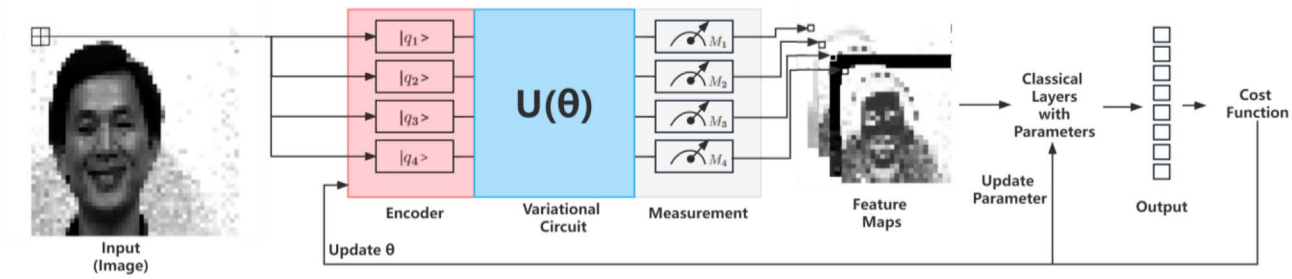

Fig. 5. Figure shows the overall structure of the model in the experiment. The original image is encoded into the quantum circuit with a 2×2 area and a stride of 2. U(θ) represents a unitary operation consisting of quantum gates. For our MGQCNN, it is a unitary operation composed of R_X gate and R_Z gate, each gate contains one parameter, a total of 8 as shown in Fig. 4 above. After different measurement methods, $24 \times 24 \times 4$ feature maps are obtained, the classic fully connected layer is used for classification, and the parameter θ is updated after cost function. The model approaches the target after many iterations.

For classic models, there is a universal approximation theorem (UAT) [57] to support its approximation capabilities. Similarly, for quantum models, UAT can be used to demonstrate approximation capabilities. According to [58], a quantum analog can be constructed on the basis of UAT. For any function $g : \chi \to \mathbb{R}$ and for any $\varepsilon > 0$, there exist $n \in \mathbb{N}$ and $w \in \mathbb{R}$ such that

$$\left|\omega\psi_n(x) - g(x)\right| < \varepsilon. \tag{4}$$

For all x in χ and $\psi_n(x)$ is the basis function. Based on quantum UAT, we can think that our quantum circuits can fit the functions.

C. Decoder

In the decoder, the processed quantum features will be measured with the help of Pauli Z gates. The expected value of each qubit will be derived from repeated measurements. Through the decoder, the quantum data is converted into classical data, and the classical data can be used as the input for the next layer to continue processing.

Here we consider two cases, measuring all qubits and measuring one qubit but setting multiple convolution kernels. Both measurements have their potential advantages. Measuring a single qubit but setting up multiple convolution kernels makes the quantum convolution layer more similar to the classical convolution operation, but the number of qubits required grows exponentially, as does the number of parameters, which may increase the training time. We will test both measures in experiments to analyze their impact on the overall performance of the model.

After a convolution operation with quantum filters, we obtain a set of feature maps of the original images. Since our quantum architecture is easy to expand, classical neural networks or quantum neural networks can be used to continue processing and further extract features. However, in this study, since our goal is to verify the performance of our architecture, we do not extend the network deeper. Keeping the model structure simple allows us to better analyze the experimental results. We directly feed the resulting feature map into a classical fully connected layer for classification. Through the cost function, the model can update the parameters on the eight quantum rotation gates in the variational quantum circuit. By repeating this process many times, the parameters are continuously updated, and the model is continuously fitted to the objective function we need.

IV. Experimental Setup And Result Evaluation

In this section, we present our experiments using the method described in the previous section, where the results obtained will be analyzed. Our first experiment will use the Yale Face Database [26], a small database. Due to the limitations of current QML simulation algorithms, it is difficult for us to use large databases, and we think using Yale Face Database is a suitable challenge to demonstrate the lowest performance bounds for systems using minimal parameters. After this, as an increase in difficulty, our experiments will use the ORL face dataset [27]. The ORL face dataset has more data than the Yale Face Database, which is more challenging for our method. We believe that the application of our method to studies on small databases is valuable for its further potential application in medicine.

A. Experimental Setup

We choose the Yale face database [26] and the ORL database of faces [27] as the experimental data. The Yale face dataset was created by Yale University and contains 15 people, each of whom has 11 face images with different expressions, poses and lighting: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink. In a total of 165 images, the original size of each image is 320×243 pixels. The ORL database of faces contains a set of images of human faces taken in the laboratory. Each of the 40 different subjects had ten different images, varying lighting, facial expressions (eyes open/closed, smiling/not smiling), and facial details (with/without glasses). The size of each image is 92×112 pixels, and each pixel has 256 gray levels.

In this experiment, we test five methods: our proposed MG-QCNN, VQNN [19], HQNN [52], HQCCNN [50], and QCNN [23]. Fig. 5 illustrates the basic architecture of the MG-QCNN. It consists of a VQC layer with one filter and a fully-connected layer with 15 classes. The input data is a 48×48 face image. The kernel size and stride of the VQC layer are chosen to be 2×2 and 2, respectively. An input image of 48×48 pixels is encoded into a four-qubit state using the R_Y rotation gate and then entangled through a CNOT gate with trainable parameters. The decoding part is designed in two ways: either all qubits are measured or only one qubit is measured but with four convolution kernels. The quantum convolutional layer will thus extract a $24 \times 24 \times 4$ feature tensor from the 48×48

TABLE III
TEST RESULT OF THE MODELS WITH THE DIFFERENT STRUCTURES

	TEST LOSS		MEAN ACCURACY		MAX ACCURACY		RUNNING TIME		MEMORY	
DATASETS	Yale	ORL	Yale	ORL	Yale	ORL	Yale	ORL	Yale	ORL
MG-QCNN-ALL	0.347	0.307	96.000%	95.959%	97.778%	96.667%	29567s	69169s	3.837GB	5.086GB
3×3 FLITER	0.398	0.521	93.333%	92.917%	97.778%	96.667%	56158s	167732s	7.663GB	8.137GB
RX-REDUCTION	0.346	0.301	89.667%	89.333%	93.333%	94.167%	28423s	70285s	3.677GB	5.011GB
RZ-REDUCTION	0.352	0.317	91.333%	90.333%	95.556%	94.167%	29016s	67293s	3.701GB	4.863GB
NO-ENTANGLEMENT	0.341	0.299	88.222%	88.417%	91.111%	91.667%	28774s	68024s	3.762GB	4.882GB

best performance on both datasets. It takes the least amount of time and consumes the least amount of resources, which has advantages. In addition, for comparison, we also trained a classical CNN on The ORL database of faces, replacing the quantum layer with the classical convolutional layer, and the accuracy reached 95%. Compared with the classical CNN, our proposed MG-QCNN-All model has an accuracy advantage. According to the comparison between the MG-QCNN-All model and the MG-QCNN-Single model, the performance of the MG-QCNN-All model comprehensively exceeds the MG-QCNN-Single model, and the structure similar to the classical CNN does not bring better performance to the quantum model but increases the model training time. This is due to the imitation of the classical CNN structure resulting in a fourfold increase in the parameters of the filter to 32 parameters. With so many parameters, the speed of training will be greatly slowed down. Especially when using Yale Face Database as the dataset, MG-QCNN-Single is difficult to converge. However, the performance of the QCNN model is second only to the MG-QCNN-All model, and it converges faster than the MG-QCNN-All model in the training phase.

The problem of difficulty in training is also reflected in HQCCNN. Its structure is more complex than MGQCNN and it takes up more resources during training. In general, the training phase converges faster, and the model should perform better on the test set. If the model does not perform well, it may be overfitting. Therefore, we changed the learning rate and epoch of training, tried different combinations, and the performance of the model did not improve. We think the reason for this phenomenon is that the number of parameters of the QCNN model is too large, the data we use in the experiment is limited, and the QCNN model cannot fit the data well. However, as the amount of data increases, the training time of the QCNN model increases substantially. Due to our resource limitation, we were unable to train and test models on large datasets. We adopt the quantum neural network in the hope that the mechanism of quantum computing can provide help for the marginal effect problem and improve the efficiency of the neural network. From our experimental results, it does not seem to be a reasonable choice to use a quantum model with a similar structure to the classical CNN. By contrast, the VQNN model, HQNN and the MG-QCNN-All model have obvious advantages in terms of training speed and hardware resource usage. Even with only one filter, the quantum layer can convert the input 2-D image into four feature maps, and output the correlation between the channels of the feature maps under the action of quantum entanglement.

Overall, from the perspective of loss functions, the results of each quantum model in multiple experiments are relatively stable. However, from the perspective of average accuracy and maximum accuracy, the robustness of HQNN's performance is relatively poor. We believe this is due to the fact that HQNN has fewer quantum gates carrying parameters and does not fully cover every qubit. Apart from this, the quantum models exhibit similar robustness and are generally within an acceptable range.

The VQNN model with four parameters per filter is less accurate and takes longer to train than the MG-QCNN-All model with eight parameters per filter. We analyze that this may be because the quantum circuits of the VQNN model are randomly generated. As a result, the VQNN model requires a large number of random circuits to be generated during training, thereby slowing down its training speed. The fixed design circuits of the MG-QCNN-All model, however, eliminate the need for this step and thus improve training efficiency.

Our experimental results for different structural modifications of our method are presented in Table III. Judging from the results, increasing the size of the convolution kernel does not have a positive impact on the experimental results, but instead reduces the efficiency of the model. We believe this is because larger convolution kernels are currently less efficient for existing loss functions and optimization methods, and there are concerns about "barren plateaus" with more qubits. This problem needs to be solved by redesigning the loss function. In the experiment of removing the R_X gate and R_Z gate, the accuracy of the model dropped significantly. Although the required memory and time decreased, this was due to the overall parameter decrease. In comparison, removing the R_X gate causes a greater decrease in accuracy. This is because the R_X gate affects the Z-axis in Bloch space, thus affecting the final measurement results. In the experiment of removing quantum entanglement, the accuracy of the model dropped significantly, which proves that the correlation between the data brought by quantum entanglement is very important, which is also reflected in the experiment of the measurement method.

We extract the incorrectly recognized images from our model experiments for analysis. Fig. 7 shows the accuracy results for specific classes in the data set. Relatively speaking, for the Yale data set, there are a large number of shadows in the background of some data, such as subject 08, which has a higher error rate than other images. For the ORL data set, there is little difference in error rates between faces without glasses and those with glasses. However, for subject 31, the reflection of glasses in five images interferes with the model, and the error rates of these