantum circuits for approximate state preparation, as well as the variational circuit-based approach to approximately prepare the FRQI states shown in Shen et al. (2024).

We can see that, besides angle embedding, most of the other approaches emphasise on encoding classical data as

quantum states, which cannot work with the data reuploading circuit adopted in this paper. It is hard to scale up to larger-dimension datasets for angle embedding with the available number of qubits on current quantum devices. Here, we combine the quantum Hamiltonian embedding method, described in Section 2.1, and the data reuploading approach, described in Section 2.2, for image classification tasks. Compared to angle embedding, embedding the image as a Hamiltonian puts the pixels in the image on a more equal footing, meaning that they go through the same type of mathematical operation. Also, the matrix representation of a quantum Hamiltonian for a qubit system is naturally "two-dimensional," in the sense that it has the same shape as a (grey-scale) image, making it more suitable for modelling images.

2 Data reuploading classifier with quantum Hamiltonian embedding

In this section, we will discuss the quantum neural network classifier model based on the discussions from the previous section. We opt for Hamiltonian image embedding (Section 2.1) for data encoding and the data reuploading classifier (Pérez-Salinas et al. 2020) (Section 2.2), for both their simplicity in terms of implementation with linear algebra libraries such as JAX (Bradbury et al. 2018), and the intuitive connection with classical neural networks for the data reuploading circuit. While looking for quantum advantage in terms of training and inference speedups is a legitimate aim, it is not the primary goal here. Instead, we seek to integrate heuristics from deep learning into quantum machine learning model design.

In the context of quantum computing, a quantum Hamiltonian can be written as a square matrix. This representation provides the opportunity to encode image data in a two-dimensional way, rather than flattening the image and/or using the pixel values as rotation angles of parameterised gates, which could introduce unwanted bias on the decision boundary. Also, the possibility of encoding an entire image as a quantum Hamiltonian with (polynomial) less qubits required than amplitude embedding and angle embedding gives us the chance to reduce classical preprocessing to a minimum. We will see in Eq. 6 that encoding an image as a quantum Hamiltonian provides a richer nonlinearity compared to that provided by quantum measurements, which could further enhance the expressivity of our model.

2.1 Hamiltonian image embedding

As mentioned in the previous section, one of the most important components of a quantum machine learning model is how to embed classical data into the quantum computer. Since we are working with image data, there is a preference for data embedding methods that preserve two-dimensional structures of images and transform image pixels with the same nonlinearity function (activation functions) to avoid unwanted bias on the decision boundaries.

In this paper, we adopt the Hamiltonian embedding method (Schuld and Petruccione 2021; Yang et al. 2023) for image data encoding. First, we "Hermitianise" our square, grey-scale, real-valued image matrix M by the following:

$$H_M = \frac{M + M^T}{2}. \tag{4}$$

This is the only classical preprocessing required for our model, in addition to padding the image with zeros for the MNIST and FashionMNIST datasets. Here, the embedding unitary for the input image data is simply the matrix exponentiation of H_M:

$$W(t; M) = e^{\frac{-iH_M t}{2}}, \tag{5}$$

where t is a trainable parameter instead of the physical time. If we expand $W(t; M)$ in a Taylor series, we have the following:

$$W(t; M) = 1 - \frac{iH_M t}{2^1} - \frac{H_M^2 t^2}{2! \times 2^2} + \frac{iH_M^3 t^3}{3! \times 2^3} + \cdots \tag{6}$$

We can see that by simply time-evolving the (Hermitianised) image, a (matrix) polynomial function is applied on the whole image level, bringing "cheaper" nonlinearity compared to angle embedding with single-parameter rotation gates. Later in Section 3, we demonstrate that with the quantum Hamiltonian embedding approach, our model could outperform QCNN for various datasets. In our model, the Hamiltonian embedding of image M, parameterised by a single parameter t, will act as the data encoding unitary for our quantum machine learning model, which will be discussed in the following subsections.

2.2 Data reuploading

The data reuploading variational quantum circuit, first proposed in Pérez-Salinas et al. (2020), is derived from the guiding principles of classical neural networks that the data are reused multiple times in classical deep neural networks. Originally, it was designed for classification tasks and was later extended to applications in reinforcement learning (Coelho et al. 2024) to replace the classical Q network. Researchers have also demonstrated the effectiveness of the data reuploading circuit on small-scale datasets when trained on a superconducting quantum processor (Tolstobrov et al. 2024).

Variational circuits representing quantum versions of neural networks can be written as follows (before measurement):

$$|\psi(\mathbf{x};\boldsymbol{\theta})\rangle = V(\boldsymbol{\theta})U_{\boldsymbol{\phi}}(\mathbf{x})|0\rangle^{\otimes n}, \tag{7}$$

where $V(\boldsymbol{\theta})$ are the variational layers parameterised by $\boldsymbol{\theta}$ and could be absorbed in the measurement observables O, becoming $\boldsymbol{O}(\boldsymbol{\theta}) = V^{\dagger}(\boldsymbol{\theta})OV(\boldsymbol{\theta})$:

$$\begin{aligned}\langle\psi(\mathbf{x};\boldsymbol{\theta})|O|\psi(\mathbf{x};\boldsymbol{\theta})\rangle &= \langle 0|^{\otimes n}U^{\dagger}_{\boldsymbol{\phi}}(\mathbf{x})V^{\dagger}(\boldsymbol{\theta})OV(\boldsymbol{\theta})U_{\boldsymbol{\phi}}(\mathbf{x})|0\rangle^{\otimes n}\\ &= \langle 0|^{\otimes n}U^{\dagger}_{\boldsymbol{\phi}}(\mathbf{x})\boldsymbol{O}(\boldsymbol{\theta})U_{\boldsymbol{\phi}}(\mathbf{x})|0\rangle^{\otimes n},\end{aligned} \tag{8}$$

$\mathbf{x}$ is the input data, and $U_{\boldsymbol{\phi}}(\mathbf{x})$ is the data encoding unitary and could be parameterised with some other set of parameters $\boldsymbol{\phi}$. In this form, the input data only appears once in the model, while in classical neural networks, one input neuron can be accessed by more than one neuron in the hidden layer. Motivated by this difference, the data reuploading circuit can be written as follows:

$$|\Psi(\boldsymbol{x};\vec{\boldsymbol{\omega}})\rangle = \prod_{i=1}^{L}\left[V(\boldsymbol{\omega}_i)U_{\boldsymbol{\phi}}(\boldsymbol{x})\right]|0\rangle^{\otimes n}. \tag{9}$$

In this definition, the data encoding unitary $U_{\boldsymbol{\phi}}(\mathbf{x})$, together with the parameterised layer V, are repeated L times with the same data encoding unitary but with different parameters for $V, \vec{\boldsymbol{\omega}} = \{\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \cdots, \boldsymbol{\omega}_L\}$. Also, it has been proved that data reuploading circuits in principle exhibit a quantum advantage in terms of function approximation (Yu et al. 2023).

2.3 The model

Combining the data reuploading circuit and Hamiltonian embedding, we have the following quantum machine learning model (prior to measurement, also shown in Fig. 1):

$$|\varphi(\boldsymbol{t},\vec{\boldsymbol{\omega}};M)\rangle = \prod_{i=1}^{L}[V(\boldsymbol{\omega}_i)W(t_i;M)]\,|+\rangle^{\otimes n}. \tag{10}$$

Here, we set the circuit to begin with an equal superposition of all basis states ($|+\rangle^{\otimes n}$).

There is flexibility in the structure of the parameterised layer V. For small datasets, we opt for a parameterised layer composed of SU(4) unitary gates in a brick wall layout with different parameters. Generally, a SU(N) gate, where $N = 2^n$, n being the number of qubits the gate acts on, can be written as follows:

$$\mathrm{SU}(N)(\boldsymbol{\theta}) = \exp\Big(\sum_{i=1}^{m} \mathrm{i}\theta_i G_i\Big), \tag{11}$$

where $m = 4^n - 1$ and $G_i \in \{I, X, Y, Z\}^{\otimes n}\backslash\{I^{\otimes n}\}$. $\boldsymbol{\theta} = \{\theta_1, \cdots, \theta_{4^n-1}\}$.

In our model, the classical data pass through a nonlinear operation first (time evolution), then followed by a

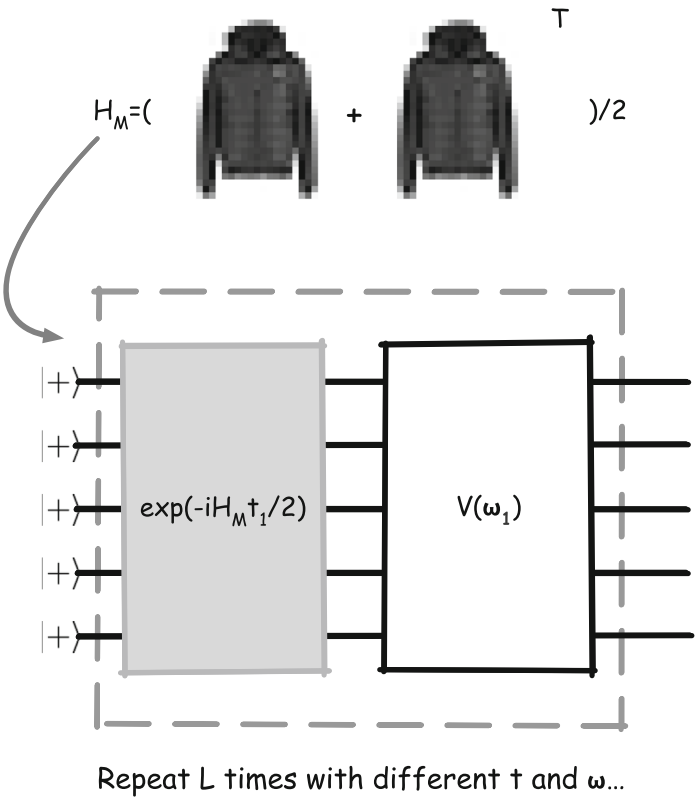


Fig. 1 The quantum machine learning model described in Eq. 10. The model shown in the figure has a five-qubit circuit for the images from the FashionMNIST dataset (and MNIST as well). The grey-scale images are padded from 28×28 to 32×32 with zeros. Then, the quantum Hamiltonian H_M is constructed with the padded image matrices. The data encoding unitary (grey box in the circuit diagram) and the parameterised circuit unitary (white box in the circuit diagram) are repeated L times for an L-layered data reuploading circuit

parameterised unitary layer. Although this is not common in deep learning practice, where nonlinear operations (activation functions) normally occur after linear and convolutional layers, it could be viewed as a form of pre-activation, which enabled the training of a 1001-layer ResNet in He et al. (2016).

For a three-qubit circuit, there could be two different configurations, as shown in Fig. 2a. These two different configurations generally do not have a noticeable impact on the performance of the model. The same holds for the SU(4) gates in a five-qubit circuit, as shown in Fig. 2b. For larger datasets, the SU(N) gate on all the qubits in the circuit will be adopted.

For the classification of $K-$ classes, the probability for each label $i \in \{0, 1, \cdots, K-1\}$ can be obtained by measuring the $\lceil \log_2 K \rceil-$ qubit projection operator $P_i = |i\rangle\langle i|$:

$$p(M; i) = \langle \varphi(\boldsymbol{t}, \vec{\boldsymbol{\omega}}; M) | (P_i \otimes \boldsymbol{I}_{n-\lceil \log_2 K \rceil}) | \varphi(\boldsymbol{t}, \vec{\boldsymbol{\omega}}; M) \rangle, \tag{12}$$

where $\boldsymbol{I}_{n-\lceil \log_2 K \rceil}$ is the $(n - \lceil \log_2 K \rceil)-$qubit identity operator, and n is the total number of qubits in the circuit. The loss function for training is the cross-entropy cost function:

$$\text{Cross-Entropy Loss } (M) = -\sum_{i=0}^{K-1} y_i \log_2 p(M; i), \tag{13}$$

where y_i is the true probability of class i for the input M, which, in the case of one-hot encoding, is 1 for the true class and 0 for all others. For a more detailed explanation of the cross-entropy function, readers could refer to deep learning-related textbooks, such as Chapter 5.7 in Prince (2023).

To avoid taking the log of 0, we use $\text{Softmax}(p(M; i))$ to replace $p(M; i)$:

$$\text{Softmax}(p(M; i)) = \frac{e^{p(M;i)}}{\sum_{k=0}^{K-1} e^{p(M;k)}}. \tag{14}$$

The purpose of the Softmax function is to convert a real-valued vector to another real-valued vector, but with values in (0, 1) and sum to one (Prince 2023).

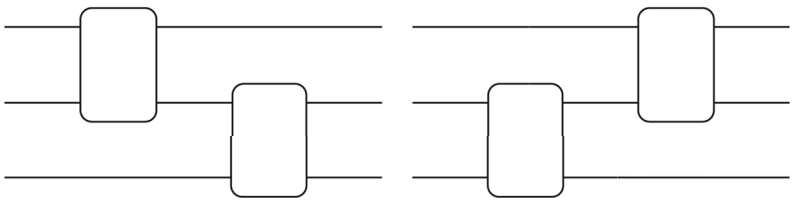
3 Simulation experiments and results

3.1 Baseline model and datasets

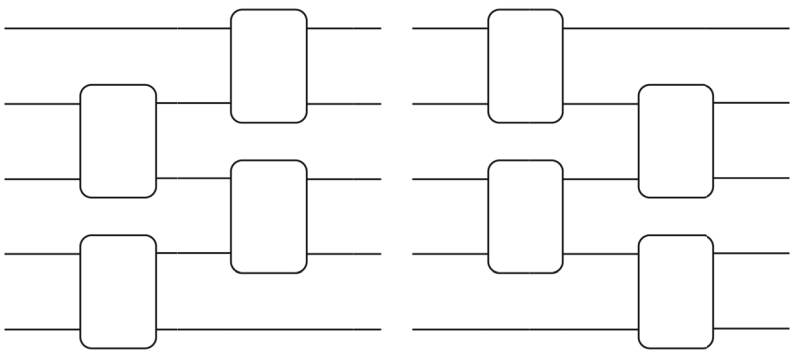
Our baseline model for comparison is the quantum convolutional neural network proposed in Cong et al. (2019). The structure and implementation of the baseline model follow Kottmann et al. (2022). For the baseline model, since amplitude embedding is used, all input images were flattened into vectors and padded. The size and number of parameters of the model depend on the size of the input and the number of classes.

We trained and tested our model on four different datasets:

The Kaggle CT Medical Image dataset This is a small subset of images from Albertina et al. (2016), obtained from the Kaggle website (Scott Mader 2017). This dataset contains 100 CT medical images that have binary labels "True" or "False" for "Contrast". The original dimension of the images



(a) Two possible layouts for two SU(4) gates on a three-qubit circuit.



(b) Two possible layouts for four SU(4) gates on a five-qubit circuit.

Fig. 2 Potential layouts for the SU(4) gate in three- and five-qubit circuits used in the quantum machine learning model in this paper

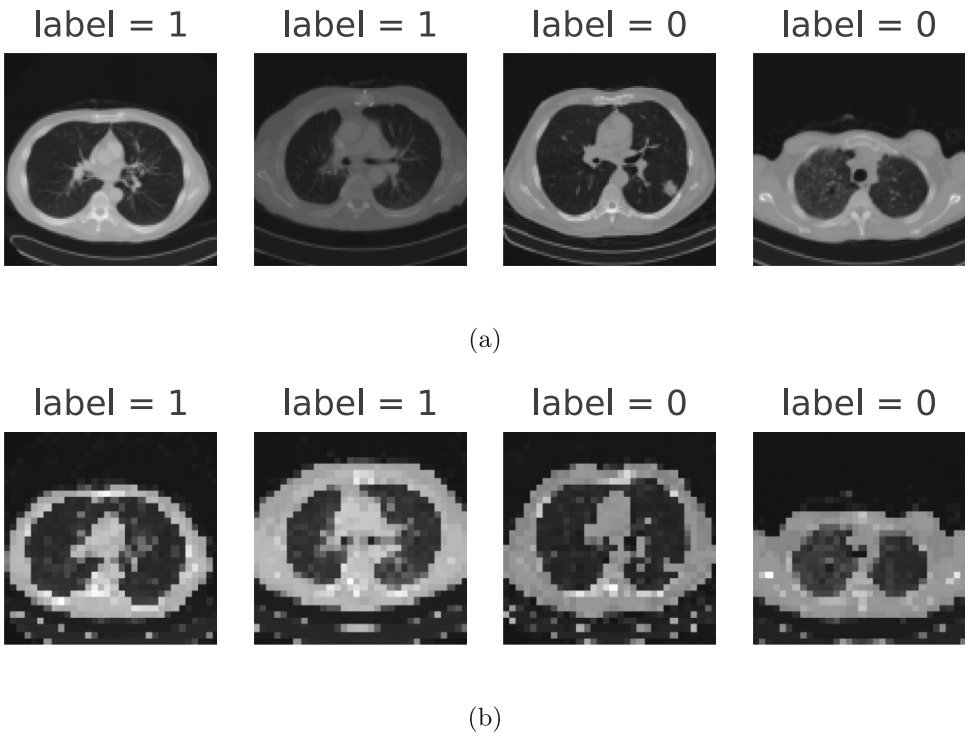


Fig. 3 Sample images from the Kaggle CT Medical Image dataset. **a** Original image samples from the CT medical image dataset. **b** Same images as **a**, but resized to 32 by 32. The resize is achieved via OpenCV-Python's resize function. The pixels are also normalised before being fed into the quantum neural networks using OpenCV-Python's normalise function

is 512 by 512. To reduce the simulation cost, the images were resized to 32 by 32 using the Python package of OpenCV (Itseez 2015). The resized images were randomly divided into train (80 images) and test (20 images) datasets. Sample images of the dataset are shown in Fig. 3.

Subset of the Sklearn digits dataset This data (Alpaydin and Kaynak 1998) is obtained through the machine learning package "Scikit-learn" (Pedregosa et al. 2011). The dimensions of the images are 8 by 8. Only images with labels 0 to 7 (eight classes in total) were sampled when splitting train (1200 images) and test (100 images) datasets. Sample images are shown in Fig. 4.

Subset of the MNIST dataset The data (LeCun et al. 2010) is obtained through the corresponding data loading module in Torchvision (maintainers and contributors 2016). Only images with labels 0 to 7 (eight classes in total) were sampled when constructing the train datasets (48,200 images) and the test datasets (8017 images). The original dimension of the images in the MNIST dataset is 28 by 28. The images were padded to 32 by 32 with zeros. Image samples are shown in Fig. 5.

Subset of the FashionMNIST dataset Data (Xiao et al. 2017) is obtained through the corresponding data loading module in Torchvision (maintainers and contributors 2016). Only images with labels 0 to 7 (eight classes in total) were sampled when constructing the train datasets (48,000 images) and the test datasets (8000 images). The original dimension of the images in the FashionMNIST dataset is 28 by 28. The images were padded to 32 by 32 with zeros. Image samples are shown in Fig. 6.

3.2 Results

Although our model does not provide speedups for either training or inference when compared to classical techniques, it still outperforms the baseline QCNN model with different

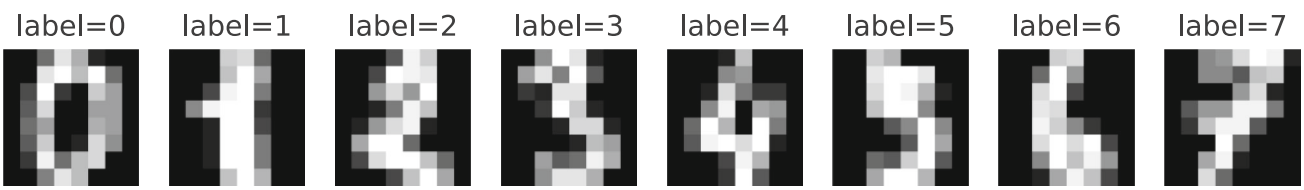


Fig. 4 Sample images from the Sklearn digits dataset. The size of the images is eight by eight

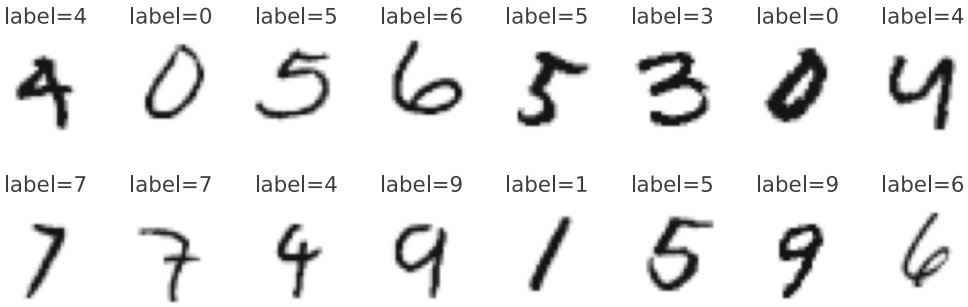


Fig. 5 Sample images from the MNIST dataset. The size of the original images is 28 by 28. Images were padded with zeros to 32 by 32 before constructing the Hermitian operators of the images

random initialisations and without extensive hyperparameter search. These results show how machine learning models based on straightforward heuristics could easily achieve good performance, even outperforming famous quantum machine learning models, and reach a level close to classical machine learning models.

Both the baseline and the proposed model were trained with 20 different parameter initialisations for the Kaggle CT Medical Images dataset and subset of the Sklearn Digits dataset, each for 500 iterations with the Adam optimiser (Kingma and Ba 2017). In contrast, the MNIST and FashionMNIST subset datasets were trained for five different parameter initialisations, each for 100 iterations with the Adam optimiser (Kingma and Ba 2017). For the CT Medical Images and Sklearn digits datasets, the loss and accuracy are averaged over the 20 different parameter initialisations. For the MNIST and FashionMNIST datasets, the loss and accuracy are averaged over the five different parameter initialisations. The averaged values of the evaluation metrics (loss and accuracy) of the final iteration for each of these four datasets can be found in Table 1, and the curve plot of the metrics through the training iterations could be found in Figs. 8, 9, 10, and 11 in the Appendix. A summary of results can be found in Table 1.

3.3 Analysis

The performance gaps for the three datasets Sklearn Digits, MNIST, and FashionMNIST between the proposed models and the baseline models can clearly be seen in Figs. 9, 10, and 11. The performances of the proposed model on the test dataset are consistently better than those of the baseline model on the training dataset. However, the performance separation on the Kaggle CT Medical Images dataset is not as significant. A major difference between the Kaggle CT Medical Images dataset and the other three datasets is that it requires downsampling (from 512×512 to 32×32) before data embedding. From Fig. 3 a and b, we can see that the downsampling process obscured many fine-grained features in the image. This downsampling process, in addition to the limited number of samples in the dataset, could be one of the reasons that led to this smaller performance difference between the proposed model and the baseline model.

Comparing the performance in the Sklearn Digits dataset and the MNIST dataset, we can see that although both the baseline model and the proposed model experienced a performance drop when the size of the input data increased, the performance of the baseline model dropped more and widened the gap between the baseline model and the proposed model. This phenomenon shows that the baseline

Fig. 6 Sample images from the FashionMNIST dataset. The size of the original images is 28 by 28. Images were padded with zeros to 32 by 32 before constructing the Hermitian operators of the images

Table 1 The performance of the baseline and proposed models at the last iteration on all the datasets

Dataset	Metrics	QCNN	HamEmb	HamEmb (2× depth)
Medical Images	Train loss	0.6571	**0.5457**	–
	Test loss	0.6843	**0.6318**	–
	Train accuracy	73.25%	**83.75%**	–
	Test accuracy	58.25%	**67.75%**	–
Digits	Train loss	1.9461	**1.5454**	–
	Test loss	1.9493	**1.5609**	–
	Train accuracy	79.77%	**95.22%**	–
	Test accuracy	78.95%	**94.40%**	–
MNIST	Train loss	1.9889	**1.5742**	–
	Test loss	1.9871	**1.5740**	–
	Train accuracy	46.94%	**89.24%**	–
	Test accuracy	47.03%	**89.72%**	–
FashionMNIST	Train loss	1.9738	1.6072	**1.5810**
	Test loss	1.9741	1.6114	**1.5876**
	Train accuracy	43.46%	78.52%	**80.46%**
	Test accuracy	42.90%	77.77%	**79.61%**

For the Kaggle CT Medical Images dataset, the model performance data is averaged over 20 different parameter initialisations; for the Digits dataset, the performance is also averaged over 20 different parameter initialisations. For the MNIST and FashionMNIST datasets, the performances are both averaged over five different parameter initialisations. For the FashionMNIST dataset, we also have additional results, in which we doubled the depth of the data reuploading circuit. However, we can see that the performance increase is only marginal compared to the increase in the number of parameters

model, which adopts amplitude embedding as the data encoding method, cannot efficiently capture the features in such a high-dimensional image dataset. The performance drop of the proposed model, although smaller compared to that of the baseline model, still indicates that it could also have trouble dealing with high-dimensional image data, but is saved by the increased number of parameters.

When comparing the performance of our scheme on MNIST and FashionMNIST, the baseline model has a smaller performance drop compared to the proposed model, as shown in Figs. 10 and 11, as well as Table 1, albeit still performing worse than the proposed model. FashionMNIST has richer and more complex spatial features compared to the MNIST dataset. The small change in performance of the baseline models indicates that it is likely to fail in effectively capturing the features in both datasets. The performance drop for our proposed model implies that we could deduce that the features in the FashionMNIST dataset are also more challenging compared to those in the MNIST dataset. By encoding the entire image as a quantum Hamiltonian, our model could potentially have characteristics similar to a convolution layer with a very large kernel. Large convolutional kernels cannot capture fine-grain details in the image as well as smaller-sized convolutional kernels, and the lack of local features could be the reason the performance drops for both the baseline model and the proposed model.

4 Discussion

In the previous sections, we designed a quantum neural network model based on the data reuploading circuit (Pérez-Salinas et al. 2020), using the quantum Hamiltonian embedding (Schuld and Petruccione 2021) approach as the data encoding unitary, demonstrating that our model could achieve reasonably better performance than the well-known QCNN model without extensive architecture and hyperparameter search on multiple datasets, or dedicated pre-trained variational circuits to approximate quantum-embedded classical images (Shen et al. 2024).

It should also be noted that our numerical experiments use larger datasets compared to previous quantum machine learning research, since data scaling is also an important research question in different areas of deep learning, such as large language models (Kaplan et al. 2020; Hoffmann et al. 2022). It is common for machine learning models that have good performance on a small subset of common datasets, such as MNIST and FashionMNIST, to perform badly on a larger scale, both in terms of number of labels and number of data in each label.

In this section, we point out similarities between our model and classic neural network designs. We followed heuristic techniques when choosing the data embedding methods and the structure of the QNN model. In this section, we will dive

into the details of these heuristics and propose six essential principles for quantum neural network design to inspire future research in this direction.

4.1 Resemblances to classical neural network design

Possible connection to pre-activation in classical neural networks In He et al. (2016), a modified version of the original ResNet (He et al. 2015), the activation function (ReLU), was placed before the convolutional layers (and after the batch normalisation layer). This modification (He et al. 2016) has been shown to increase the trainability of a 1000-layer ResNet and reduce overfitting. In the proposed model presented here, we can see from the expansion of the Hamiltonian embedding in Eq. 6 that our training data first go through a non-linear transform, then a parameterised layer. This implicit transform of the input data could be the reason that during training, the proposed model did not suffer from any obvious barren plateau with different random initialisations. However, at this stage, this is just a conjecture and still requires further investigation.

Possible connection to the Gated Linear Unit (Dauphin et al. 2017) Generally, the Gated Linear Unit (GLU) follows the form

$$\mathrm{GLU}(x) = f(x) \cdot \sigma(g(x)), \tag{15}$$

where both f and g are linear transformations (such as the linear layer in an MLP or the convolution layer in a CNN), and σ is usually a non-linear activation function, such as ReLU or the tanh function. Recall the mathematical form of our model from Eq. 10:

$$|\varphi(\boldsymbol{t}, \vec{\boldsymbol{\omega}}; M)\rangle = \prod_{i=1}^{L} [V(\boldsymbol{\omega}_i) W(t_i; M)] \, |+\rangle^{\otimes n}, \tag{16}$$

which can be rewritten as follows:

$$v(\mathbf{x}; \boldsymbol{t}, \boldsymbol{V}) = \prod_{i=1}^{N} [V_i \cdot \sigma_{t_i}(\mathbf{x})] v_0, \tag{17}$$

where $V_i \cdot \sigma_{t_i}(\mathbf{x})$ can be viewed as a gated linear unit parameterised by weight matrix V_i and parameter t_i. Although it is hard to say that there is a one-to-one correspondence between GLU and the proposed model, the implicit data transform in the quantum Hamiltonian embedding unitary could contribute to the superior performance of our model compared to other QCNN schemes.

We observed a performance degradation in the proposed model occurring when the complexity of the data increases (Sklearn digits → MNIST → FashionMNIST). An intuitive explanation is that the images in the FashionMNIST dataset contain more details than those in the MNIST dataset. Most pairs of MNIST digits have been shown to be well distinguished by using just a single pixel (Xiao et al. 2017). Our result for the FashionMNIST brings us close to the performance achieved by classical machine learning algorithms on the benchmark provided in Xiao et al. (2017). Since we did not extensively search for better structures, there is ample room for improvement when it comes to model architecture. For example, our model encodes the entire image as a whole, whereas modern deep learning practice shows that even without convolution, we should still divide the image into patches, allowing the model to focus more on local spatial features, such as the embedding layer in the vision transformer (Dosovitskiy et al. 2020).

4.2 Guiding principles of quantum machine learning model design

In this paper, we assume that, as with their classical counterparts, quantum machine learning with quantum neural network (QNN) models also needs to process input data and to discover hidden patterns, which means that they may also benefit from incorporating similar intuition to that used for classical deep neural networks, while operating under the framework of quantum computing. In the previous sections, we briefly mentioned some of the reasons for our design choices of the QNN model. In this section, we lay out the following, but non-exhaustive, guiding principles for the design of QML models (shown schematically in Fig. 7).

1. Less initial focus on speedups: During the conceptualisation of our model, we did not put speedup quantum advantage as the initial goal (Fig. 7a). Instead, we searched for options that align with the structure of the data (which will be further discussed later), as well as using heuristics from classical neural networks (the data reuploading circuit). Since we have demonstrated with numerical experiments that our model has better performance relative to QCNN results (Table 1), in the future, research could be focused on how to efficiently implement such quantum Hamiltonian embedding on current quantum hardware. A similar evolution has occurred in the research of deep learning models. For example, after the transformer model proved its superior performance in language processing tasks (Vaswani et al. 2017), acceleration methods such as flash attention (Dao et al. 2022) were proposed to speed up the calculation of the attention layers. When developing quantum machine learning models, one could follow a similar principle: first, find a framework that has acceptable performance on common datasets, and then move on to the optimisation of the model.

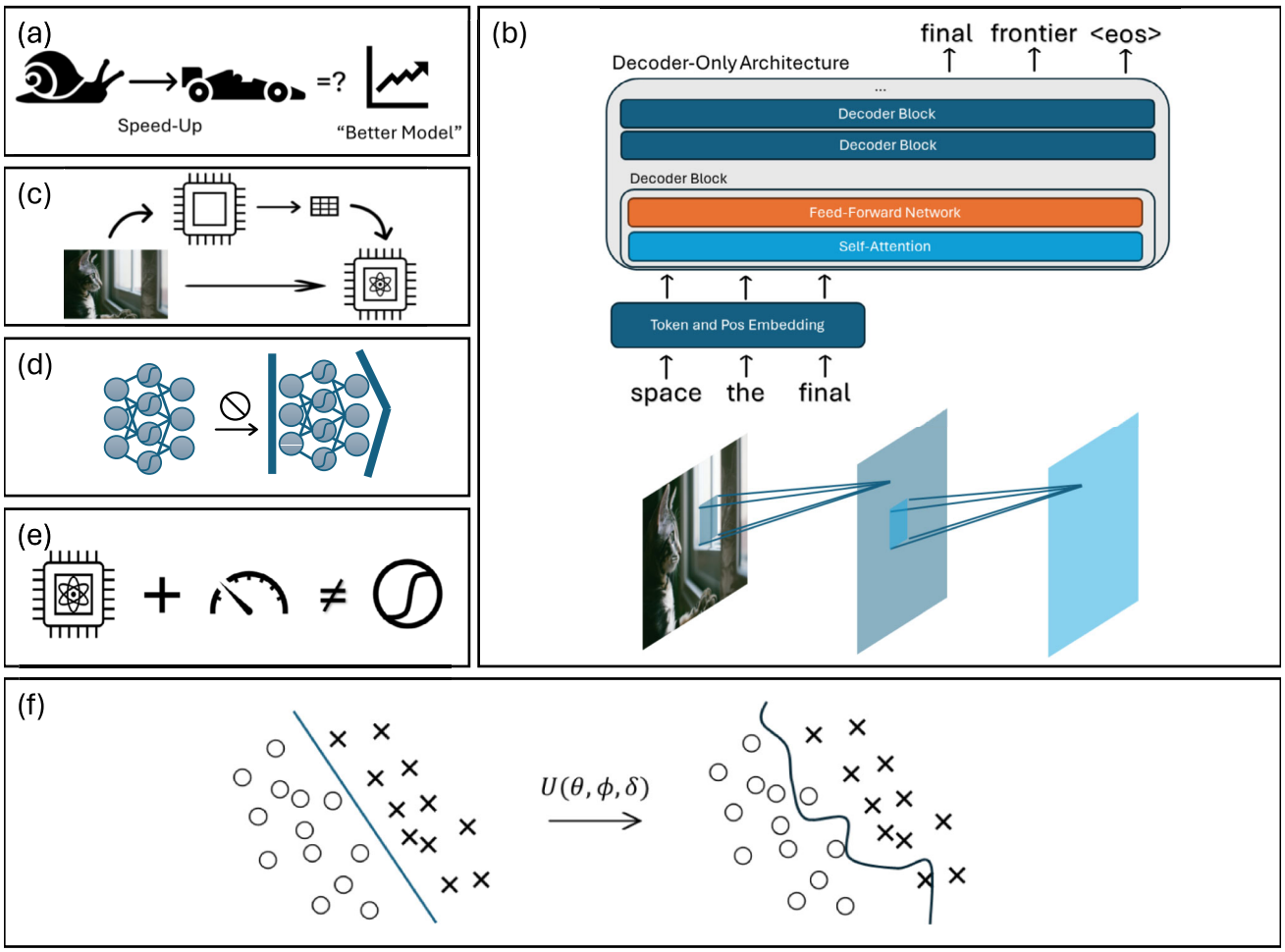


Fig. 7 Our proposed six guiding principles required for designing quantum machine learning models. **a** Speedup is not the first thing to consider when designing new quantum machine learning models, since it does not necessarily lead to "better models", i.e. we should instead focus on improvements in metrics of performance, such as accuracy. **b** The intrinsic structure of the data should be taken into account when designing the model architecture. Images have two major spatial directions, so the convolution kernel in a CNN (lower part of **b**) will scan in both directions, while text data only has a single temporal dimension, so the model needs to generate the words one by one (upper part of **b**, which is a decoder-only generative transformer Radford et al. 2018). **c** Using a classical model, such as a neural network backbone or PCA to reduce the dimension of the data (the upper path), obscures the real effectiveness of the quantum machine learning model, so we should minimise classical preprocessing as much as possible. **d** Avoid direct "quantisation", i.e. avoid using quantum circuits to implement the exact mathematical operations of a classical machine learning model. **e** Measurements in the quantum circuit of the quantum machine learning model may not have the same kind of nonlinearity as the activation functions in classical neural networks, such as the ReLU (rectified linear unit, $\mathrm{ReLU}(x) = \max(0, x)$) function. **f** Some data embedding methods, such as angle embedding, may introduce unwanted bias toward certain kinds of decision boundaries, harming the performance of the machine learning model

2. KEEP THE DATA IN MIND: One of the major reasons that the quantum Hamiltonian embedding approach, instead of many other popular data embedding methods, was selected during the model design process is that by embedding the images as quantum Hamiltonian in the form of matrices, we could preserve the two-dimensional structure of the images as much as possible. This particular choice is often referred to as the inductive bias in machine learning literature. In classical machine learning, inductive bias is a set of assumptions a model makes to generalise better on unseen inputs, since there are no completely general learning algorithms according to the No Free Lunch theorem (Goyal and Bengio 2022). This inductive bias is also integrated into the design of classical convolutional neural networks, where the convolution kernel usually moves along the height and width of the image to capture the spatial dependence of pixels on different locations (see Fig. 7b). Even when the convolutional neural network is applied to sequence data (Kim 2014), the kernel usually moves along the time dimension to capture temporal correlations, which is different from the CNNs that operate on image data. In more recent

research, transformer-based architectures (Vaswani et al. 2017; Radford et al. 2018) have often been used for language modelling. We can see from these two examples that even though the underlying operation (convolution) is the same, different kinds of data will require different kinds of convolution kernels.

3. MINIMISE CLASSICAL PREPROCESSING: In our model, the only classical preprocessing steps involved are the normalisation of pixel values, which is also a common practice in classical deep learning, as well as the padding, transpose of and addition between the image matrices to construct the quantum Hamiltonian (see Fig. 1c). It has been a common practice in some of the quantum machine learning research to adopt the backbone of a classical neural network, such as the ResNet-18 (He et al. 2015), to extract task-specific features from the high-dimensional image data (for an example, see Zaman et al. 2024). In essence, this kind of approach replaces the last fully connected layer of the backbone classical neural network with a parameterised quantum circuit. Usually, for classification tasks, the activation function at the last layer (the fully connected layer) is the softmax function, making the last layer a multinomial logistic regression. Figuratively speaking, we can say that the classical backbone neural network has done most of the "heavy lifting" of the downstream task by extracting features from the image. Even a trivial (multinomial) logistic regression classifier could complete the task, which makes one question the necessity of introducing quantum models at the end of the classical model. In the design of our model, we avoided this potential issue by minimising the classical preprocessing as much as we could.
4. AVOID DIRECT "QUANTISATION" OF CLASSICAL MODELS: Instead of directly "quantising" the classical machine learning model to a quantum one which performs the same arithmetic operations on a quantum computer via quantum linear algebra, it would be preferable to develop quantum neural network models that could utilise operations that are intrinsic to the underlying quantum system, such as the time evolution of a quantum Hamiltonian, indicated by the numerical experiment results that the quantum Hamiltonian embedding, which is related to the time evolution of a quantum system, outperforms the QCNN, which "quantise" the classical convolution and pooling operations via qubit-local unitary operators and measurement-outcome-controlled unitary operators. We could see similar trends happening in classical deep learning research: models that harvest most of the hardware prevail. One of the reasons that transformers (Vaswani et al. 2017) have such an advantage over recurrent-neural-network-based models such as the long-short term memory (LSTM) network and the Gated Recurrent Unit (GRU) network is that the structure of transformer models enables it to be easily parallelised during training and accelerated by GPUs. In other words, the structure of transformer models is more compatible with current GPU architectures.
5. NONLINEARITY IS NOT WHAT YOU THINK: During the design of our model, we relied on the evolution of the image Hamiltonian to introduce "nonlinearity" into the (quantum) neural network, which resembles the gated linear unit, as shown in Section 4.1. However, in quantum computing, measurement operations are often considered a "nonlinear" operation since the time evolution of the system is no longer defined by the Schrödinger equation, which is a reversible linear differential equation. While in deep learning research, nonlinearity refers to the ability of the model or a layer to nonlinearly (that is, not just translation, scaling, or rotation) transform the data manifold (Olah 2014). Recent research shows that nonlinear functions (activation functions) play a more important role rather than merely providing nonlinear transformations on the data manifold. In Humayun et al. (2024), the authors suggest that nonlinearity (ReLU) in deep neural networks divides the input space into non-overlapping linear regions. In Teney et al. (2024), the authors suggest that activation functions introduce a non-trivial bias to the neural network, making the neural network favour functions with certain levels of complexity. For example, ReLU-activated neural networks would favour low-complexity (low-frequency) functions which often align with the training target. Features learned by neural networks can also be regarded as directions in the activation space (Elhage et al. 2022). Since operations conditioned on mid-circuit measurement results could be converted to quantum-controlled operations via deferred measurements, the transformations on the input states by the quantum convolutional neural network could be represented as a complete-positive trace preserving (CPTP) map, which is linear (but not reversible). Although quantum neurons with repeat-until-success circuits could produce nonlinear responses akin to a classical activation function (Cao et al. 2017), the input is first passed to an R_Y gate, which is already nonlinear. The repeat-until-success model could be viewed as a filtering process to produce the tanh-like signal. In addition, the repeat-until-success process makes it difficult to scale to a large number of neurons, which is common for today's large models.
6. BE CAREFUL OF UNWANTED BIAS: The other reason that we opted for quantum Hamiltonian embedding rather than popular choices, such as angle embedding, was to avoid unwanted bias. In recent research, the authors of Bowles et al. (2024) showed that angle embedding for quantum machine learning models could introduce bias on decision boundaries that are formed based on periodic

functions, which is also taken into account during the design of our model (Fig. 7f), since the rotation angles passed as gate parameters will first go through trigonometric functions. This is essentially the same as using the sine or cosine functions as activation functions, which is very rare in deep learning practice. If parameterised gates, which take multiple rotation angles as input parameters, such as the U gate $U(\theta, \phi, \delta)$, different elements (pixels) of the same datum (image) will go through different non-linear operations. Unless we are sure that this is required due to the nature of the data or the task, we should avoid such different treatments to the same datum.

In summary, our proposed principles for designing quantum machine learning models shed new light on how to combine quantum computing and machine learning for classical data. Some of these principles could be viewed as inductive biases, which enable the algorithm to prioritise certain hypotheses over others. Others are motivated by the underlying hardware that runs the model. However, it should be noted that these six principles are far from complete. The hardware-based principles could be extended to analogue neural networks on the quantum processors (or, in Hinton's words, "mortal computation" Kleiner 2024 on a quantum processor). The principles for inductive biases could be in a difficult position in the future due to recent research in large language models, especially those on the scaling laws (Kaplan et al. 2020; Hoffmann et al. 2022). According to Sutton (2019), one could say that the past 70 years of AI research could be summarised into the development of more and more general methods with weaker modelling assumptions or inductive biases, adding more data and compute power, or in other words, to scale up. There has been evidence that given enough data and compute budget, even MLPs (multi-layer perceptron) and the closely related MLP-Mixer models could perform in-context learning, which is the ability to solve a task from only input examples (Tong and Pehlevan 2024). Also, in addition in Nguyen et al. (2024), through numerical experiments with a pixel transformer that treats an image as a set of pixels and employs randomly initialised and learnable position embeddings without any information about 2D structure, the authors questioned the necessity of the inductive bias of locality which presents in many computer vision models, from LeNet (LeCun et al. 1989) to the vision transformer (Dosovitskiy et al. 2021). It remains to be seen if such considerations impact the introduction of inductive biases from the quantum side to QML models (Bowles et al. 2023). With limited computation, we still need inductive biases during the design of QML models, which should be comforting for researchers in this area. However, in the future, when (classical and/or quantum) computing is cheaper and more accessible for machine learning and AI research, enabling the ability to train even larger models with a larger amount of data, it would be necessary to remove such inductive biases from the design of the model (Chung 2024).

It also should be noted that whether to continue investment of resources for the scaling law should be the future of AI research is still under heavy debate. There are still many issues and limitations existing in the current auto-regressive decoder-only transformer large language models (Barbero et al. 2024; Abbe et al. 2024; Nezhurina et al. 2024; Ofir Press et al. 2022; Verma et al. 2024; Wu e tal. 2024; Kambhampati et al. 2024; Zhou et al. 2023), and not all of them could be solved by scaling up the size of the data and computation resources invested in the training process, especially those regarding compositional reasoning (Dziri et al. 2023; Wang et al. 2024). In the context of current work in quantum computing, there is still a significant gap between the major concerns in today's AI research and the research on quantum advantage and utility for AI. The research presented here seeks to find ways to bridge this gap.

To summarise the main points and contributions of the paper, we have the following:

- We have developed an embedding approach for image data that requires as little preprocessing as possible and preserves the spatial features as much as possible. Our embedding approach could incorporate large-dimension inputs but still work with the data re-uploading circuit.
- We have developed a QNN model based on the data reuploading circuit that can achieve decent performance on image classification tasks.
- Furthermore, based on our model design process and experiment results, we proposed six guidelines for the design of quantum machine learning models, which incorporate intuitions and heuristics from classical AI research.

It should be noted that our results are achieved with numerical simulations of the quantum model. For future research, we would like to investigate approaches that enable the Hamiltonian embedding to be run on physical devices without severely degrading the performance of the model.

Appendix. Plots of the loss and accuracy curves during the training of the model

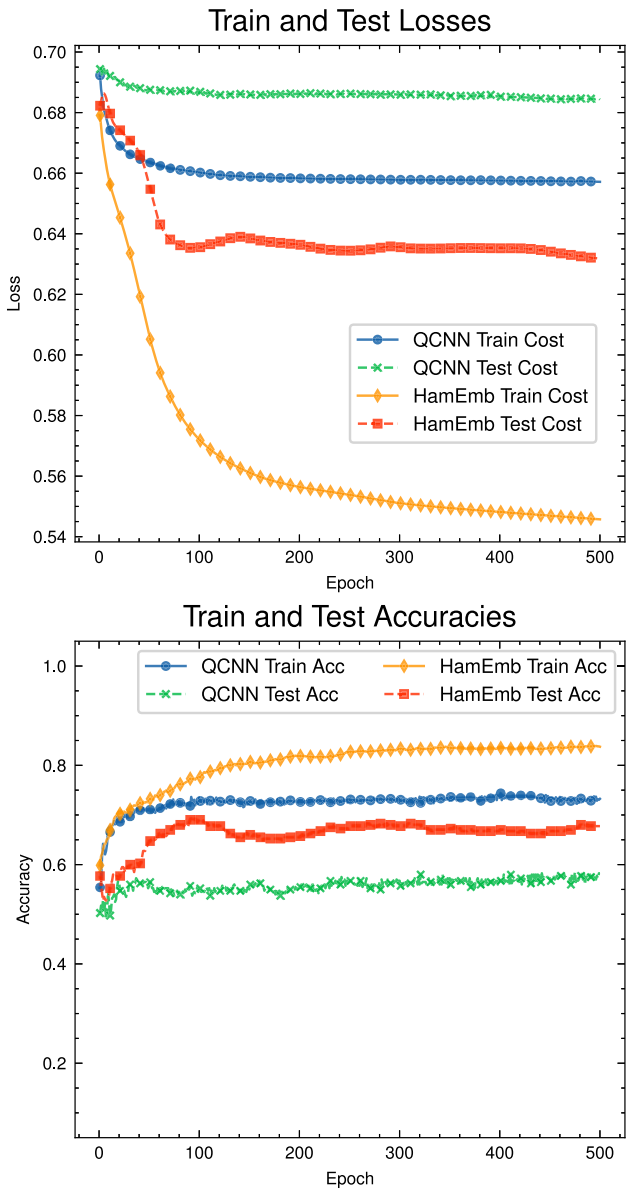


Fig. 8 Plots for the averaged train and test metrics over 20 different parameter initialisations for the Kaggle CT Medical Images dataset. Although the proposed model (HamEmb) outperforms the baseline model, we could still see the gap between performances on the train and test dataset, potentially due to the small size of the dataset

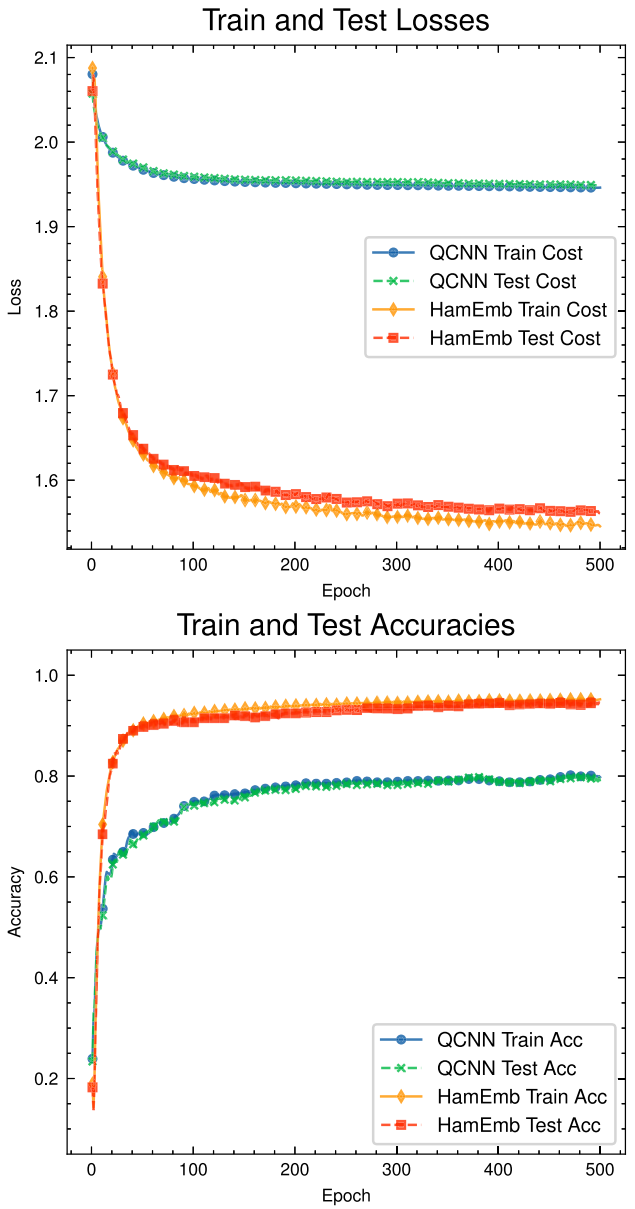


Fig. 9 Plots for the averaged train and test metrics over 20 different parameter initialisations for the Sklearn digits dataset. We can see that the model proposed (HamEmb) drastically outperforms the baseline model (QCNN with amplitude embedding)

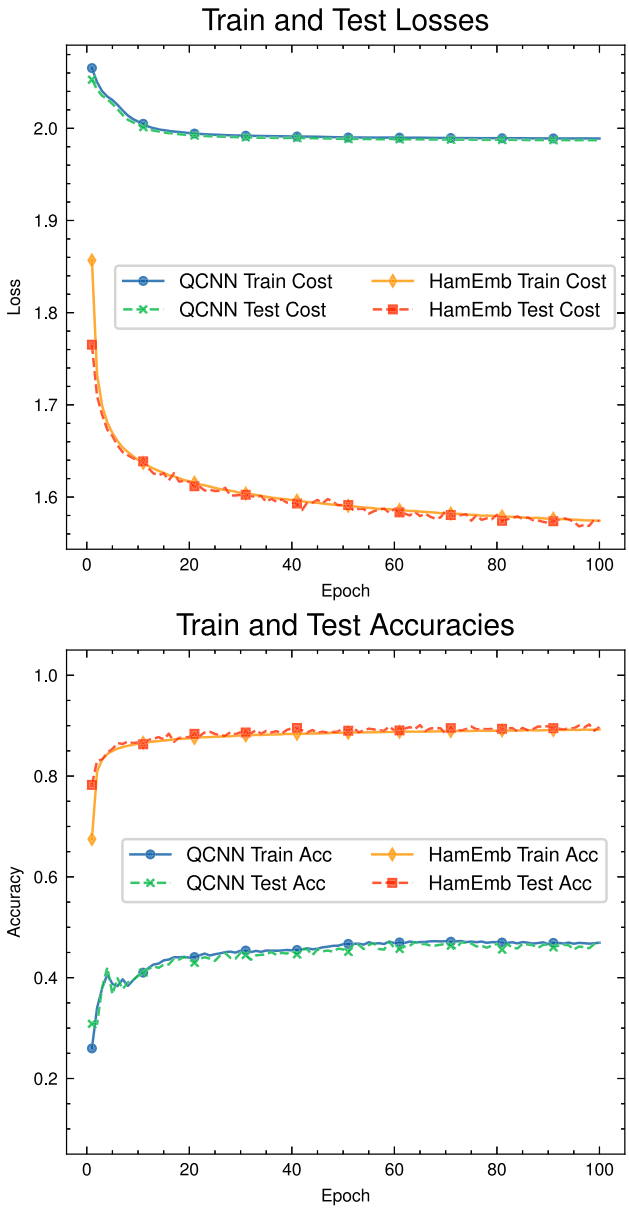


Fig. 10 Plots for the averaged train and test metrics over 5 different parameter initialisations for the MNIST handwritten digits dataset. We can see that the model proposed (HamEmb) outperforms the baseline model (QCNN with amplitude embedding). However, we can observe that both the performances of the baseline and proposed model have dropped compared to the performance shown in Fig. 9, also for handwritten-digit-type dataset, which could potentially be caused by the increased dimension of the image size in the dataset

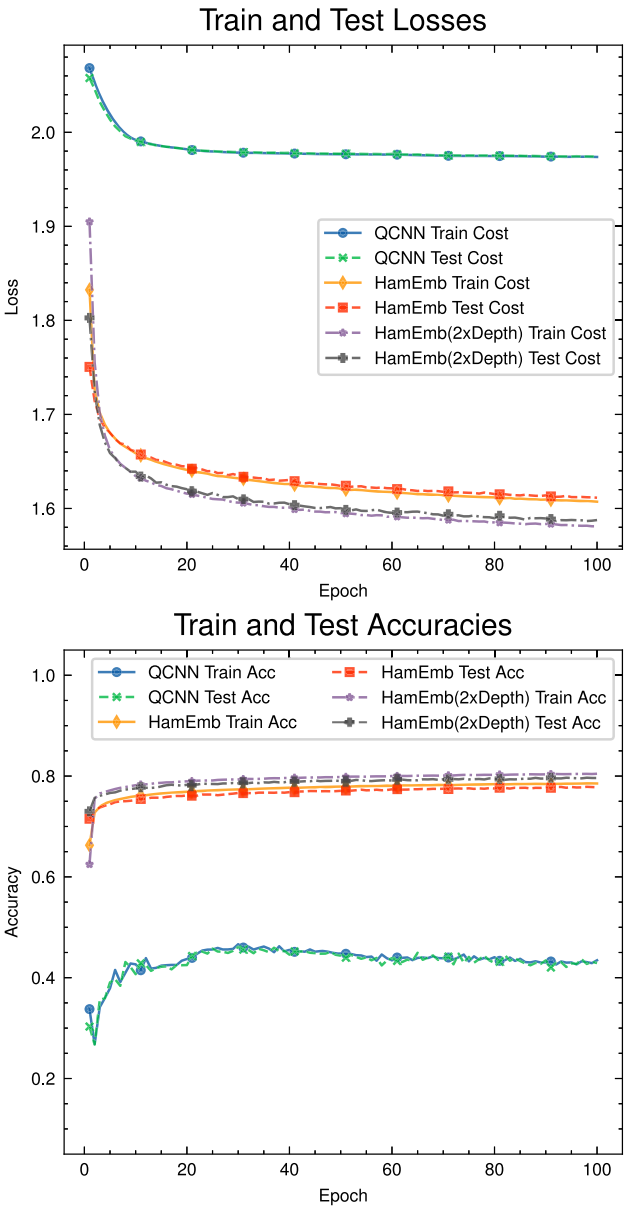


Fig. 11 Plots for the averaged train and test metrics over five different parameter initialisations for the FashionMNIST dataset. We can see that the model proposed (HamEmb) outperforms the baseline model (QCNN with amplitude embedding). However, we can observe that both the performances of the baseline and proposed model have dropped compared to the performance shown in Fig. 10, and increasing the depth two times does not increase the performance by a significant margin. This outcome could potentially be due to the decreased sparsity of the images in the FashionMNIST dataset compared to the MNIST dataset

Acknowledgements The authors acknowledge the support of IBM Quantum Hub at the University of Melbourne and the Seed Grant from the School of Computing and Information Systems, the University of Melbourne.

Author contribution P. W. proposed the research idea, wrote the code for the numerical experiments, wrote the main manuscript text, and prepared all the figures. C.R.M, L.C.L.H and U.P provided comments for the research ideas, numerical experiments and the manuscript. All authors reviewed the manuscript.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions.

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

Code availability Code available on https://github.com/peiyong-addwater/HamEmbedding.

Declarations

Conflict of interest The authors declare no competing interests.

References

Abbe E, Bengio S, Lotfi A, Sandon C, Saremi O (2024) How far can transformers reason? The locality barrier and inductive scratchpad. arXiv:2406.06467

Albertina B, Watson M, Holback C, Jarosz R, Kirk S, Lee Y, Rieger-Christ K, Lemmerman J (2016) The Cancer Genome Atlas Lung Adenocarcinoma Collection (TCGA-LUAD). Cancer Imaging Arch. https://doi.org/10.7937/K9/TCIA.2016.JGNIHEP5https://www.cancerimagingarchive.net/collection/tcga-luad/

Alpaydin E, Kaynak C (1998)Optical recognition of handwritten digits. UCI machine learning repository. https://doi.org/10.24432/C50P49

Barbero F, Banino A, Kapturowski S, Kumaran D, Araújo JGM, Vitvitskyi A, Pascanu R, Veličković P (2024) Transformers need glasses! Information over-squashing in language tasks. arXiv:2406.04267 Accessed 18-June-2024

Biamonte J, Wittek P, Pancotti N, Rebentrost P, Wiebe N, Lloyd S (2017) Quantum machine learning. Nature 549(7671):195–202. https://doi.org/10.1038/nature23474

Bowles J, Ahmed S, Schuld M (2024) Better than classical? The subtle art of benchmarking quantum machine learning models. arXiv:2403.07059 [quant-ph]

Bowles J, Wright VJ, Farkas M, Killoran N, Schuld M (2023) Contextuality and inductive bias in quantum machine learning. arXiv:2302.01365

Bradbury J, Frostig R, Hawkins P, Johnson MJ, Leary C, Maclaurin D, Necula G, Paszke A, VanderPlas J, Wanderman-Milne S, Zhang Q (2018) JAX: composable transformations of Python+NumPy programs. http://github.com/google/jax

Bubeck S, Chandrasekaran V, Eldan R, Gehrke J, Horvitz E, Kamar E, Lee P, Lee YT, Li Y, Lundberg S, Nori H, Palangi H, Ribeiro MT, Zhang Y (2023) Sparks of artificial general intelligence: early experiments with GPT-4

Cao Y, Guerreschi GG, Aspuru-Guzik A (2017) Quantum neuron: an elementary building block for machine learning on quantum computers. arXiv:1711.11240

Chung HW (2024) Shaping the future of AI from the history of transformer. https://www.youtube.com/watch?v=orDKvo8h71o

Coelho R, Sequeira A, Paulo Santos L (2024) VQC-based reinforcement learning with data re-uploading: performance and trainability 6:1–23. https://doi.org/10.1007/s42484-024-00190-z Accessed 09-Nov-2024

Cong I, Choi S, Lukin MD (2019) Quantum convolutional neural networks. Nat Phys 15(12):1273–1278. https://doi.org/10.1038/s41567-019-0648-8

Creevey FM, Hill CD, Hollenberg LCL (2023) GASP: a genetic algorithm for state preparation on quantum computers. Sci. Rep. 13(1):11956

Dao T, Fu DY, Ermon S, Rudra A, Ré C (2022) FlashAttention: fast and memory-efficient exact attention with IO-awareness arXiv:2205.14135 [cs.LG]

Dauphin YN, Fan A, Auli M, Grangier D (2017) Language modeling with gated convolutional networks

Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N (2020) An image is worth 16x16 words: transformers for image recognition at scale. arXiv:2010.11929

Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N (2021) An image is worth 16x16 words: transformers for image recognition at scale

Dziri N, Lu X, Sclar M, Li XL, Jiang L, Lin BY, West P, Bhagavatula C, Bras RL, Hwang JD, Sanyal S, Welleck S, Ren X, Ettinger A, Harchaoui Z, Choi Y (2023) Faith and fate: limits of transformers on compositionality. arXiv:2305.18654 [cs.CL]

Easom-Mccaldin P, Bouridane A, Belatreche A, Jiang R (2021) On depth, robustness and performance using the data re-uploading single-qubit classifier. IEEE Access 9:65127–65139. https://doi.org/10.1109/ACCESS.2021.3075492

Elhage N, Hume T, Olsson C, Schiefer N, Henighan T, Kravec S, Hatfield-Dodds Z, Lasenby R, Drain D, Chen C, Grosse R, McCandlish S, Kaplan J, Amodei D, Wattenberg M, Olah C (2022) Toy models of superposition. Transformer circuits thread. https://transformer-circuits.pub/2022/toy_model/index.html

Gong L-H, Pei J-J, Zhang T-F, Zhou N-R (2024) Quantum convolutional neural network based on variational quantum circuits. Opt Commun 550:129993. https://doi.org/10.1016/j.optcom.2023.129993

Goyal, A., Bengio, Y.: Inductive biases for deep learning of higher-level cognition. Proc. Math. Phys. Eng. Sci. **478**(2266) (2022). 10.1098/rspa.2021.0068

Henderson M, Shakya S, Pradhan S, Cook T (2020) Quanvolutional neural networks: powering image recognition with quantum circuits. Quantum Mach Intell 2(1):2. https://doi.org/10.1007/s42484-020-00012-y

Heredge J, Hill C, Hollenberg L, Sevior M (2024) Permutation invariant encodings for quantum machine learning with point cloud data. Quantum Mach Intell 6(1):1–14. https://doi.org/10.1007/s42484-024-00156-1

He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition
He K, Zhang X, Ren S, Sun J (2016) Identity mappings in deep residual networks. In: Computer vision – ECCV 2016. Springer, pp 630–645. https://doi.org/10.1007/978-3-319-46493-0_38
Hoffmann J, Borgeaud S, Mensch A, Buchatskaya E, Cai T, Rutherford E, de Las Casas D, Hendricks LA, Welbl J, Clark A, Hennigan T, Noland E, Millican K, van den Driessche G, Damoc B, Guy A, Osindero S, Simonyan K, Elsen E, Rae JW, Vinyals O, Sifre L (2022) Training compute-optimal large language models
Hoffmann J, Borgeaud S, Mensch A, Buchatskaya E, Cai T, Rutherford E, de Las Casas D, Hendricks LA, Welbl J, Clark A, Hennigan T, Noland E, Millican K, van den Driessche G, Damoc B, Guy A, Osindero S, Simonyan K, Elsen E, Rae JW, Vinyals O, Sifre L (2022) Training compute-optimal large language models. arXiv:2203.15556 [cs.CL]
Humayun AI, Balestriero R, Baraniuk R (2024) Deep networks always grok and here is why. arXiv:2402.15555 [cs.LG]
Hur T, Kim L, Park DK (2022) Quantum convolutional neural network for classical data classification. Quantum Mach Intell 4(1):3. https://doi.org/10.1007/s42484-021-00061-x
Itseez (2015) Open source computer vision library. https://github.com/itseez/opencv
Jaderberg B, Anderson LW, Xie W, Albanie S, Kiffner M, Jaksch D (2022) Quantum self-supervised learning. Quantum. Sci Technol 7(3):035005. https://doi.org/10.1088/2058-9565/ac6825
Kambhampati S, Valmeekam K, Guan L, Stechly K, Verma M, Bhambri S, Saldyt L, Murthy A (2024) LLMs can't plan, but can help planning in LLM-Modulo frameworks. arxiv:2402.01817
Kaplan J, McCandlish S, Henighan T, Brown TB, Chess B, Child R, Gray S, Radford A, Wu J, Amodei D (2020) Scaling laws for neural language models
Kerenidis I, Landman J, Prakash A (2019) Quantum algorithms for deep convolutional neural networks
Khatun A, Usman M (2024) Quantum transfer learning with adversarial robustness for classification of high-resolution image datasets
Khatun A, Usman M (2024) Quantum transfer learning with adversarial robustness for classification of high-resolution image datasets. arXiv:2401.17009
Kim Y (2014) Convolutional neural networks for sentence classification
Kingma DP, Ba J (2017) Adam: a method for stochastic optimization
Kleiner J (2024) Consciousness qua mortal computation. arXiv:2403.03925 [q-bio.NC]
Kottmann K, Calderon LM, Weber M (2022) Generalization in QML from few training data. Xanadu. Accessed: 07-Feb-2024
Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 25. Accessed 15-Sept-2024
Le PQ, Dong F, Hirota K (2011) A flexible representation of quantum images for polynomial preparation, image compression, and processing operations. Quantum Inf Process 10(1):63–84. https://doi.org/10.1007/s11128-010-0177-y
LeCun Y, Boser B, Denker JS, Henderson D, Howard RE, Hubbard W, Jackel LD (1989) Backpropagation applied to handwritten zip code recognition. Neural Comput 1(4):541–551. https://doi.org/10.1162/neco.1989.1.4.541
LeCun Y, Cortes C, Burges C (2010) MNIST handwritten digit database. ATT Labs 2. http://yann.lecun.com/exdb/mnist
Lloyd S, Mohseni M, Rebentrost P (2014) Quantum principal component analysis. Nat Phys 10(9):631–633. https://doi.org/10.1038/nphys3029
maintainers T, contributors (2016) TorchVision: PyTorch's Computer Vision library. GitHub
Nezhurina M, Cipolina-Kun L, Cherti M, Jitsev J (2024) Alice in wonderland: simple tasks showing complete reasoning breakdown in state-of-the-art large language models. arXiv:2406.02061
Nguyen D-K, Assran M, Jain U, Oswald MR, Snoek CGM, Chen X (2024) An image is worth more than 16x16 patches: exploring transformers on individual pixels. arXiv:2406.09415 [cs.CV]
Ofir Press, Zhang M, Min S, Schmidt L, Smith NA, Lewis M (2022) Measuring and narrowing the compositionality gap in language models. arXiv:2210.03350
Olah C (2014) Neural networks, manifolds, and topology. https://colah.github.io/posts/2014-03-NN-Manifolds-Topology/. Accessed: 20-Feb-2024
Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: machine learning in Python. J Mach Learn Res 12:2825–2830
Pérez-Salinas A, Cervera-Lierta A, Gil-Fuster E, Latorre JI (2020) Data re-uploading for a universal quantum classifier. Quantum 4(226):226. https://doi.org/10.22331/q-2020-02-06-226
Prince SJD (2023)Understanding deep learning. The MIT Press, Cambridge, MA. http://udlbook.com
Radford A, Narasimhan K, Salimans T, Sutskever I (2018)Improving language understanding with unsupervised learning. https://openai.com/index/language-unsupervised/ Accessed 11-July-2024
Riaz F, Abdulla S, Suzuki H, Ganguly S, Deo RC, Hopkins S (2023) Accurate image multi-class classification neural network model with quantum entanglement approach. Sensors 23(5). https://doi.org/10.3390/s23052753
Schuld M, Killoran N (2022) Is quantum advantage the right goal for quantum machine learning? PRX Quantum 3(3):030101. https://doi.org/10.1103/PRXQuantum.3.030101
Schuld M, Petruccione F (2021) Representing data on a quantum computer. In: Schuld M, Petruccione F (eds) Machine learning with quantum computers, pp 147–176. Springer, Cham. https://doi.org/10.1007/978-3-030-83098-4_4
Scott Mader K (2017) CT medical images. https://www.kaggle.com/datasets/kmader/siim-medical-images
Shen K, Jobst B, Shishenina E, Pollmann F (2024) Classification of the Fashion-MNIST dataset on a quantum computer
Shen K, Jobst B, Shishenina E, Pollmann F (2024) Classification of the Fashion-MNIST dataset on a quantum computer. arXiv:2403.02405 [quant-ph]
Sutton R (2019) The bitter lesson. http://www.incompleteideas.net/IncIdeas/BitterLesson.html. Accessed: 23-Dec-2023
Teney D, Nicolicioiu A, Hartmann V, Abbasnejad E (2024) Neural redshift: random networks are not random functions
Tolstobrov A, Fedorov G, Sanduleanu S, Kadyrmetov S, Vasenin A, Bolgar A, Kalacheva D, Lubsanov V, Dorogov A, Zotova J, Shlykov P, Dmitriev A, Tikhonov K, Astafiev OV (2024) Hybrid quantum learning with data reuploading on a small-scale superconducting quantum simulator 109:012411. https://doi.org/10.1103/physreva.109.012411. Accessed 09-Nov-2024
Tong WL, Pehlevan C (2024) MLPs learn in-context. arXiv:2405.15618 [cs.LG]
Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need arXiv:1706.03762 [cs.CL]
Verma M, Bhambri S, Kambhampati S (2024) On the brittle foundations of ReAct prompting for agentic large language models. arXiv:2405.13966
Wang B, Yue X, Su Y, Sun H (2024) Grokked transformers are implicit reasoners: a mechanistic journey to the edge of generalization. arXiv:2405.15071 [cs.CL]
West MT, Nakhl AC, Heredge J, Creevey FM, Hollenberg LCL, Sevior M, Usman M (2023) Drastic circuit depth reductions with

preserved adversarial robustness by approximate encoding for quantum machine learning
Wu W, Morris JX, Levine L (2024) Do language models plan ahead for future tokens? arxiv:2404.00859
Xiao H, Rasul K, Vollgraf R (2017) Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms
Yan F, Iliyasu AM, Venegas-Andraca SE (2016) A survey of quantum image representations. Quantum Inf Process 15(1):1–35. https://doi.org/10.1007/s11128-015-1195-6
Yang R, Bosch S, Kiani B, Lloyd S, Lupascu A (2023) Analog quantum variational embedding classifier. Phys Rev Appl 19(5):054023. https://doi.org/10.1103/PhysRevApplied.19.054023
Yu Z, Chen Q, Jiao Y, Li Y, Lu X, Wang X, Yang JZ (2023) Provable advantage of parameterized quantum circuit in function approximation. arXiv:2310.07528
Zaman K, Ahmed T, Hanif MA, Marchisio A, Shafique M (2024) A comparative analysis of hybrid-quantum classical neural networks. arXiv:2402.10540
Zhou N-R, Zhang T-F, Xie X-W, Wu J-Y (2023) Hybrid quantum-classical generative adversarial networks for image generation via learning discrete distribution 110:116891. https://doi.org/10.1016/j.image.2022.116891 Accessed 10-Nov-2024
Zhou H, Bradley A, Littwin E, Razin N, Saremi O, Susskind J, Bengio S, Nakkiran P (2023) What algorithms can transformers learn? A study in length generalization. arXiv:2310.16028