Land Cover Classification From Sentinel-2 Images With Quantum-Classical Convolutional Neural Networks

Fan Fan, Yilei Shi, *Member, IEEE*, and Xiao Xiang Zhu, *Fellow, IEEE*

Abstract—**Exploiting machine learning techniques to automatically classify multispectral remote sensing imagery plays a significant role in deriving changes on the Earth's surface. However, the computation power required to manage large Earth observation data and apply sophisticated machine learning models for this analysis purpose has become an intractable bottleneck. Leveraging quantum computing provides a possibility to tackle this challenge in the future. This article focuses on land cover classification by analyzing Sentinel-2 images with quantum computing. Two hybrid quantum-classical deep learning frameworks are proposed. Both models exploit quantum computing to extract features efficiently from multispectral images and classical computing for final classification. As proof of concept, numerical simulation results on the LCZ42 dataset through the TensorFlow Quantum platform verify our models' validity. The experiments indicate that our models can extract features more effectively compared with their classical counterparts, specifically, the convolutional neural network (CNN) model. Our models demonstrated improvements, with an average test accuracy increase of 4.5% and 3.3%, respectively, in comparison to the CNN model. In addition, our proposed models exhibit better transferability and robustness than CNN models.**

Index Terms—**Earth observation (EO), land cover classification, multispectral imagery, quantum circuit, quantum machine learning (QML), remote sensing, sentinel-2 data.**

I. Introduction

LAND use and land cover (LULC) classification, an important field of Earth observation (EO) data analysis, attracts great attention in the remote sensing community, which contributes to tracking changes on the surface of the Earth resulting from natural processes, human activities, etc. Multispectral images, such as Sentinel-2 data having 13 spectral bands, play an important role in land cover classification tasks.

Numerous studies have attempted to utilize different machine learning techniques to extract features from multispectral remote sensing imagery for land cover classification [1]. Belgiu et al. [2] and Mountrakis et al. [3] summarized the contributions regarding using random forest and support vector machine algorithms for land cover classification. In recent years, deep learning, which extracts critical features automatically from images without feature engineering, has achieved significant success in land cover classification [4]. For example, the convolutional neural network (CNN), which can automatically extract high-level features from images by applying various filters in its sequential structure, draws great attention in the remote-sensing community due to its promising performances.

To achieve accurate LULC mapping results, researchers have been exploring various techniques and data sources. Bouslihim et al. [5] evaluated the capabilities of Landsat-9 data to generate LULC maps. Pan et al. [6] presented a CNN-based model that analyzes multispectral LiDAR data as a new data source for LULC classification. Dou et al. [7] proposed feature relations map learning to enhance the separability of different entities in an image for improving classification accuracy, and Chen et al. [8] utilized the attention mechanism between channels in the feature fusion part of a U-Net for Sentinel-2 A images analysis. In addition, rather than only relying on the features from a single-date image for LULC classification, some studies [9], [10] attempt to leverage time-series EO data for accurate classification. Moreover, apart from investigating a single-classifier structure for recognizing various LULC categories, studies, such as Dou et al.'s [11] work, explore multiple classifier systems to merge the benefits of different base classifiers for more accurate LULC mapping results.

However, the integration of sophisticated mechanisms in deep learning models for more accurate LULC classification comes at the cost of increased computational power required for training such models. Meanwhile, the advances in remote sensing technologies lead EO into the big data era, resulting in the rapid growth of EO data for analysis. These lead to an intractable challenge for classical computers [12], and quantum computing might provide a solution to tackle this challenge in the future since the main expectation of quantum computing is to solve classically expensive problems efficiently [13].

Manuscript received 29 November 2022; revised 20 April 2023, 19 June 2023, 30 October 2023, and 17 February 2024; accepted 23 May 2024. Date of publication 18 July 2024; date of current version 24 July 2024. The work of X. Zhu was supported by the German Federal Ministry of Education and Research (BMBF) in the framework of the international future AI lab "AI4EO – Artificial Intelligence for Earth Observation: Reasoning, Uncertainties, Ethics and Beyond" (grant number: 01DD20001) and by the Munich Center for Machine Learning. *(Corresponding author: Xiao Xiang Zhu.)*

Fan Fan is with the Chair of Data Science in Earth Observation, Technical University of Munich (TUM), 80333 Munich, Germany, and also with the Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), 82234 Weßling, Germany (e-mail: fan.fan@dlr.de).

Xiao Xiang Zhu is with the Chair of Data Science in Earth Observation, Technical University of Munich (TUM), 80333 München, Germany, and also with the Munich Center for Machine Learning, 80333 Munich, Germany (e-mail: xiaoxiang.zhu@tum.de).

Yilei Shi is with the School of Engineering and Design, Technical University of Munich, 80333 Munich, Germany (e-mail: yilei.shi@tum.de).

Our models and code are available at: https://github.com/zhu-xlab/LULC_MSI_QCNN

Digital Object Identifier 10.1109/JSTARS.2024.3411670

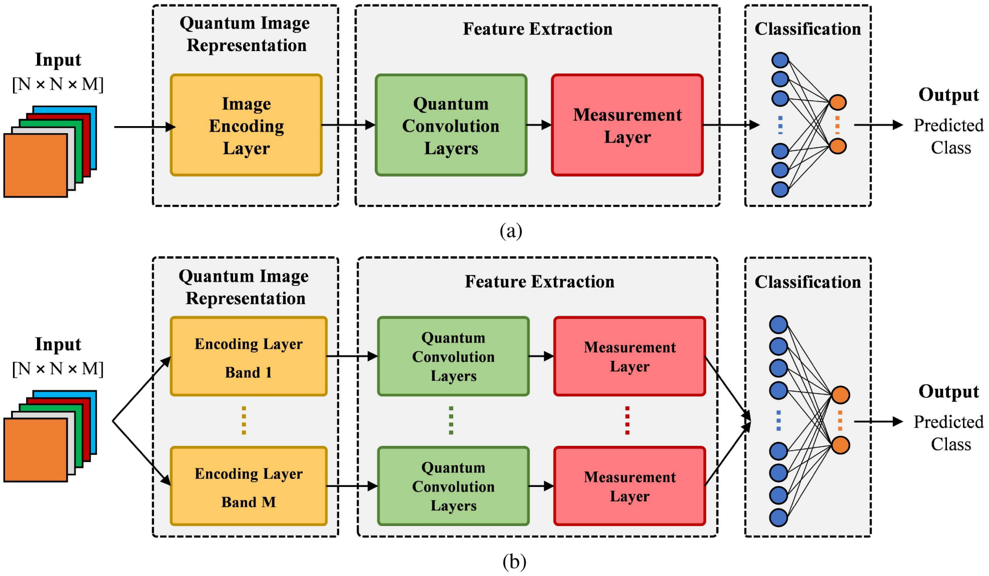


Fig. 1. Structures of the two proposed models for classifying M-spectral image with the size of $N \times N$. (a) MQCNN model. (b) FQCNN model.

The field integrating machine learning and quantum computing is quantum machine learning (QML). Regarding applying QML for land cover classification, some studies use quantum annealers [14], [15]. Besides that, exploiting quantum circuits also attracts great attention. Gawron et al. [16] used the PCA algorithm to obtain features from multispectral images and propose a quantum neural network model to classify land cover based on the extracted features. Zaidenberg et al. [17] investigated a hybrid model to classify the EO data, in which a CNN model is applied to extract features from multispectral remote sensing data, and a quantum circuit is used for the final classification task. Sebastianelli et al. [18] proposed a hybrid QCNN model with three sequential parts to analyze multispectral images for land cover classification. Specifically, one classical CNN extracts the features from images first, then one quantum circuit transforms the extracted features, and one classical dense layer is for the final prediction. Otgonbaatar et al. [19] used the VGG16 with a convolutional autoencoder to extract features from Sentinel-2 data first. Then, they use a parameterized quantum circuit (PQC) for the final classification. Abdel-Khalek et al. [20] proposed a hybrid model named QNN-MLAIC that uses Faster R-CNN with Inception with Residual Network-v2 model as the baseline model for object detection and a QNN model for classification. Note that the aforementioned studies mainly rely on classical algorithms for feature extraction and quantum computing for feature transformation or final prediction. Jing et al. [21] proposed two quantum-convolutional ansaetze (HQconv and FQconv) which can extract features from RGB images for classification. In addition, studies about using quantum computing to transform the input EO data for further analysis are also undertaken. Gupta et al. [22] utilized the projected quantum kernels to reduce the large training set dimension to a smaller classical space for land cover classification with sentinel-2 data.

Despite various contributions related to QML made so far, it is still unclear whether QML algorithms are superior to their classical counterparts. In addition, most studies, such as [18], [19], and [20] as introduced before, still rely on classical algorithms to extract features from EO data, and quantum computing is mainly applied to transform the extracted features for final classification. However, feature extraction is more computationally expensive compared to the final classification procedure. Thus, further studies on QML are still necessary.

However, existing quantum machines have limitations in terms of full fault tolerance and support only a few qubits, which constrains the practical applications of QML algorithms, and many algorithms can only be tested via small-scale simulations. Still, it is important to recognize the significance and necessity of investigating QML algorithms, as it is essential for the development of future practical applications when more advanced quantum devices become available.

This study seeks to investigate how to exploit the power of quantum computing to efficiently extract critical features from multispectral images using quantum circuits for land cover classification. In this article, we introduce two hybrid quantum-classical CNNs, which can comprehend multispectral images and perform multicategory classification accordingly. Specifically, our models exploit quantum computing for efficient feature extraction from multispectral images and classical computing for final classification to handle multicategory classification effectively.

To evaluate the validity of our proposed models on the land cover classification tasks, we conducted experiments aimed at classifying the local climate zone (LCZ)-based urban land covers. Note that the LCZ scheme was originally introduced for urban climate studies, such as [23] and [24]. Nevertheless, researchers have also explored the potential of LCZ for urban land cover classification, providing insights into human settlement for sustainable urbanization monitoring [25].

In our experiments, we trained our models using the LCZ42 [26] dataset through the TensorFlow Quantum (TFQ) platform [27]. The experimental results indicate that our models are superior regarding feature extraction and transferability compared with their classical counterparts.

The main contributions of this work include the following.

1) Two hybrid quantum-classical deep learning models for land cover classification are introduced, and they both can efficiently extract features from multispectral images using quantum computing and handle multicategory classification tasks.
2) We investigate and analyze the transferability and robustness of the two introduced models.
3) This study demonstrates proof of concept of implementing the CNN model in the quantum domain for land cover classification with multispectral images.

The rest of this article is organized as follows. Section II details the structures of our proposed models. Section III presents the experiments on the LCZ-based land cover classification tasks and discusses the experimental results. Section IV discusses the suitability and efficiency of our models based on the required quantum resources. Finally, Section V concludes this article.

II. Methodology

In this article, we introduce two hybrid deep learning models for land cover classification by analyzing Sentinel-2 images, namely the MCQI-based QC-CNN (**MQCNN**) model and the FRQI-based QC-CNN (**FQCNN**). As can be seen in Fig. 1, three key tasks are conducted sequentially in both models, namely *A) quantum image representation*, *B) feature extraction*, and *C) classification*. We will explain them individually for each model.

A. Quantum Image Representation

The two proposed models apply different image representation methods to encode multispectral images, and these methods both take advantage of quantum properties and amplitude embedding to represent images efficiently. To encode a M-spectral image with the size of $N \times N$, notated as $N \times N \times M$, the number of required qubits and basic quantum gates are different.

1) Image Encoding in MQCNN: The MCQI [28] method is adopted to encode images with multiple bands in the MQCNN model. To accurately represent this multispectral image, we need a quantum register qL with $2n$ qubits ($N = 2^n$) to encode the input image's spatial information, a register qM with m qubits ($2^m \geq M > 2^{m-1}$) to indicate the band information, and one register qC (single qubit) to represent the corresponding pixel value. The entanglement among qL, qM, and qC aims to match the information accurately. The following equation shows its implementation:

$$|I\rangle = \frac{1}{N \times \sqrt{M}} \sum_{i=0}^{M-1} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} |l_{x,y}\rangle |i\rangle |c^i_{x,y}\rangle \tag{1}$$

where $|l_{x,y}\rangle$ indicates the spatial information, $|i\rangle$ identifies the band information, and $|c^i_{x,y}\rangle$ embeds the pixel's value in the ith band.

To obtain the desired quantum state for this image, we prepare a quantum circuit constructed with $2n + m + 1$ qubits. Given the initial quantum state, $2n$ Hadamard gates are applied on the qL register to prepare all the needed location information. A series of quantum gates are applied on the qM register to obtain M basis states with the equivalent amplitudes ($1/\sqrt{M}$) for band identification. In the end, we use RY gates to rotate the qC with specific degrees around the Y-axis based on the states of the qL and qM that specify the location and the band information, respectively. The rotated qC can be represented by (2), in which $|0\rangle$ and $|1\rangle$ are the basis states and $\theta^i_{x,y}$ is the pixel value in the ith band mapped from [0,255] to $[0, \pi/2]$

$$|c^i_{x,y}\rangle = \cos\theta^i_{x,y}|0\rangle + \sin\theta^i_{x,y}|1\rangle. \tag{2}$$

Fig. 2(b) illustrates one circuit encoding a target image (2 × 2 × 3), in which qL (white) with two qubits refers to the spatial information, qM (red) with two qubits identifies the band information, and qC (gray) encodes the converted pixel values.

2) Image Encoding in FQCNN: The quantum image representation applied in the FQCNN model is based on the FRQI method [29] that was originally proposed to encode classical gray-scale images into quantum states. To represent a multispectral image using FRQI, we employ multiple quantum circuits rather than one, and each circuit encodes one spectral band information of the image. Thus, the quantum register for band identification (qM) in the MQCNN model is not necessary for the FQCNN model.

Specifically, the classical image with the size $N \times N \times M$ can be represented as $I = \{I_1, \ldots, I_M\}$, and each element indicates one band information with the size $N \times N$. An arbitrary band I_i can be encoded into a quantum state following (3), in which $|l_{x,y}\rangle$ indicates the location, and $|c^i_{x,y}\rangle$ embeds the pixel's value in the ith band, where $i = 1, 2, 3, \ldots M$. Thus, the quantum state of the multispectral image can be represented as $|I\rangle = \{|I_1\rangle, \ldots, |I_M\rangle\}$

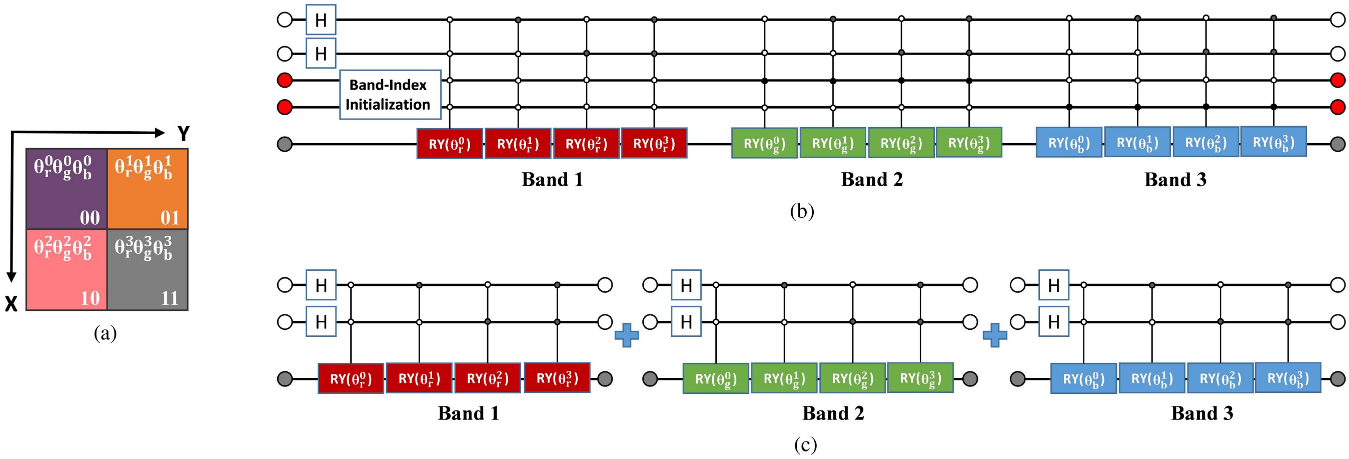
$$|I_i\rangle = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} |l_{x,y}\rangle |c^i_{x,y}\rangle. \tag{3}$$

To represent one band information of the given image (e.g., $|I_i\rangle$), a quantum circuit containing $2n + 1$ qubits is prepared. Given the initial quantum state, we apply $2n$ Hadamard gates on the qL register. Then controlled rotation gates are used to rotate the qC around the Y-axis based on the qL's state with specific degrees so as to encode the pixel values from the ith band and entangle them with the corresponding location. The rotated qC can be represented by (2).

Fig. 2(c) illustrates the circuits to encode the target image. Totally, three circuits are needed, and each circuit requires three qubits: two qubits for qL (white) for the spatial information and one qubit for qC (gray) for the converted pixels' values in each band.

B. Feature Extraction

The quantum circuit ansatz used for automatic feature extraction in our models is rooted in the QC-CNN model [30].


Y
X
θr0θg0θb0
00
θr1θg1θb1
01
θr2θg2θb2
10
θr3θg3θb3
11
H
H
Band-Index Initialization
RY(θr0) RY(θr1) RY(θr2) RY(θr3)
RY(θg0) RY(θg1) RY(θg2) RY(θg3)
RY(θb0) RY(θb1) RY(θb2) RY(θb3)
Band 1
Band 2
Band 3

(a)

(b)

(c)

Fig. 2. Circuit structure of the encoding layer in our models, illustrated using a classical RGB Image as an example: 1) the qubit color indicates the qubit register: white for qL; red for qM; gray for qC; 2) dot markers indicate the controlled state: white dots for $|0\rangle$ and black dots for $|1\rangle$; 3) H represents the Hadamard gate, and RY(θ) represents the RY gate with θ degrees; 4) the gate for the band-index initialization can rotate two qubits from qM so that the quantum state $(|00\rangle/\sqrt{3} + |01\rangle/\sqrt{3} + |10\rangle/\sqrt{3})$ is obtained for band identification; 5) (a) target classical RGB image with the size of 2×2, and θ indicates the pixel value converted from [0,255] to $[0, \pi/2]$; 6) (b) and (c) are the circuit examples of the encoding layer in the MQCNN model and the FQCNN model, respectively. (a) Target image. (b) Encoding layer of the MQCNN model. (c) Encoding layer of the FQCNN model.

As shown in Fig. 1, the circuit for feature extraction sequentially includes two types of layers as follows. *1) Quantum convolution layer* to extract features with quantum computing. *2) Measurement layer* to obtain generated feature values from the quantum states to the classical states.

1) Quantum Convolution Layer: The quantum convolution layer in our models conducts the convolutional operation for feature extraction. Both models independently extract the critical features from different bands. We set the kernel's size as 2×2 and the convolutional stride as 2 to reduce the dimension of the generated feature maps, so there is no pooling layer in the models.

In the quantum convolution layer of our models, we employ controlled U3 gates that can rotate a qubit with three Euler angles according to the controlling state, and their degrees will be optimized during the training process. As the kernel size is set to be 2×2, we require four U3 gates with trainable parameters, and each U3 gate represents one weight of the kernel. To distinguish the locations of these four weights, we add two controllers to the U3 gates. In cases where multiple kernels are used, the number of controllers required for the U3 gates will increase accordingly. Nevertheless, due to quantum parallelism, only four U3 gates are necessary to execute the convolution operation with one kernel over the entire input, regardless of its size.

Specifically, for the MQCNN model, Fig. 3(a) represents a quantum circuit example for a single quantum convolution layer with two filters (two kernels for each filter) to process an input image with the size of $4 \times 4 \times 2$. As depicted in Fig. 3(a), each feature map consists of the features extracted from two bands, and in the end, two generated feature maps with the size of $2 \times 2 \times 2$ are expected accordingly.

As for the FQCNN model, due to the adopted image representation method, one quantum circuit ansatz only extracts features from one band of the input image. Thus, to process a $4 \times 4 \times 2$-sized image, two circuits are needed for feature extraction. The example circuits are depicted in Fig. 3(b). In the end, two feature maps with the size of 2×2 for each band (two bands in total) will be generated.

2) Measurement Layer: This layer can obtain the extracted feature maps from the quantum states to the classical data following:

$$E(V) = \langle \varphi | V | \varphi \rangle \tag{4}$$

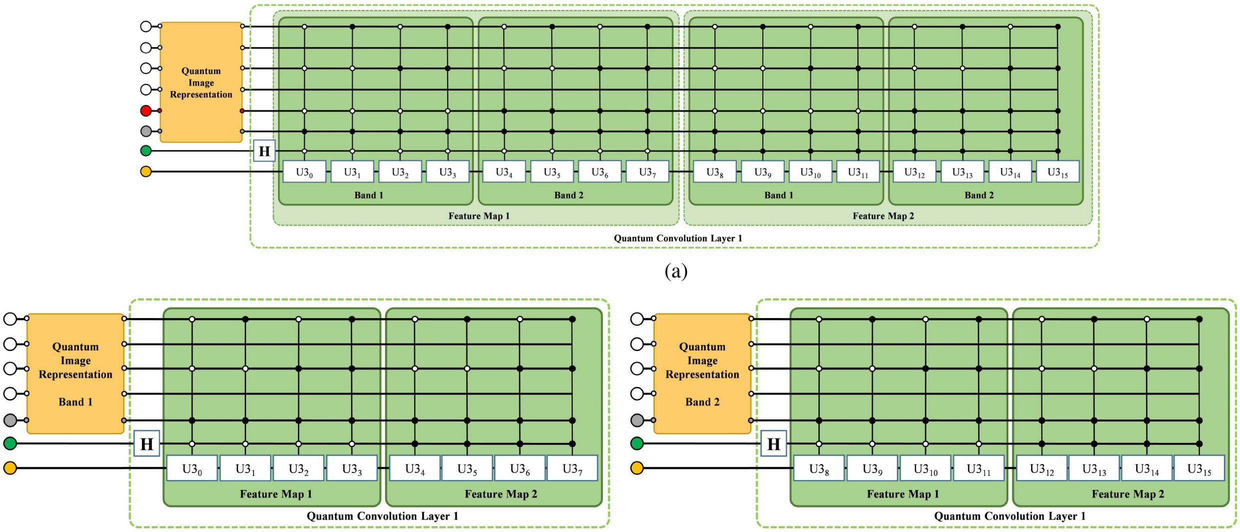
where V is the operator for measurement, and $|\varphi\rangle$ is the quantum state embedding generated feature maps. The obtained expectation value, $E(V)$, will be utilized as the extracted feature value.

To be more specific, we define a set of operators V and perform measurements on the state $|\varphi\rangle$ individually in the *X*-basis, *Y*-basis, and *Z*-basis. Each basis is formed by a set of states $|b\rangle$, and for each state $|b_i\rangle$ of interest, we can identify an operator V_i using $|b_i\rangle\langle b_i|$ and obtain the corresponding expectation value $E(V_i)$ as the ith feature using (4). Ultimately, we concatenate all the obtained expectation values and generate a feature vector for the final classification.

Fig. 4 illustrates a circuit example of the measurement layer for the MQCNN model, where five qubits are measured to obtain all the extracted features, including two qubits from qL for the spatial information of the feature maps, one qubit from qM for the band indication, one qubit from qK for the kernel information, and one qubit from qR for the value of the feature maps. Different from the MQCNN model which extracts all the features in one measurement layer, the FQCNN model applies multiple circuits for feature extraction, so several measurement layers are needed. Each layer will only obtain the high-level features from one band, so compared with the MQCNN model, only four qubits will be measured for each circuit since there is no qubit for band information in the FQCNN model.

C. Classification

For the final classification, with the classical feature vector from the measurement layer, a classical dense layer is utilized


Quantum Image Representation
H
$U3_0$ $U3_1$ $U3_2$ $U3_3$ $U3_4$ $U3_5$ $U3_6$ $U3_7$ $U3_8$ $U3_9$ $U3_{10}$ $U3_{11}$ $U3_{12}$ $U3_{13}$ $U3_{14}$ $U3_{15}$
Band 1
Band 2
Feature Map 1
Feature Map 2
Quantum Convolution Layer 1
Quantum Image Representation Band 1
Quantum Image Representation Band 2

(a)

(b)

Fig. 3. Circuit example of the quantum convolution layer for the proposed models: (a) and (b) are the circuit examples for the convolutional transformation on the image (4 × 4 × 2) with two kernels (2 × 2) for the MQCNN model and the FQCNN model, respectively; 1) the qubit color indicates the qubit register: white for qL; red for qM, gray for qC; green for qK; yellow for qR; 2) dot markers in the circuit indicate the controlled state: white dots for $|0\rangle$ and black dots for $|1\rangle$; 3) H represents the Hadamard gate, and U3 represents the U3 gate. (a) Circuit example of the quantum convolution layer in the MQCNN model. (b) Circuit example of the quantum convolution layer in the FQCNN model.

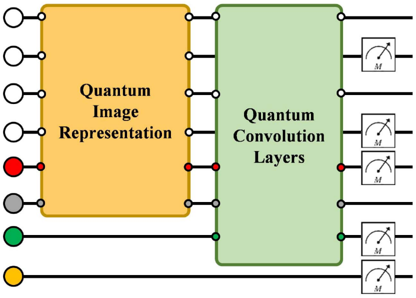


Fig. 4. Circuit example of the measurement layer for the MQCNN model. 1) White for qL; red for qM; gray for qC; green for qK; and yellow for qR.

for the final classification in both models. Each neuron encodes one feature value, and they are fully connected. The activation function (softmax) will be applied to output a probability distribution for multicategory classification tasks.

D. Training Procedure

The quantum convolution layers with trainable parameters in our models are PQCs, in which the gates are fixed, but their parameters can be optimized during the training process.

More precisely, as shown in Algorithm 1, the training process is composed of the following steps.

1) The training dataset can be represented as $\mathcal{D} = (\mathbf{x}, \mathbf{y})$, where $\mathbf{x}$ are the normalized images and $\mathbf{y}$ are the corresponding one-hot encoded labels.
2) For input images, the model outputs probability distributions $\tilde{\mathbf{y}} = f(\mathbf{x}, \theta)$, where θ denotes the trainable parameters in the models.

Algorithm 1: Training Procedure for Our Models.

```
Input: Training dataset $\mathcal{D}$, Learning rate $\alpha$, Number of
       epochs $E$
Output: Learned parameters $\theta$
Initialize $\theta$ randomly; Shuffle $\mathcal{D}$;
for each epoch in $E$ do
    for each batch $(\mathbf{x}, \mathbf{y})$ in $\mathcal{D}$ do
        Output the model's predictions $\tilde{\mathbf{y}} = f(\mathbf{x}, \theta)$;
            Represent $\mathbf{x}$ into quantum states;
            Generate quantum feature maps;
            Obtain classical feature vectors;
            Yield probability distributions $\tilde{\mathbf{y}}$;
        Compute the cross-entropy loss $L(\tilde{\mathbf{y}}, \mathbf{y})$;
        Compute the gradients $\nabla_\theta L(\tilde{\mathbf{y}}, \mathbf{y})$;
        Update the parameters $\theta \leftarrow \theta - \alpha \nabla_\theta L(\tilde{\mathbf{y}}, \mathbf{y})$;
```

3) The cross-entropy loss function is used to compare the output against the label $L(\tilde{\mathbf{y}}, \mathbf{y})$.
4) The trainable parameters θ will be optimized with the Adam algorithm [31].

We repeat these steps until θ is optimized.

For inference on the images $\mathcal{I}$, the learned parameters will be assigned to the gates in the quantum convolution layers and weight variables in the dense layers of our models. The output $f(\mathcal{I}, \theta)$ indicates the predicted classes of input images $\mathcal{I}$.

III. Experiment

To evaluate and compare the performance of the MQCNN model and the FQCNN model on the LCZ-based land cover

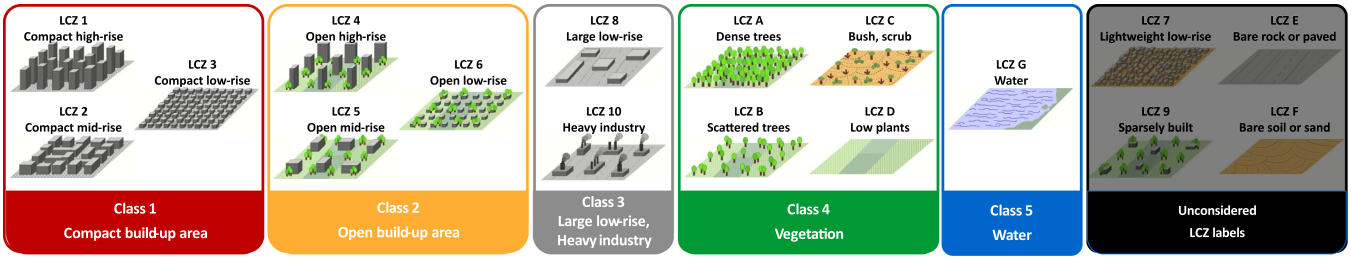


Fig. 5. Derived land cover classes and their corresponding LCZ labels, and the visualization of the LCZ concept from WUDAPT [32].

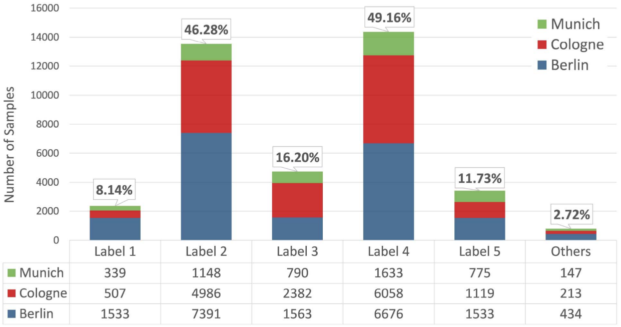


Fig. 6. Distribution of the land cover classes from the three target cities.

classification tasks, we conducted experiments with different models on the LCZ42 [26] dataset.

A. Data Preparation

The LCZ42 dataset consists of around half a million Sentinel-1 and Sentinel-2 image patches in 42 urban areas across the globe with annotated 17 LCZ labels. In the dataset, the Sentinel-2 image patches (32×32) contain ten bands, including bands B2, B3, B4, B8 with 10 m ground sampling distance (GSD), bands B5, B6, B7, B8a, B11, B12 with 10 m GSD upsampled from 20 m.

Due to the high requirement on computation power for quantum simulation, downsampling the training images is necessary. Thus, in our experiments, the patch size was compressed from 32×32 to 8×8, and only four bands were selected, including B2 (blue), B3 (green), B4 (red), and B8 (nearinfrared) with 10 m GSD. Moreover, the labeled patches from three German cities, Berlin, Munich, and Cologne, were used for the experiments rather than all 42 cities.

In addition, to simplify the experiments, we regrouped 13 LCZ labels into five semantic classes instead of considering all 17 LCZ labels following the land cover classification scheme [25]. Fig. 5 shows the semantic classes with the corresponding LCZ labels. Note that four LCZ labels (7, 9, E, F) were not considered in our experiments. The reason is that, as shown in Fig. 6, only around 2.72% labeled patches from these three cities belong to these four LCZ labels. Therefore, we only concentrate on the five dominant semantic classes shown in Fig. 5 in our experiments.

We randomly selected 10 000 labeled patches in these three cities from these five classes for the experiments. Each class has 2000 samples to build a balanced dataset. The distribution of the experimental data is shown in Table I.

TABLE I
DISTRIBUTION OF THE EXPERIMENTAL DATA FOR CLASSIFICATION PERFORMANCE EVALUATION

	Class 1	Class 2	Class 3	Class 4	Class 5
Munich	289	179	340	209	463
Cologne	424	735	1021	859	660
Berlin	1287	1086	639	932	877

B. Experiment Settings

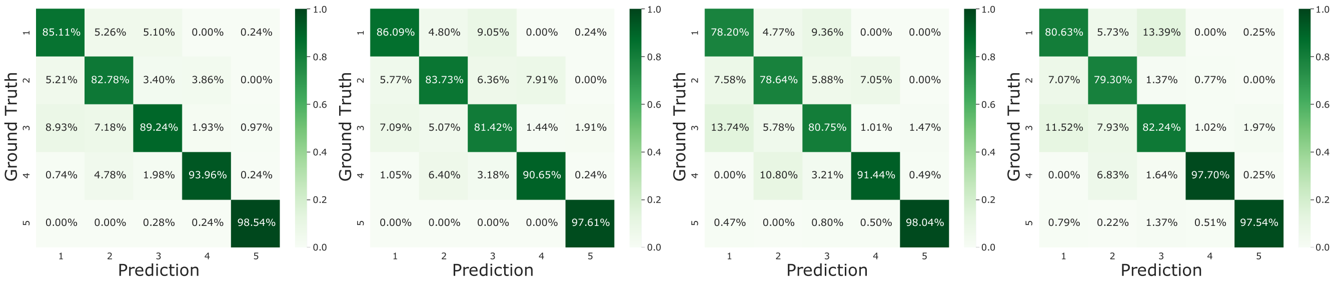
In the experiments, we trained our quantum models with two quantum convolution layers, and each layer applied two filters. Besides, the epoch number was set as 200, and the batch size was given as 50. The cross-entropy loss function and the Adam optimizer [31] with the learning rate 0.003 were applied.

Due to the goal of the experiments being to verify the models' validity, we adopted a noiseless simulator from the TFQ platform in our experiments. This simulator outputs the analytic results, and it runs faster than other simulators. Regarding the backpropagation procedure, we applied the Adjoint differentiator from TFQ that can compute the gradients faster than others, and it is compatible with the adopted simulator.

To fully validate our models, we designed a five-fold cross-validation experimental setup. Thus, we split the data evenly into five sets, and accordingly, we conducted five different experiments. In each experiment, one set of data was used for accuracy assessment, whereas the other four sets were used for training models. In the end, the average results over all the conducted experiments were summarized for further comparison and discussion.

C. Classification Performance Comparison

We compared the performance of our models with one classical CNN model and one quantum framework (QCNN [18]). The QCNN model exploits the classical CNN as a feature extractor so it can handle images with four bands, and it utilizes a classical dense layer at the end, which supports multicategory classification. For a fair comparison, we adopted the QCNN model with a similar number of trainable parameters as our models as the competitor. Particularly, the CNN part of the QCNN model comprises two convolutional layers: one with



(a) (b) (c) (d)

Fig. 7. Confusion matrices for the models with the best performance in the experiments on the test dataset. (a) MQCNN with the OA 0.90. (b) FQCNN with the overall accuracy 0.88. (c) CNN with the overall accuracy 0.85. (d) QCNN with the overall accuracy 0.87; Classes 1–5 are compact built-up area, open built-up area, large low-rise and heavy industry, vegetation, and water, respectively. (a) MQCNN. (b) FQCNN. (c) CNN. (d) QCNN.

TABLE II
Comparison of Multicategory Classification Performance of Different Models

Model	Kernel	Parameter	Training Accuracy	Test Accuracy
MQCNN	16	677	0.891 ± 0.002	0.884 ± 0.009
FQCNN	16	677	0.874 ± 0.003	0.872 ± 0.006
QCNN[18]	64	749	0.904 ± 0.005	0.866 ± 0.007
CNN	56	735	0.869 ± 0.006	0.839 ± 0.010

four kernels and the other with 12 kernels. Each kernel size is set to 2 × 2, with a convolutional stride of 2 as our models in the experiment. In addition, a dense layer within the CNN produces eight features. These extracted features serve as input for the quantum component, which in turn generates 16 feature values to feed into the subsequent classical dense layer for final prediction.

Regarding the classical CNN model, to assess the effectiveness of our models for feature extraction, we selected a CNN with two convolution layers, each comprising two and 24 kernels, respectively, as the competitor. The kernel size was set to 2 × 2, and the convolutional stride to 2, consistent with our models. These convolution layers have a similar number of parameters as the quantum layers in our models, and the same number of the features (96 features) would be extracted for the final prediction. Thus, we could analyze the validity of the quantum circuits for feature extraction by comparing the classification accuracy without considering the effects of the classical dense layers.

Table II shows the averaged classification accuracy achieved by different models. As shown in the table, our models outperform the classical CNN model and the QCNN model in terms of classification accuracy. For the performance difference between our models, as indicated in the table, the MQCNN model is superior to the FQCNN model, but the improvement is moderate (from 0.872 to 0.884).

Fig. 7 visualizes the confusion matrix for each model with the best performance from the experiments. From the figure, we observe that our models can identify the compact built-up area and the open built-up area more accurately than the competitors. When comparing our models, according to the confusion matrices [see Fig. 7(a) and 7(b)], the FQCNN can reach slightly higher accuracy for the compact built-up area and the open built-up area classification, whereas the MQCNN model performs better in general.

Fig. 8 illustrates the land cover maps for Berlin, Munich, and Cologne, and they were produced by the trained models with the best performance via applying the sliding window algorithm on the cloud-removed Sentinel-2 data of these three cities from Google Earth Engine [33]. In general, the maps produced by these models basically all display reasonable urban structures.

According to the experimental results, we could conclude that our proposed models have comparable performance in terms of classification accuracy (0.872 for FQCNN and 0.884 for MQCNN). They both can extract critical features more effectively than the classical CNN model and have better performance than the QCNN model.

In addition, besides the QCNN model, we also considered other quantum frameworks for comparison, for example, the HQconv [21] and the QNN4EO model [17]. However, the HQconv ansatz is designed to process RGB images rather than images with an arbitrary number of bands. The QNN4EO model is a binary classifier that can perform multicategory classification by applying the one-versus-one strategy. For a fair comparison, we carried out another binary classification experiment with RGB images for classification performance evaluation.

For this binary classification experiment, we used the patches from *Class 1* and *Class 2* in Table I that are the compact built-up area and the open built-up area, respectively. To meet the requirement of all the experimented models and compare the performance fairly, we only kept the RGB bands of the target patches for the experiment. We also applied the five-fold cross-validation procedure for this experiment, and the averaged results were compared and discussed.

Table III shows the binary classification experiment results. As shown in the table, our proposed models are superior regarding classification accuracy compared with the tested quantum frameworks.

Moreover, note that the QCNN model and the QNN4EO model rely on the classical CNN models for feature extraction, and the quantum circuits aim to conduct the feature transformation or final classification. In comparison, our models use quantum circuits to extract features, which can speed up the convolutional operation by simultaneously transforming all

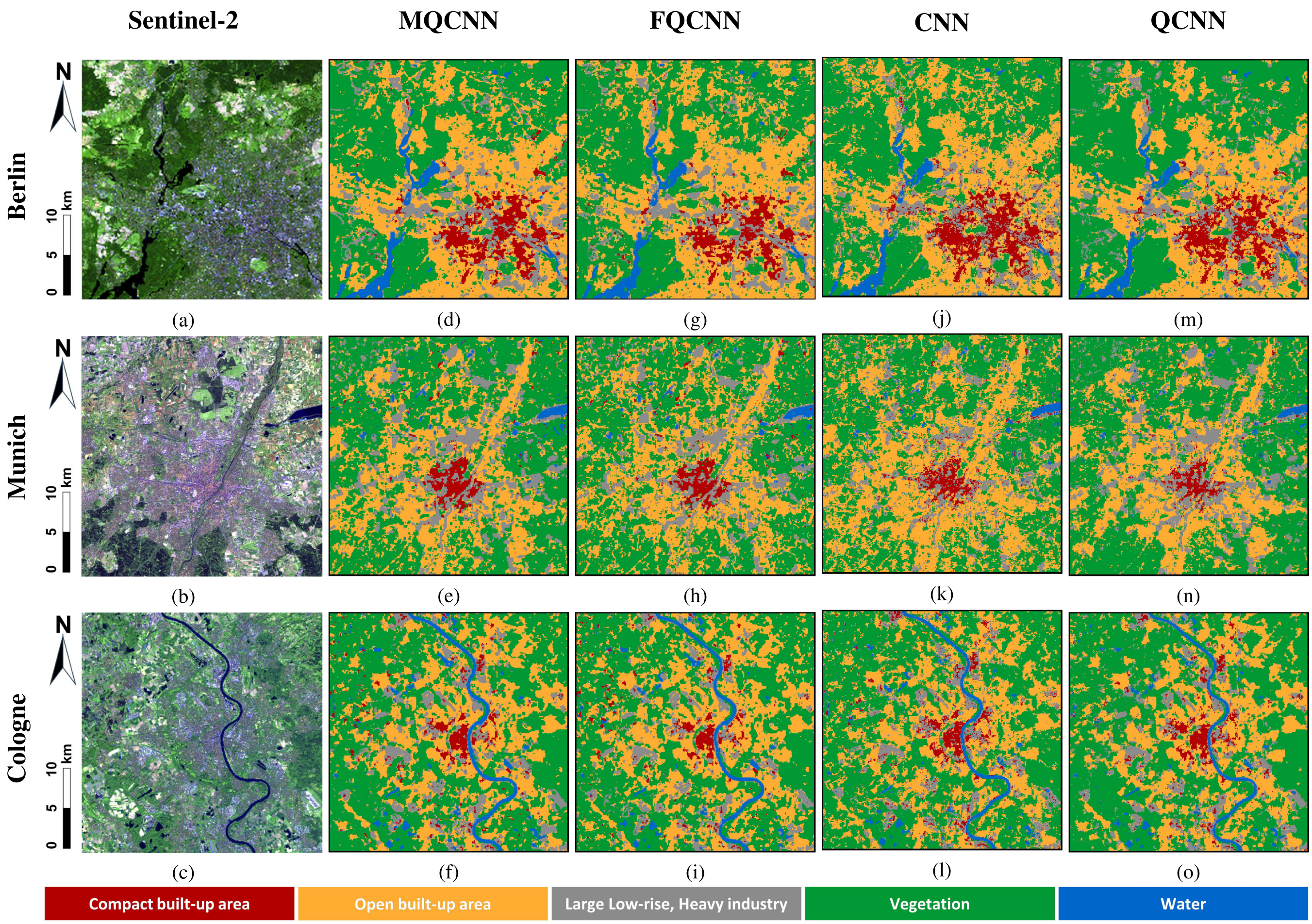


Fig. 8. Land cover maps produced by the trained models with the best performance for Berlin, Munich, and Cologne, covering about 30 × 30 km^2 and the GSD of each map is 10 m: (a–c) for the false-colored cloud removed Sentinel-2 Google Earth images of the three cities; (d–f) land cover maps produced by the trained MQCNN model; (g–i) land cover maps produced by the trained FQCNN model; (j–l) land cover maps produced by the trained CNN model; (p–r) land cover maps produced by the trained QCNN mode.

TABLE III
Comparison of Binary Classification Performance of Different Models

Model	Parameter	Training Accuracy	Test Accuracy
MQCNN	217	0.880 ± 0.005	0.879 ± 0.011
FQCNN	217	0.872 ± 0.002	0.874 ± 0.013
QNN4EO [17]	253	0.879 ± 0.009	0.871 ± 0.011
QCNN[18]	334	0.895 ± 0.011	0.870 ± 0.010
HQconv [21]	745	0.860 ± 0.007	0.817 ± 0.020

desired quantum states. In addition, the QNN4EO model is a binary classifier. To handle g-category classification tasks, training $g \times (g-1)/2$ classifiers is necessary when using the one-versus-one strategy, which will compromise the overall efficiency. Besides, the HQconv model uses a quantum circuit as the kernel and the sliding window mechanism for feature extraction. On the contrary, our models can perform the convolutional operation over the image without applying the sliding window mechanism because of quantum parallelism. Thus, our models can extract features more efficiently than the HQconv model.

D. Transferability and Consistency Evaluation

For land cover classification, the cities where we would like to apply the trained classifier can be significantly different from the source dataset used for training. The model with good transferability can manage the discrepancy across cities and classify the land cover types of the target cities accurately.

To evaluate the transferability of our models, we carried out three additional experiments with different datasets following the cross-validation method. In each experiment, we used two cities as the source cities to train the model and the last city as the target city to assess the model's performance. Therefore, no samples from the target city will be used to train the models. We repeated each experiment three times, and the average results over the repeated experiments were compared and discussed to evaluate the transferability of the models.

Regarding the data for each experiment, we randomly reselected 800 patches for each semantic class from the source cities in the LCZ42 dataset and built a balanced training dataset. As for the test dataset, the numbers of labeled patches from the target cities are not equivalent in quantity (Berlin: 18 696,

TABLE IV
Number of Land Cover Samples for Transferability Evaluation

			Label 1	Label 2	Label 3	Label 4	Label 5
Dataset 1	Training data (4000)	Cologne Munich	476 324	655 145	616 184	634 166	496 304
	Test data (4685)	Berlin	364	1886	377	1678	380
Dataset 2	Training data (4000)	Berlin Cologne	616 184	491 309	339 461	404 396	486 314
	Test data (4685)	Munich	339	1148	790	1633	775
Dataset 3	Training data (4000)	Berlin Munich	652 148	697 103	521 279	647 153	511 289
	Test data (4685)	Cologne	178	1509	771	1882	345

TABLE V
Comparison of Classification Accuracy for Each Dataset in Transferability Evaluation

Dataset	Model	Kernel	Parameter	Training	Imbalanced Test			Balanced Test	
				OA	OA	AA	WA	OA	Kappa
1	MQCNN	16	677	0.873 ± 0.006	0.893 ± 0.006	0.867 ± 0.006	0.893 ± 0.006	0.867 ± 0.011	0.834 ± 0.014
	FQCNN	16	677	0.863 ± 0.006	0.833 ± 0.006	0.833 ± 0.006	0.833 ± 0.006	0.832 ± 0.012	0.790 ± 0.015
	QCNN[18]	64	749	0.920 ± 0.000	0.867 ± 0.021	0.827 ± 0.012	0.870 ± 0.017	0.833 ± 0.008	0.791 ± 0.010
	CNN	56	735	0.857 ± 0.006	0.777 ± 0.015	0.793 ± 0.012	0.777 ± 0.015	0.801 ± 0.007	0.752 ± 0.009
2	MQCNN	16	677	0.897 ± 0.006	0.770 ± 0.010	0.800 ± 0.000	0.770 ± 0.010	0.804 ± 0.004	0.755 ± 0.005
	FQCNN	16	677	0.880 ± 0.000	0.783 ± 0.012	0.803 ± 0.006	0.780 ± 0.010	0.806 ± 0.011	0.757 ± 0.014
	QCNN[18]	64	749	0.923 ± 0.015	0.777 ± 0.038	0.760 ± 0.036	0.777 ± 0.038	0.752 ± 0.037	0.690 ± 0.047
	CNN	56	735	0.887 ± 0.006	0.707 ± 0.015	0.700 ± 0.026	0.700 ± 0.017	0.693 ± 0.029	0.616 ± 0.037
3	MQCNN	16	677	0.910 ± 0.010	0.820 ± 0.010	0.750 ± 0.020	0.837 ± 0.015	0.818 ± 0.014	0.773 ± 0.018
	FQCNN	16	677	0.900 ± 0.000	0.807 ± 0.006	0.743 ± 0.006	0.827 ± 0.006	0.814 ± 0.007	0.767 ± 0.009
	QCNN[18]	64	749	0.963 ± 0.006	0.750 ± 0.017	0.667 ± 0.025	0.777 ± 0.015	0.722 ± 0.027	0.652 ± 0.034
	CNN	56	735	0.923 ± 0.006	0.750 ± 0.010	0.660 ± 0.017	0.777 ± 0.006	0.719 ± 0.020	0.649 ± 0.025

Munich: 4685, Cologne: 15 052). To eliminate its potential impact on transferability evaluation, we downsampled the data for Berlin and Cologne and randomly selected 4685 patches for each of these cities. Note that we maintained the distribution over the semantic classes in each city since the land cover types in the real world are not comparable in quantity. Table IV shows the distribution of the data for the evaluation.

In our experiments, we evaluated the models' performance on the target cities (test datasets) with two accuracy assessment strategies, namely *imbalanced test* and *balanced test*. For the first strategy, we assess the classification performance of the models on all the labeled patches of the target cities. Since the test datasets are imbalanced, as shown in Table IV, we compared overall accuracy (OA), average accuracy (AA), and weighted accuracy (WA) for performance evaluation.

To overcome the potential problems due to the imbalanced test datasets, we also considered the second strategy, where we conducted 20 assessments for each repeated experiment. For each assessment, we randomly selected the same number of samples for each category in the target city. To obtain more samples for the assessment, the number of the selected samples for each category is the same as the number of samples in the category with the least samples (i.e., the numbers of the selected samples for each category for Berlin, Munich, and Cologne are 364, 339, and 178, respectively). In this way, 20 balanced subtest datasets were built for one experiment. The averaged OA and kappa coefficient over all the experiments would be compared and discussed.

Table V shows the results for each experiment, and Table VI shows the averaged results over all experiments for the transferability evaluation. To evaluate the transferability of our models, the CNN model with 56 kernels and the QCNN model worked as the baseline models. As indicated in Table VI, our proposed models have higher transferability than baseline models no matter in the imbalanced test setting or the balanced test setting. Between the MQCNN and FQCNN models, the former is superior in terms of transferability.

Fig. 9 shows the land cover maps and the agreement maps of Berlin, Munich, and Cologne. For each target city, the land cover maps were produced by our models with the best performance in the transferability evaluation experiments, so the applied models were trained without samples from the target city. In general, these land cover maps display reasonable urban structures, indicating the experimented models are able to handle the difference between the source cities and target cities at certain levels.

Regarding the agreement maps shown in Fig. 9, they illustrate the difference between the land cover maps generated by the models trained with and without the samples from the target city, which imply the impact of the samples in the target cities on classification inference results. As shown in the figure, our models can have more consistent predictions than the CNN model and the QCNN model (the averaged agreement rate: 88.46% for MQCNN; 88.67% for FQCNN; 79.36% for CNN; 81.01% for QCNN). Thus, training without samples from the target city has less influence on the classification results for our

TABLE VI
COMPARISON OF CLASSIFICATION ACCURACY FOR TRANSFERABILITY EVALUATION

Model	Kernel	Parameter	Training OA	Imbalanced Test			Balanced Test	
				OA	AA	WA	OA	Kappa
MQCNN	16	677	0.893	0.828	0.806	0.833	0.830	0.787
FQCNN	16	677	0.881	0.808	0.793	0.813	0.817	0.771
QCNN[18]	64	749	0.936	0.798	0.751	0.808	0.769	0.711
CNN	56	735	0.889	0.744	0.718	0.751	0.738	0.672

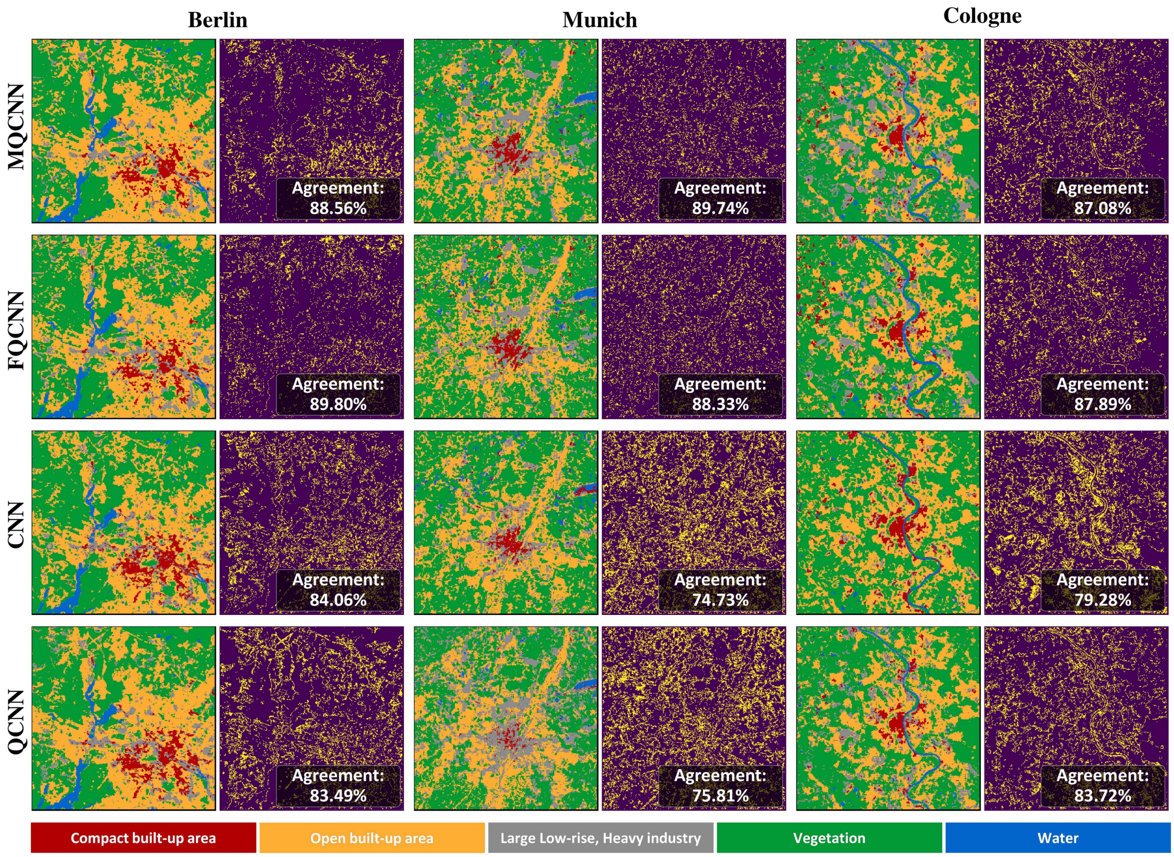


Fig. 9. Land cover maps of the target cities produced by the models trained without samples from the target cities and agreement maps of the target cities generated based on the models trained with and without samples from the target cities (purple pixels for the consistent classification results and yellow for the inconsistent results). The map covers 30 × 30 km^2 and the GSD of each map is 10 m.

models than the competitors, implying that our models have higher consistency.

In conclusion, our models can extract important features more effectively than the classical CNN model and have superior classification performance than the experimented quantum frameworks. In addition, the experimental results indicate that our models have higher transferability and consistency than the competitors, implying that they have a greater capacity to cope with the discrepancy between the cities for land cover classification.

As for the comparison between our models, the MQCNN model has moderate advantages regarding classification accuracy and transferability than the FQCNN model. However, note that these two models require different quantum resources, which will be discussed in Section IV in detail. Besides, the quantum simulators used in the experiments are noiseless, so the quantum models' performance can be compromised when adopting noisy quantum devices.

IV. DISCUSSION

The quantum resources required for classification are a crucial factor in evaluating and comparing our models. This criterion not only reflects the suitability of the proposed models for different EO applications but also indicates the efficiency of our models.

A. Analysis of the Suitability of Our Models for EO Applications on Quantum Computers

According to our experimental results, the MQCNN model can have a better classification performance and transferability than the FQCNN model. However, note that the performance improvement is moderate.

Besides, as introduced in Section II, the MQCNN model requires extra qubits to indicate the band information of the input image, so it requires more qubits and more complex quantum gates for image encoding and convolutional operation compared to the FQCNN model. The computational resources needed for the quantum system simulation generally grow exponentially with the size of the system instead of linearly [34]. Thus, training the MQCNN model requires more time than the FQCNN model when analyzing the same input images.

Due to the difference between the structures of the MQCNN model and the FQCNN model, they are suitable for different classification tasks, especially in the noisy intermediate-scale quantum (NISQ) era, since the number of qubits in quantum machines is constrained.

As mentioned before, we experimented with our models on the same input without considering the difference in the number of qubits between these two models. For a quantum system with a fixed number of qubits, the MQCNN model requires some qubits to encode the band information, so the number of qubits for spatial information is limited, which results in a decrease in the size of the input image. Thus, less spatial information will be considered for classification. The performance of the MQCNN model will be jeopardized due to the smaller receptive field for feature extraction. When analyzing the EO data with more bands, the disadvantage of the MQCNN model could become more obvious. Differently, the FQCNN model applies multiple circuits to analyze multispectral images, so the increase of the band information will not influence the number of qubits for spatial information encoding of the input image.

However, because the MQCNN model only utilizes one quantum circuit for classification, all the information is encoded and transformed in the same circuit. It provides the potential to achieve a more complex transformation between different bands of the input image to extract more sophisticated features for classification. It is beneficial for classification performance, but it might be challenging for the FQCNN model without changing its structure.

Ultimately, the selection of these two models for EO applications should depend on the requirements of the tasks and the capacity of the available quantum computers. Given a quantum computer that can support both models for encoding and analyzing the EO data, the MQCNN model might be preferred due to its superior classification performance. When the qubit number is limited such that the MQCNN model has to reduce the size of the input image, the FQCNN model could be more compatible since the input information will not be compromised because of the constraints of the quantum machine. When desiring feature extraction across bands of the input image, the MQCNN model is more suitable, but it requires further investigation.

B. Analysis of the Efficiency of Our Models

To assess the efficiency of our models for classification, we evaluate them from two different aspects: *quantum gates* and *qubits*, reflecting the required quantum resources, which can affect the speed of the classification algorithms. For comparison, we concentrate on the convolution operation as it is the crucial component for feature extraction. Specifically, we consider an example of the convolution layer, involving K kernels of size 2×2 and a convolution stride of 2, applied to an M-band image of size $N \times N$.

1) Quantum Gates: A quantum circuit consists of quantum gates, with each gate representing a specific operation, so the number of gates directly correlates with the number of operations required for computation. It has been widely used for efficiency analysis and comparison.

Based on the information provided in Section II, the convolution layer in our models requires $4KM$ operations since $4KM$ U3 gates are involved in the circuits, regardless of the input's spatial size. However, the U3 gates for the MQCNN model are more complex than those in the FQCNN model, since each former gate has $2 + \log(KM)$ controllers, while each latter gate has $2 + \log(K)$ controllers. In contrast, a classical convolution layer requires MKN^2 operations, which increase quadratically with the input's spatial size. Therefore, our models could offer higher computational efficiency for feature extraction compared to classical counterparts, particularly when processing large images.

However, note that when considering the encoding layer and measure layer, the overall efficiency of our models will be compromised, but how to encode images more efficiently is beyond the scope of this study.

2) Quantum Qubits: The number of qubits is also an important factor in evaluating the efficiency of a quantum algorithm, especially in the NISQ era. With the usage of amplitude encoding, our models require relatively fewer qubits compared to those using computational encoding for accurate information embedding and transformation. Specifically, the FQCNN model needs $\log(N^2 + K) + 2$ qubits, and the MQCNN model needs $\log(N^2 + K + M) + 2$ qubits. Thus, the qubit requirements for both models increase logarithmically with the size of the input and the number of applied kernels.

V. Conclusion and Future Work

In this work, two quantum-classical CNNs, MQCNN, and FQCNN, are introduced and compared for land cover classification tasks. They both can use quantum computing to extract critical features from images with multiple bands for multicategory classification purposes. Furthermore, our efficiency analysis indicates our models could extract features more efficiently with a small number of qubits due to quantum parallelism and amplitude encoding. The numerical experimental results indicate the advantages of our models for effective feature extraction compared to their classical counterparts. In addition, they both have higher transferability and consistency than the CNN model and other quantum frameworks. It is important to note that due to computational constraints in quantum simulation, we downsampled the images to 8×8 for experiments, which may not fully reflect practical scenarios in the EO domain. Nevertheless, the experiments demonstrate a proof of concept of using quantum computing for land cover classification and verify the validity of our proposed models.

Toward the comparison between our models, the MQCNN model can outperform the FQCNN model in terms of classification performance and transferability without considering the limit of quantum computers. When the quantum resources are not adequate for the MQCNN model to encode and analyze the original EO data, the FQCNN might be able to extract features using all the available information from the EO data for more accurate classification.

Regardless, future research could continue to explore the following directions.

1) Exploring how to encode EO data into quantum states more efficiently.
2) Investigating quantum circuits that can extract important features across different bands.
3) Studying how to leverage the power of quantum computing for hyperspectral remote sensing data analysis and classification.

References

[1] Z. Xiong, F. Zhang, Y. Wang, Y. Shi, and X. X. Zhu, "EarthNets: Empowering AI in Earth observation," 2022, *arXiv:2210.04936*.

[2] M. Belgiu and L. Drăguţ, "Random forest in remote sensing: A review of applications and future directions," *ISPRS J. Photogrammetry Remote Sens.*, vol. 114, pp. 24–31, 2016.

[3] G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: A review," *ISPRS J. Photogrammetry Remote Sens.*, vol. 66, no. 3, pp. 247–259, 2011.

[4] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. A. Johnson, "Deep learning in remote sensing applications: A meta-analysis and review," *ISPRS J. Photogrammetry Remote Sens.*, vol. 152, pp. 166–177, 2019.

[5] Y. Bouslihim, M. H. Kharrou, A. Miftah, T. Attou, L. Bouchaou, and A. Chehbouni, "Comparing pan-sharpened Landsat-9 and Sentinel-2 for land-use classification using machine learning classifiers," *J. Geovisualization Spatial Anal.*, vol. 6, no. 2, 2022, Art. no. 35.

[6] S. Pan et al., "Land-cover classification of multispectral LiDAR data using CNN with optimized hyper-parameters," *ISPRS J. Photogrammetry Remote Sens.*, vol. 166, pp. 241–254, 2020.

[7] P. Dou and C. Zeng, "Hyperspectral image classification using feature relations map learning," *Remote Sens.*, vol. 12, no. 18, 2020, Art. no. 2956.

[8] H. Chen et al., "A landslide extraction method of channel attention mechanism U-net network based on Sentinel-2 A remote sensing images," *Int. J. Digit. Earth*, vol. 16, no. 1, pp. 552–577, 2023.

[9] S. Yao et al., "A convLSTM neural network model for spatiotemporal prediction of mining area surface deformation based on SBAS-InSAR monitoring data," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art no. 5201722.

[10] Y. He et al., "Time-series analysis and prediction of surface deformation in the Jinchuan mining area, Gansu Province, by using InSAR and CNN–PhLSTM network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 6732–6751, Aug. 2022.

[11] P. Dou, H. Shen, Z. Li, X. Guan, and W. Huang, "Remote sensing image classification using deep–shallow learning," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 3070–3083, Mar. 2021.

[12] G. Cheng, X. Xie, J. Han, L. Guo, and G.-S. Xia, "Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 3735–3756, Jun. 2020.

[13] D. Ristè et al., "Demonstration of quantum advantage in machine learning," *npj Quantum Inf.*, vol. 3, no. 1, 2017, Art. no. 16.

[14] G. Cavallaro, D. Willsch, M. Willsch, K. Michielsen, and M. Riedel, "Approaching remote sensing image classification with ensembles of support vector machines on the D-wave quantum annealer," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2020, pp. 1973–1976.

[15] S. Otgonbaatar and M. Datcu, "A quantum annealer for subset feature selection and the classification of hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 7057–7065, Jul. 2021.

[16] P. Gawron and S. Lewiński, "Multi-spectral image classification with quantum neural network," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2020, pp. 3513–3516.

[17] D. A. Zaidenberg, A. Sebastianelli, D. Spiller, B. Le Saux, and S. L. Ullo, "Advantages and bottlenecks of quantum machine learning for remote sensing," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2021, pp. 5680–5683.

[18] A. Sebastianelli, D. A. Zaidenberg, D. Spiller, B. Le Saux, and S. L. Ullo, "On circuit-based hybrid quantum neural networks for remote sensing imagery classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 565–580, Dec. 2021.

[19] S. Otgonbaatar and M. Datcu, "Classification of remote sensing images with parametrized quantum gates," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, Sep. 2021.

[20] S. Abdel-Khalek, M. Algarni, R. F. Mansour, D. Gupta, and M. Ilayaraja, "Quantum neural network-based multilabel image classification in high-resolution unmanned aerial vehicle imagery," *Soft Comput.*, pp. 1–12, 2021.

[21] Y. Jing et al., "RGB image classification with quantum convolutional ansatz," *Quantum Inf. Process.*, vol. 21, no. 3, 2022, Art. no. 101.

[22] M. K. Gupta, M. Beseda, and P. Gawron, "How quantum computing-friendly multispectral data can be," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2022, pp. 4153–4156.

[23] L. Zhou, B. Yuan, F. Hu, C. Wei, X. Dang, and D. Sun, "Understanding the effects of 2D/3D urban morphology on land surface temperature based on local climate zones," *Building Environ.*, vol. 208, 2022, Art. no. 108578.

[24] B. Yuan, L. Zhou, F. Hu, and Q. Zhang, "Diurnal dynamics of heat exposure in Xi'an: A perspective from local climate zone," *Building Environ.*, vol. 222, 2022, Art. no. 109400.

[25] C. Qiu, L. Mou, M. Schmitt, and X. X. Zhu, "Local climate zone-based urban land cover classification from multi-seasonal Sentinel-2 images with a recurrent residual network," *ISPRS J. Photogrammetry Remote Sens.*, vol. 154, pp. 151–162, 2019.

[26] X. X. Zhu et al., "So2Sat LCZ42: A. benchmark data set for the classification of global local climate zones [Software and Data Sets]," *IEEE Geosci. Remote Sens. Mag.*, vol. 8, no. 3, pp. 76–89, Sep. 2020.

[27] M. Broughton et al., "TensorFlow quantum: A. software framework for quantum machine learning," 2020, *arXiv:2003.02989*.

[28] B. Sun, A. Iliyasu, F. Yan, F. Dong, and K. Hirota, "An RGB multi-channel representation for images on quantum computers," *J. Adv. Comput. Intell. Intell. Inform*, vol. 17, no. 3, pp. 404–417, 2013.

[29] P. Q. Le, F. Dong, and K. Hirota, "A flexible representation of quantum images for polynomial preparation, image compression, and processing operations," *Quantum Inf. Process.*, vol. 10, pp. 63–84, 2011.

[30] F. Fan, Y. Shi, T. Guggemos, and X. X. Zhu, "Hybrid quantum-classical convolutional neural network model for image classification," *IEEE Trans. neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2023.3312170.

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[32] I. D. Stewart and T. R. Oke, "Local climate zones for urban temperature studies," *Bull. Amer. Meteorological Soc.*, vol. 93, no. 12, pp. 1879–1900, 2012.

[33] N. Gorelick, M. Hancher, M. Dixon, S. Ilyushchenko, D. Thau, and R. Moore, "Google Earth engine: Planetary-scale geospatial analysis for everyone," *Remote Sens. Environ.*, vol. 202, pp. 18–27, 2017.

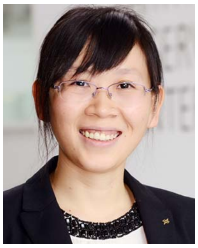
[34] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge, U.K.: Cambridge Univ. Press, 2010.

Fan Fan received the bachelor of design degree in industrial design from Xi'an Jiaotong University, Xi'an, China, in 2016, and the master's degree in human–computer interaction from Bauhaus University Weimar, Weimar, Germany, in 2020. He is currently working toward the Ph.D. degree under the Chair of Data Science in Earth observation with Technical University of Munich, Munich, Germany.

Since 2020, he has been a Research Assistant with the Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Weßling, Germany. His research interests include quantum algorithms in EO data analysis, including quantum machine learning, quantum optimization, etc.

Yilei Shi (Member, IEEE) received the Dipl.-Ing. degrees in mechanical engineering and signal processing from the Technical University of Munich (TUM), Munich, Germany, in 2010 and 2019, respectively.

He is a Senior Scientist with the Chair of Remote Sensing Technology, TUM. His research interests include fast solver and parallel computing for large-scale problems, high-performance computing and computational intelligence, advanced methods on synthetic aperture radar (SAR) and InSAR processing, machine learning, and deep learning for a variety of data sources, such as SAR, optical images, and medical images, and partial differential equation-related numerical modeling and computing.



Xiao Xiang Zhu (Fellow, IEEE) received the M.Sc., Dr.-Ing. and Habilitation degrees in signal processing from the Technical University of Munich (TUM), Munich, Germany, in 2008, 2011, and 2013, respectively.

She is the Chair Professor for data science in Earth observation with TUM and was the founding Head of the Department “EO Data Science,” Remote Sensing Technology Institute, German Aerospace Center, Cologne, Germany. She is a Visiting AI Professor with ESA’s Phi-lab, Frascati, Italy. She was a Guest Scientist or Visiting Professor with Italian National Research Council (CNR-IREA), Naples, Italy; Fudan University, Shanghai, China; the University of Tokyo, Tokyo, Japan; and the University of California, Los Angeles, Los Angeles, CA, USA, in 2009, 2014, 2015, and 2016, respectively. Since 2020, she has been the PI and Director of the International Future AI Lab “AI4EO—Artificial Intelligence for Earth Observation: Reasoning, Uncertainties, Ethics and Beyond,” Munich, Germany. Since October 2020, she has been a Director of the Munich Data Science Institute (MDSI), TUM. From 2019 to 2022, she was a Co-Coordinator of the Munich Data Science Research School (www.mu-ds.de) and the Head of the Helmholtz Artificial Intelligence—Research Field “Aeronautics, Space and Transport.” Her research interests include remote sensing and Earth observation, signal processing, machine learning and data science, with their applications in tackling societal grand challenges, e.g., global urbanization, UN’s SDGs, and climate change.

Dr. Zhu has been a Member of young academy (Junge Akademie/Junges Kolleg), Berlin-Brandenburg Academy of Sciences and Humanities and the German National Academy of Sciences Leopoldina and the Bavarian Academy of Sciences and Humanities. She is a Fellow of of the Academia Europaea (the Academy of Europe). She serves in the scientific advisory board in several research organizations, among others the German Research Center for Geosciences (GFZ, 2020–2023) and Potsdam Institute for Climate Impact Research (PIK). She is an Associate Editor for IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, *Pattern Recognition*, and was the Area Editor responsible for special issues of IEEE Signal Processing Magazine (2021–2023). She is a Fellow of AAIA and ELLIS.