*Supplementary Material for*

Hybrid quantum-classical convolutional neural network for phytoplankton classification

Shangshang Shi†, Zhimin Wang†,\*, Ruimin Shang, Yanan Li, Jiaxin Li, Guoqiang Zhong, and Yongjian Gu\*

Faculty of Information Science and Engineering, Ocean University of China, Qingdao, China

**\*Correspondence:** Zhimin Wang, wangzhimin@ouc.edu.cn; Yongjian Gu, yjgu@ouc.edu.cn

†These authors contribute equally to this work and share first authorship

# Template CNN and ResNet models

In this work, we introduce two novel models for phytoplankton classification: the quantum-classical convolutional neural network (QCCNN) and the quantum-classical residual network (QCResNet). The QCCNN is constructed based on a classical CNN model, while the QCResNet is based on a ResNet model. Figure 1 illustrates the architecture of a typical CNN, which includes two convolutional layers, a max-pooling layer, and two fully connected layers. We use this CNN as a template framework to construct the QCCNN in our experiments. Figure 2 shows the architecture of a ResNet, which comprises two residual units, an adaptive average pooling layer, and a fully connected layer. The shortcut connections in the ResNet use a 1×1 convolution operation. We use this ResNet as a template framework to construct the QCResNet in our experiments.



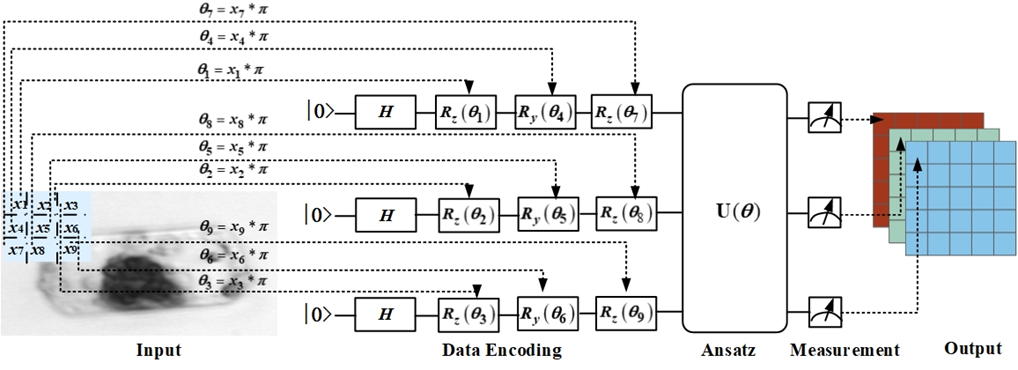
Supplementary Figure 1 Architecture of the template CNN that is used to construct QCCNN.

H:\Article\13. QCCNN\画图\Fig2.tif

Supplementary Figure 2 Architecture of the template ResNet that is used to construct QCResNet.

# Another architecture of quantum convolutional layer

The QCResNet models use quantum residual unit as the basic building block, as shown in Figure 5 in the main text. The quantum residual unit #1 uses a filter window size of 3×3, and outputs three feature channels. To accomplish this, the dense angle encoding method is applied, which employs 3 qubits to embed 9 data elements. Figure 3 shows the detailed architecture of the quantum convolutional layer used in the quantum residual unit #1.



Supplementary Figure 3 Architecture of the quantum convolutional layer used in the quantum residual unit #1 in the QCResNet models.

# Comparison between quantum and classical models

# In our experiments, six neural networks are evaluated using the phytoplankton dataset, which are the template CNN (Supplementary Figure 1), template ResNet (Supplementary Figure 2), QCCNN-1 and QCCNN-2 (Figure 2 in main text), QCResNet-1 and QCResNet-2 (Figure 5 in main text). The detailed information of these models is summarized in the following two tables.

# Table 1 Architecture and parameters of CNN and QCCNN models

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Network | Level | Layers | Input | Filter, Channels | # of parameters |
| Template  CNN | 1 | Conv2D+ReLU | 20×20×1 | 2×2, 4 | 16 |
| 2 | Conv2D+ReLU | 11×11×4 | 3×3, 9 | 324 |
| 3 | MaxPooling2D | 11×11×9 | — | — |
| 4 | FC1+ReLU | 5×5×9 | 120 | 27000 |
| 5 | FC2 (Output) | 120 | 4 | 480 |
| QCCNN-1 | **1** | **Quantum Conv 1** | 20×20×1 | 2×2, 4 | **28** |
| 2 | Conv2D+ReLU | 11×11×4 | 3×3, 9 | 324 |
| 3 | MaxPooling2D | 11×11×9 | — | — |
| 4 | FC1+ReLU | 5×5×9 | 120 | 27000 |
| 5 | FC2 (Output) | 120 | 4 | 480 |
| QCCNN-2 | **1** | **Quantum Conv 1** | 20×20×1 | 2×2, 4 | **28** |
| **2** | **Quantum Conv 2** | 10×10×4 | 3×3, 9 | **108** |
| 3 | MaxPooling2D | 10×10×9 | — | — |
| 4 | FC1+ReLU | 5×5×9 | 120 | 27000 |
| 5 | FC2 (Output) | 120 | 4 | 480 |

# Table 2 Architecture and parameter of ResNet and QCResNet models

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Network | Level | Layers | Input | Filter, Channels | # of parameters |
| Template  ResNet | 1 | Residual Unit | 20×20×1 | 3×3, 3 | 111 |
| 2 | Residual Unit | 20×20×3 | 3×3, 9 | 999 |
| 3 | Adaptive Average Pooling2D | 20×20×9 | — | — |
| 4 | FC (Output) | 5×5×9 | 4 | 900 |
| QCResNet-1 | **1** | **Quantum Residual Unit 1** | 20×20×1 | 3×3, 3 | **102** |
| 2 | Residual Unit | 20×20×3 | 3×3, 9 | 999 |
| 3 | Adaptive Avg. Pooling2D | 20×20×9 | — | — |
| 4 | FC (Output) | 5×5×9 | 4 | 900 |
| QCResNet-2 | **1** | **Quantum Residual Unit 1** | 20×20×1 | 3×3, 3 | **102** |
| **2** | **Quantum Residual Unit 2** | 20×20×3 | 3×3, 9 | **864** |
| 3 | Adaptive Avg. Pooling2D | 20×20×9 | — | — |
| 4 | FC (Output) | 5×5×9 | 4 | 900 |

# Shots used in the quantum measurement

As discussed in Section 2 of the main text, the probability or expectation in the process of quantum measurement of QNN is estimated by repeating the measurement a certain number of times, the number of repetitions is referred to as the number of shots. An appropriate number of shots must be set beforehand to ensure sufficient statistics while avoiding excessive use of computing resources. Moreover, according to classical deep learning theory, adding small noise in the training process can improve the robustness and generalization performance of the learning model. The statistical error introduced by the limited number of shots can serve as this added noise.

To this end, we conducted a preliminary experiment using QCCNN-1. The experimental results are shown in Figure 4. We find that when the number of shots is set to 1500, QCCNN-1 achieves the best classification accuracy. This demonstrates the importance of choosing an appropriate number of shots for QNN. In our experiments, we used 1500 shots for quantum measurement.

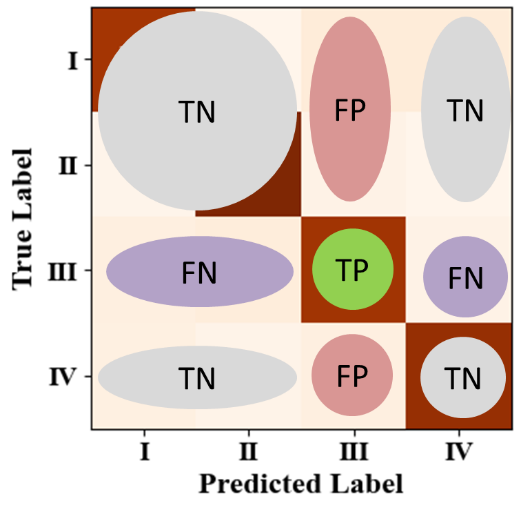


Supplementary Figure 4 Test accuracy of QCCNN-1 on the phytoplankton classification using different number of shots in the quantum measurement.

# Evaluation metrics

In the main text, we compute the confusion matrices for the results obtained by the six neural networks. Using these matrices, we calculate eight evaluation metrics to comprehensively assess the classification performance of each model for each category. These metrics include precision, recall, F1-score, specificity, false positive rate (FPR), false discovery rate (FDR) and false negative rate (FNR).

First, let’s specify the definition of four values used to calculate the eight evaluation metrics: TP*k*, TN*k*, FP*k* and FN*k*. In multi-class classification, TP*k* represents the number of samples of class *k* that are correctly identified. FP*k* represents the number of samples that are falsely identified as class *k* by other classes. FN*k* represents the number of samples of class *k* that are falsely identified as other classes. TN*k* represents the remaining samples. Figure 5 illustrates the areas of these four values for class III in the confusion matrix (see Figure 9 in the main text).



Supplementary Figure 5 Areas of TP*k*, TN*k*, FP*k* and FN*k* for class III in the confusion matrix.

(1) Accuracy is the most widely used metric in multi-class classification. It measures the proportion of samples that are correctly classified. The formula for calculating accuracy is as follows:

, (1)

where *K* represents the number of sample types *k* = 1, 2, …, *K*, and *N* is the total number of training samples.

(2) Precision is a metric that measures the reliability of a model when predicting a positive outcome for an individual. For a specific class *k*, precision is calculated as follows:

. (2)

In the case of multi-class classification, macro precision is computed as the arithmetic mean of the precision values for each class, that is,

. (3)

(3) Recall measures the percentage of actual positive samples that are correctly predicted as positive. The formula for calculating the recall of each class and the macro recall are as follows:

. (4)

(4) Macro F1 Score is the harmonic average of macro precision and macro recall. It takes into account both the accuracy and recall of the classification model. It is calculated as

. (5)

(5) Specificity measures the proportion of actual negative samples that are correctly predicted as negative. The formulas for calculating the specificity of each class and the macro specificity are as follows:

. (6)

(6) The False Positive Rate (FPR) indicates the proportion of actual negative samples that were incorrectly predicted as positive. The formulas for calculating FPR can be expressed as

. (7)

(7) False Discovery Rate (FDR) reflects the proportion of negative samples among the samples that are identified as positive. The formulas are as follows:

. (8)

(8) False Negative Rate (FNR) is the proportion of actual positive samples that were incorrectly predicted as negative. The formulas for calculating the FNR of each class and the macro FNR are as follows:

. (9)