Multiscale Semantic Segmentation of Remote Sensing Images Based on Edge Optimization

Wubiao Huang, Fei Deng, Haibing Liu, Mingtao Ding, *Member, IEEE*, and Qi Yao

***Abstract*— Semantic segmentation of remote sensing images is crucial for disaster monitoring, urban planning, and land use. Due to scene complexity and multiscale features of targets, semantic segmentation of remote sensing images has become a challenging task. Deep convolutional neural networks capture remote contextual dependencies that are limited. Meanwhile, restoring the image size quickly leads to undersampling at object edges, resulting in poor boundary prediction. Therefore, this article proposes a multiscale semantic segmentation network of remote sensing images based on edge optimization, namely, multiscale edge optimization network (MSEONet). The decoder of the network consists of a multiscale context aggregation (MSCA) module, a coarse edge extraction (CEE) module, and an edge point feature optimization (EPFO) module. The MSCA module is used to capture multiscale contextual information and global dependencies between pixels. The CEE module is used for boundary extraction of multiclass coarse segmentation results. The EPFO module is used to optimize edge point features during the upsampling process. We conducted extensive experiments on the International Society for Photogrammetry and Remote Sensing (ISPRS) Potsdam 2-D dataset, the ISPRS Vaihingen 2-D dataset, and the FLAIR #1 dataset. The results show the effectiveness and superiority of our proposed MSEONet model compared to most of the state-of-the-art models. The CEE and EPFO modules can enhance the edge segmentation effect without increasing the computational and parametric quantities too much. The code is publicly available at https://github.com/HuangWBill/MSEONet.**

Received 25 September 2024; revised 26 February 2025; accepted 18 March 2025. Date of publication 21 March 2025; date of current version 7 April 2025. This work was supported in part by the Jianbing Program of Zhejiang under Grant 2023C01040, in part by the Key Research and Development Projects in Hubei Province under Grant 2022BAA035, in part by the National Natural Science Foundation of China under Grant 42374027, in part by the National Key Research and Development Program of China under Grant 2021YFC3000400, in part by the Opening Fund of Key Laboratory of Smart Earth under Grant KF2023YB04-01, in part by the Key Research and Development Program Projects in Zhejiang Province under Grant 2023C03177, and in part by the Fundamental Research Funds for the Central Universities under Grant 300102262203. *(Corresponding author: Fei Deng.)*

Wubiao Huang and Haibing Liu are with the School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China (e-mail: huangwubiao@whu.edu.cn; liuhb_whu@whu.edu.cn).

Fei Deng is with the School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China, and also with the Luojia Laboratory Hubei, Wuhan 430079, China (e-mail: fdeng@sgg.whu.edu.cn).

Mingtao Ding is with the College of Geological Engineering and Geomatics, Chang'an University, Xi'an 710054, China, also with the Key Laboratory of Loess, Xi'an 710054, China, and also with the Key Laboratory of Western China's Mineral Resource and Geological Engineering, Ministry of Education, Xi'an 710054, China (e-mail: mingtaoding@chd.edu.cn).

Qi Yao is with Ningxia Hui Autonomous Region Institute of Surveying and Mapping and Geographic Information, Ningxia 750000, China (e-mail: 554283438@qq.com).

Digital Object Identifier 10.1109/TGRS.2025.3553524

***Index Terms*— Edge point feature optimization (EPFO), multiscale context aggregation (MSCA), remote sensing images, semantic segmentation.**

I. Introduction

With the development of remote sensing technology, the acquisition of high-resolution remote sensing images has become more accessible. One of the important tasks in remote sensing is semantic segmentation, which aims to assign a category label to each pixel in the image [1], [2]. Semantic segmentation plays a crucial role in the fields of vegetation monitoring [3], [4], urban planning [5], [6], disaster monitoring [7], [8], and so on. This demonstrates that the study of semantic segmentation of remote sensing images has significant academic and application value.

In recent years, with the vigorous development of deep learning technology, image classification [9], [10], target detection [11], [12], semantic segmentation [13], [14], and other fields have made significant progress. In the field of semantic segmentation, fully convolutional networks (FCN) [15] was the first FCN proposed and applied in image semantic segmentation, which achieved end-to-end pixel-level semantic segmentation. The encoder extracts features by increasing feature channels and reducing spatial dimensions, while the decoder uses upsampling to recover the size of feature maps [16], [17], [18]. Subsequently, various improved networks have emerged, such as DeepLabV3+ [19], PSPNet [20], UNet [21], and FPN [22]. Many scholars have applied deep learning networks to the task of semantic segmentation of remote sensing images and achieved good results. However, the complex backgrounds of remote sensing images, significant interclass and intraclass scale differences [23], and dense distribution of small objects [24] also make the application of deep learning models in remote sensing have some challenges.

Due to the convolutional layer overemphasizing local features, the ability to capture remote dependencies is limited. The available contextual information is also limited by the size of the receptive field [25], [26]. Zhao et al. [20] proposed a pyramid pooling module (PPM) that represents feature maps through multiple regions of different sizes to expand the receptive field of the model. Chen et al. [19] further proposed an atrous spatial pyramid pooling (ASPP) with multiscale dilation rates to extract feature information of objects at different scales. Wang et al. [27] introduced a stripe convolution strategy to construct a multiscale convolutional attention module,

further expanding the model's receptive field. Xiao et al. [13] proposed a large field convolution (LFC) that can achieve a large receptive field with fewer parameters. It can be seen that relevant research mainly focuses on extracting multiscale feature information and updating different convolutional methods.

In addition, due to the limitations of deep learning feature extraction, the extracted features have a lower resolution. Upsampling is the most commonly used method to output segmentation images of the original resolution size, but this method can lead to poor segmentation edges [28]. As shown in Fig. 1, it can be seen that the misclassification regions of the state-of-the-art deep learning methods are mainly concentrated on the boundary of ground objects. Although fully convolutional networks [15] use deconvolution operations to recover the size of feature maps, they also introduce more parameters and computational complexity. A common strategy is to use postprocessing methods such as conditional random fields (CRFs) or morphological methods [29], [30]. A small number of scholars have achieved smoother segmentation results by converting traditional semantic segmentation tasks into distance map prediction tasks [31]. Some scholars have integrated low-level features with rich edge information into high-level features with more semantic information to improve the accuracy of prediction results [22], [32], [33]. However, these methods often result in oversampling of object interiors and undersampling of object edges.

To address the above problems, this article proposes a new multiscale edge optimization network (MSEONet) for semantic segmentation of high-resolution remote sensing images. MSEONet consists of three key components: the multiscale context aggregation (MSCA) module, the coarse edge extraction (CEE) module, and the edge point feature optimization (EPFO) module. The three modules are connected to jointly assist in obtaining semantically detailed segmentation results. The main contributions of this article can be summarized as follows.

1) An MSCA module based on pyramid pooling as the backbone was proposed. It can effectively capture rich global contextual information and multiscale information by fusing shallow with deep features.
2) A CEE module was designed to extract edge information for multiclass semantic segmentation effectively.
3) An EPFO module was developed to optimize the edge points of the coarse results obtained from MSCA. It can effectively alleviate the problem of inaccurate boundaries in the upsampling process and obtain more accurate segmentation results.
4) To verify the effectiveness of the proposed method, we conducted experiments on three datasets: the ISPRS Potsdam 2-D dataset, the ISPRS Vaihingen 2-D dataset, and the FLAIR #1 dataset. Our method produces optimal results on both datasets.

The remainder of this article is organized as follows. In Section II, we briefly review related work. We introduce the framework of the proposed model in detail in Section III. Section IV describes the dataset, implementation details, and evaluation metrics. Section V carries out a series of ablation experiments and compares them with state-of-the-art models. Finally, Section VI gives the conclusion of this article.

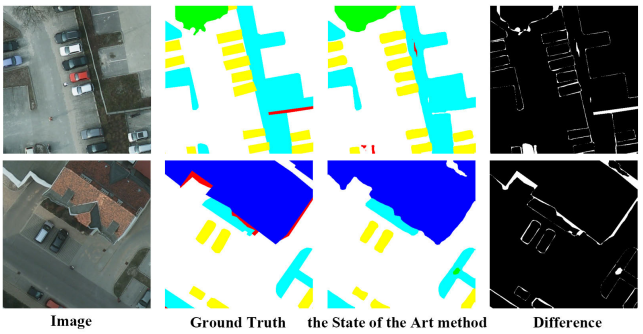

Fig. 1. Difference between the predicted results of the state-of-the-art models and the ground truth results.

II. Related Work

A. Semantic Segmentation of Remote Sensing Images

Semantic segmentation of remote sensing images is an important task in remote sensing image understanding. Traditional methods are based on manually selecting features to achieve segmentation [34]. However, this method has high labor costs, inadequate feature representation, and poor results [2]. In recent years, various models based on convolutional networks have significantly improved the segmentation accuracy of remote sensing images. The state-of-the-art framework for semantic segmentation of remote sensing images has consisted of two parts: encoder and decoder.

For the problem of multiscale and complex backgrounds in remote sensing images, the most commonly used method is to aggregate multiscale context information. Kampffmeyer et al. [35] improved FCN by introducing median frequency balancing and applied it to small target object segmentation. Diakogiannis et al. [16] constructed the ResUNet-a model based on the UNet architecture and ResBlock block with parallel dilated convolutions, and used pyramid scene parsing pooling to aggregate contextual information. Li et al. [36] proposed an attentive bilateral contextual network (ABCNet), which simultaneously preserves rich spatial details and captures global contextual information through both spatial and contextual paths. Behera et al. [37] utilized a superpixel-based multiscale convolutional neural network for the semantic segmentation of uncrewed aerial vehicle (UAV) images. The superpixel segmentation can preserve critical contextual information, and the multiscale network can extract scale invariant features. Hou et al. [24] proposed a spatial adaptive convolution-based content-aware network (SPANet) for semantic segmentation of remote sensing images. The SPANet combines a hierarchical atrous spatial pyramid (HASP) and a spatial-adaptive convolution-based feature pyramid network (SPA-FPN) decoder framework. Li et al. [38] developed an enhanced multiscale network (EMSNet) to obtain more accurate segmentation results by integrating a multistage feature mapping module and a multiscale convolutional module, but it increases computational complexity. Pang et al. [39] proposed a patch-to-region bottom-up pyramid framework to

address the problem of loss of spatial features for semantic segmentation of large-format remote sensing images. Bai et al. [40] constructed a dual-branch hybrid reinforcement network (DHRNet) to obtain more comprehensive segmentation results through a multiscale feature extraction branch and a global context and detail enhancement branch. As can be seen, the above studies focus on solving the balance problem between deep multiscale spatial feature extraction and global contextual information.

B. Edge Optimization

The feature size obtained by the encoder is much smaller than the original image size. Upsampling to the original image resolution by bilinear interpolation often leads to the absence of fine information and poor prediction of detail-rich regions such as object boundaries [41], [42]. To solve this problem, Zhu et al. [29] improved SegNet for building recognition and used morphological closure operations and erosion operations for postprocessing, effectively removing a large amount of noise. Wei et al. [30] used an improved UNet network to segment buildings from aerial images, and the edges of buildings were optimized using a boundary regularization strategy. Cheng and Lei [43] used dense CRFs (DenseCRFs) to further refine the coarse segmentation results obtained from the modified multiscale deformable convolutional neural network (mmsDCNN) model. However, these methods are often not end-to-end. The research of Li et al. [5] and Wei et al. [44] focuses on the vectorization task of building semantic segmentation results based on deep learning models. According to the shape characteristics of buildings, point-based selection and coordinate optimization are carried out to improve the vectorization accuracy of buildings. Bokhovkin and Burnaev [45] proposed a boundary loss function for binary segmentation to optimize the building segmentation boundary. These methods mentioned above are mostly studied in single-rule object semantic segmentation, and many of them are not suitable for multiclass objects. Afterward, some scholars preserved the edge information by introducing a separate branch of edge detection. Ni et al. [33] proposed an edge information-guided network that uses directional convolution modules to construct spatial detail branches to obtain accurate edge and spatial detail information. Chen et al. [46] added a separate edge enhancement branch to the original backbone network, and used the Canny algorithm to extract edge information from the original image and labels to supervise edge feature reconstruction in the model.

However, this method requires a more significant number of parameters and computational effort. Kirillov et al. [47] proposed a point-based rendering neural network module (PointRend), which is based on an iterative subdivision algorithm to output clear object boundaries during the upsampling process while significantly reducing the number of parameters. On this basis, Ding et al. [48] combined DeepLabV3 with PointRend to improve the recognition performance of ground cover details. This article also conducts relevant experiments based on the theoretical foundation of point rendering, and optimizes the selection method for uncertain points (UPs).

III. Methodology

A. Overview

The MSEONet is introduced with its overall structure in Fig. 2. The network adopts an encoder and cascaded decoder structure, and the backbone network of the encoder uses a ResNet-101 structure with dilated convolution for feature extraction. The decoder consists of two parts. Decoder 1 consists of an MSCA module, which is used to increase the multiscale representation of features. Decoder 2 consists of a CEE module and an EPFO module, which is used to optimize the segmentation edges during the upsampling process.

The deep features extracted from the backbone network for stages 2–4 have rich semantic information, which are concatenated as inputs to decoder 1. The shallow features ($edgefeatures_{fine}$) extracted from the backbone network at stage 1 have rich detailed information, which can guide the edge reconstruction in the upsampling process. $edgefeatures_{fine}$ and the output of decoder 1 as the input to decoder 2. First, the coarse result ($pred_{coarse}$) obtained from decoder 1 is upsampled two times and then uses the CEE module for edge extraction. On this basis, the EPFO module is used for UPs selection, and the features of UPs are updated by combining $edgefeatures_{fine}$ with the output of decoder 1. Finally, the final classification result is obtained. A detailed description of the critical modules is as follows. Sections III-B–III-D provide detailed descriptions of critical modules.

B. MSCA Module

In order to further reduce the loss of contextual information, this article introduces an MSCA module to mine the global contextual information. By pooling at different scales to increase the receptive field, we obtain global prior information and produce high-quality results.

Fig. 3 shows the detailed workflow of the MSCA module. First, the feature maps of encoder stages 2–4 are concatenated along the channel dimension and 1×1 convolution is performed on the concatenated feature maps. This operation can integrate the features of each stage and reduce the feature dimension, making it the same as the output dimension of the last stage of the encoder. Then, the dimensionality-reduced feature maps are subjected to an adaptive average pool at four different scales to mine global contextual information at different scales. The four sizes used in this article are (1, 2, 3, and 6). In order to maintain the weights of the global features, the dimension of the contextual representation is reduced to $(1/n)$ of the original using a 1×1 convolution after each pooling layer and upsampled to the same size as the initial feature map. Finally, the feature maps from the last stage of the encoder are concatenated with the four feature maps after upsampling in the channel dimension. A 3×3 convolution is performed on the concatenated feature maps, and the output channels are the same as the channels at each scale after pooling. In order to obtain coarse segmentation results, the output channels of MSCA are reduced to be consistent with the number of classes using 1×1 convolution so that the

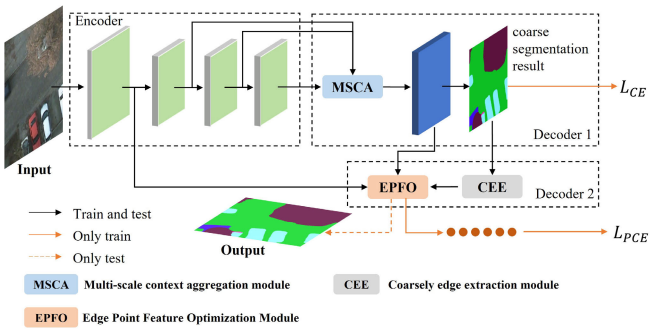

Fig. 2. Overview of the MSEONet.

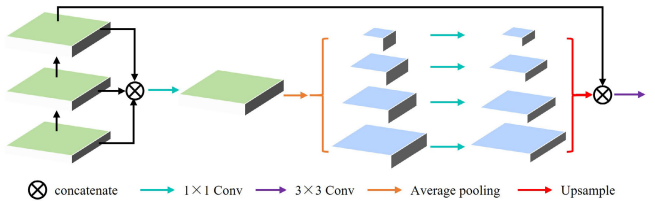

Fig. 3. Detailed workflow of the MSCA module. The rectangles of different colors represent different feature maps, and the colors correspond to Fig. 2.

probability value ($probability_{coarse}$) of each class at each pixel point is obtained.

C. CEE Module

In order to extract object edges ($edge_{pred}$) from the coarse classification results ($pred_{coarse}$), referring to the work of Bokhovkin and Burnaev [45] on the computation of the binary boundary loss function, we extend its boundary extraction to the multiclassification task. The detailed workflow of the CEE module is shown in Fig. 4(a). According to (1), one-hot coding is performed on $pred_{coarse}$. In the coded multichannel matrix ($pred_{code}$), each channel represents an object class. The matrix elements of each channel only contain 0 and 1

$$pred_{code} = onehot\left(pred_{coarse}\right). \tag{1}$$

According to (2), the max pooling operation is used to realize the boundary extraction ($multiedge_{pred}$) of each channel feature of $pred_{code}$

$$multiedge_{pred} = maxpool(1 - pred_{code}, \theta) - (1 - pred_{code}) \tag{2}$$

where the pool kernel size is θ and the pool stride size is 1. To ensure that the image size is unchanged before and after pooling, zero padding on both sides is to be used, and the width of the padding is $((\theta - 1)/2)$ downward rounded.

The $multiedge_{pred}$ obtained in the above steps is multichannel data, which is extracted for each $pred_{code}$. Therefore, it needs to be processed to obtain the final coarse boundary information. In each object class's $multiedge_{pred}$, the boundary pixel is 1, and the nonboundary pixel is 0. The $multiedge_{pred}$ of all the object classes is summed up, and the pixel position greater than 0 is taken as the final result. The calculation method is shown in the following equation:

$$edge_{pred} = sum\left(multiedge_{pred}, dim = 1\right) > 0. \tag{3}$$

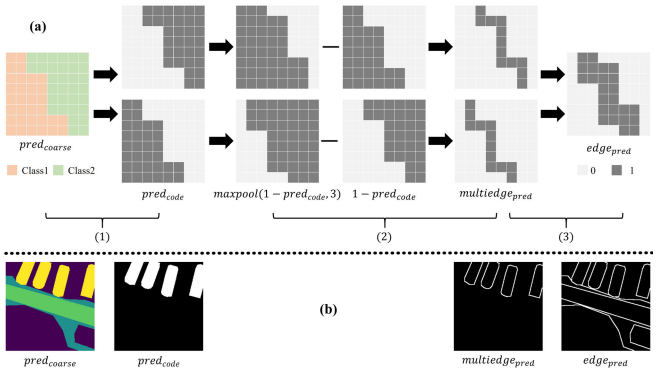

Fig. 4. (a) Detailed workflow of the CEE module and (b) example of edge extraction. The numerical labels (1)–(3) correspond to the equations.

Fig. 4(b) shows an example of edge extraction. The images from left to right correspond to $pred_{coarse}$ and $edge_{pred}$, the fourth object class ($pred_{code4}$) after one-hot coding and the coarse boundary ($multiedge_{pred4}$).

D. EPFO Module

The primary purpose of the EPFO module is to randomly select a certain number of UPs from those $edge_{pred}$ extracted in Section III-C, and for each UP, obtain its "feature vector." This "feature vector" is predicted by a multilayer perceptron (MLP) network, and the coarse prediction results are updated. This module mainly consists of the following three parts.

1) The number (N) of UP is automatically determined by the model, which is related to the minimum number of edge points in each batch. The calculation method can be defined as

$$N = max(int(min(S_k, k = 1, 2, \ldots, b) \times ratio), 1) \tag{4}$$

where S is the number of extracted edge points; b is the batch size; and $ratio$ is a customized scale parameter, which is 0.7 in this article.

2) *Selection of UP:* First, the uncertainty is calculated for all edge points. Then, the N points with the largest uncertainty are selected as the uncertainty points, and the positional coordinates (coords) of these points are saved. As shown in Fig. 5(a), the calculation method for the uncertainty of points is by extracting $probability_{coarse}$ of each point and sort it in descending on the channel dimension. The uncertainty is calculated by subtracting the top first probability value from the top second probability value as follows:

$$uncert_i = secondmax\left(probability_{coarse_i}\right) - max\left(probability_{coarse_i}\right), \quad i = 1, 2, \ldots, S \tag{5}$$

where $uncert_i$ is the uncertainty of the ith point.

3) *Feature Representation and Updating of UP:* As shown in Fig. 5(b), the "feature vectors" of the UP are obtained from $edgefeatures_{fine}$ and $probability_{coarse}$ based on the coords. These two parts are concatenated along the

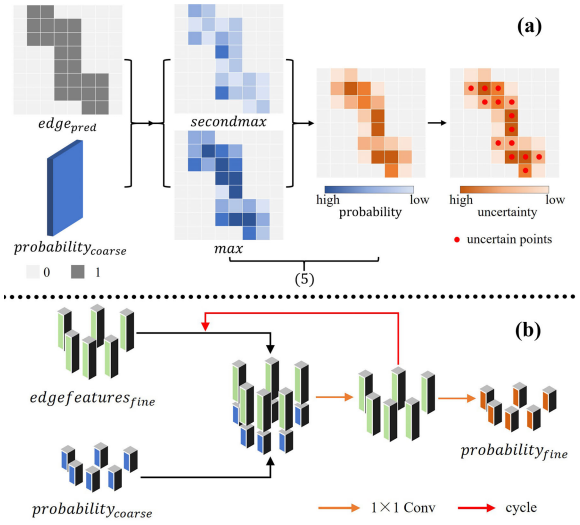

Fig. 5. Detailed workflow of (a) selection and (b) feature representation and updating of UP. The numerical label (5) corresponds to the equations.

channel dimension to form a vector as the "feature vectors." Then, use a 1×1 convolution to update the "feature vectors" of these points, and the updated "feature vectors" are taken as new $edgefeatures_{fine}$. The above updating process is repeated two times. Finally, the output channels of the updated feature vectors are reduced to be consistent with the number of classes using 1×1 convolution so that the updated probability value ($probability_{fine}$) of each class at each UP is obtained.

In the training stage, in order to facilitate the computation of the loss function and reduce the computational complexity, only $probability_{fine}$ of UP as outputs. The loss function is calculated by using outputs and extracting the actual classes of corresponding points based on coords. However, in the inference stage, it is necessary to update the probability of UP in $probability_{coarse}$ of the whole sample to $probability_{fine}$ according to the coords. Then, through the softmax to obtain the new classification result.

E. Loss Functions

As shown in Fig. 2, during the supervised training of the model, the total loss function (L) consists of two parts: the cross-entropy loss function (L_{CE}) and the pixel-point cross-entropy loss function (L_{PCE}). The calculation formula is shown in the following equation:

$$L = L_{\mathrm{CE}} + L_{\mathrm{PCE}} \tag{6}$$

where L_{CE} reflects the difference between the softmax output of decoder 1 and the true label, and the calculation formula is shown in (7). L_{PCE} reflects the difference between the prediction results of the UP of decoder 2 and the true label of the corresponding points, which is calculated as shown in (8)

$$L_{\mathrm{CE}} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{k} y_k^{(n)} \log \hat{y}_k^{(n)} \tag{7}$$

$$L_{\mathrm{PCE}} = -\frac{1}{S}\sum_{s=1}^{S}\sum_{k=1}^{k} y_k^{(s)} \log \hat{y}_k^{(s)} \tag{8}$$

where $n \in [1, 2, \ldots, N]$, in which N is the number of samples; k is the number of classes; $\hat{y}_k^{(n)}$ is the one-hot value of the predicted result of the sample; $y_k^{(n)}$ is the real label value corresponding to this sample; $S \in [1, 2, \ldots, S]$, in which S is the number of sampling points; $\hat{y}_k^{(s)}$ is the one-hot value of the prediction result of a pixel point; and $y_k^{(s)}$ is the true label value corresponding to that pixel point.

IV. Experimental Settings

A. Datasets

This article conducted experiments on two well-known open-source datasets [the International Society for Photogrammetry and Remote Sensing (ISPRS) Vaihingen 2-D dataset and the ISPRS Potsdam 2-D dataset (http://www2.isprs.org/commissions/comm3/wg4/semantic-label-ing.html)] and a multiclass dataset (FLAIR #1 [49]) to evaluate the performance of the proposed method.

1) ISPRS Potsdam 2-D Dataset: The Potsdam dataset is a typical historic city with large building complexes, narrow streets, and dense settlement structures. It consists of 38 remote sensing images with a spatial resolution of 0.05 m, all 6000×6000 pixels in size. It includes six classes of labels: impervious surfaces, building, low vegetation, tree, car, and background. In our experiments, we use only the R, G, and B bands. For dataset split, we used IDs: 2_10, 2_11, 2_12, 3_10, 3_11, 3_12, 4_10, 4_11, 4_12, 5_10, 5_11, 5_12, 6_10, 6_11, 6_12, 6_7, 6_8, 6_9, 7_10, 7_11, 7_12, 7_7, 7_8, and 7_9 for training and the remaining 14 images for testing.

2) ISPRS Vaihingen 2-D Dataset: The Vaihingen dataset is a small village with many individual buildings and small multistory buildings. It consists of 33 remote sensing images of different sizes with a spatial resolution of 0.09 m. It includes six classes of labels: impervious surfaces, building, low vegetation, tree, car, and background. In our experiments, we use only the NIR, R, and G bands. For the dataset split, we used IDs: 1, 3, 5, 7, 11, 13, 15, 17, 21, 23, 26, 28, 30, 32, 34, and 37 for training and the remaining 17 images for testing.

3) FLAIR #1 Dataset: The FLAIR #1 dataset is a part of the dataset currently used at the French National Institute of Geographical and Forest Information (IGN) to establish the French national land cover map reference. It consists of 77 412 remote sensing images with a spatial resolution of 0.2 m, all 512×512 pixels in size. It includes 12 classes of labels: building, pervious surface, impervious surface, bare soil, water, coniferous, deciduous, brushwood, vineyard, herbaceous vegetation, agricultural land, and plowed land. In our experiment, we use only the R, G, and B bands. We use the default dataset split method.

B. Implementation Details

All experiments were conducted on a Linux PC with an NVIDIA GeForce RTX 4090 GPU with 24-GB memory. All codes were implemented based on the PyTorch deep learning framework. For all comparisons, the pretrained ResNet-101

model on the ImageNet dataset is used as the backbone network. The 7×7 convolution of the input layer is replaced by three 3×3, and the final two downsampling operations are replaced by dilated convolutional layers with extension rates of 2 and 4. Taking into account the limitations of hardware conditions, our batch size is set to 4, and the number of max iterations for all training is set to 80 k.

In the training processes, we use the "AdamW" optimizer with weight decay for network optimization. The initial learning rate is set to 0.0001 and the weight decay is set to 0.001. A "poly" polynomial learning rate strategy is used, with the formula $lr = base\_lr \times (1-(iteration/max\_iteration))^{power}$, where $base\_lr$ denotes the initial learning rate, $iteration$ denotes the current iteration number, $max\_iteration$ denotes the total iteration number, and $power$ is set to 0.9.

Due to the dataset images being too large for training directly, we preprocessed the remote sensing images by cropping them to 512×512 pixels with 128 overlapping pixels. During training, we use random resizing (with scales of [0.5, 0.75, 1.0, 1.25, 1.5, 1.75, 2.0]), random cropping, and random flipping for data augmentation.

C. Evaluation Metrics

In order to comprehensively evaluate the performance of the proposed model, the overall accuracy (OA), the mean intersection over union (mIoU), and the mean $F1$-score (m$F1$) are used as evaluation metrics. Based on the accumulated confusion matrix, the OA, mIoU, and m$F1$ are computed as

$$\mathrm{OA} = \frac{\sum_{k=1}^{N} \mathrm{TP}_k}{\sum_{k=1}^{N} \mathrm{TP}_k + \mathrm{FP}_k + \mathrm{TN}_k + \mathrm{FN}_k} \tag{9}$$

$$\mathrm{mIoU} = \frac{1}{N}\sum_{k=1}^{N} \mathrm{IoU}_k = \frac{1}{N}\sum_{k=1}^{N} \frac{\mathrm{TP}_k}{\mathrm{TP}_k + \mathrm{FP}_k + \mathrm{FN}_k} \tag{10}$$

$$\mathrm{m}F1 = \frac{1}{N}\sum_{k=1}^{N} \frac{2 * \mathrm{Precision}_k * \mathrm{Recall}_k}{\mathrm{Precision}_k + \mathrm{Recall}_k} \tag{11}$$

$$\mathrm{Precision}_k = \frac{\mathrm{TP}_k}{\mathrm{FP}_k + \mathrm{TP}_k} \tag{12}$$

$$\mathrm{Recall}_k = \frac{\mathrm{TP}_k}{\mathrm{FN}_k + \mathrm{TP}_k} \tag{13}$$

where TP_k, FP_k, TN_k, and FN_k denote true positives, false positives, true negatives, and false negatives, respectively, for a particular object indexed as class k. N is the number of object classes.

V. Experimental Results and Discussion

The ablation experiments and comparative experiments in Sections V-A–V-C were only conducted on the ISPRS Potsdam and Vaihingen datasets to verify the effectiveness of each module and provide the optimal parameter combination. After the above experiments, we compared the proposed model with the state-of-the-art models on three different styled datasets to verify the advantages of the proposed method.

TABLE I
Evaluation Metrics of the MSEONet Model for Different Pooling Size Combinations on the Potsdam Dataset

Pooling size combinations	OA (%)	mF1 (%)	mIoU (%)
1, 2, 3, 6	**88.68**	**84.17**	**74.51**
1, 3, 6, 8	88.28	83.28	73.56
1, 4, 8, 12	88.61	83.96	74.31

TABLE II
Evaluation Metrics of the MSEONet Model for Different Pooling Size Combinations on the Vaihingen Dataset

Pooling size combinations	OA (%)	mF1 (%)	mIoU (%)
1, 2, 3, 6	87.61	**81.38**	**70.08**
1, 3, 6, 8	87.53	80.78	69.44
1, 4, 8, 12	**87.71**	80.58	69.37

A. Parameter Study for the MSEONet

1) Effect of Pooling Size in MSCA Module: In the MSCA module, adaptive average pools at four different sizes are used to increase the receptive field and extract global contextual information. This section analyzes the effect of different pooling size combinations on model performance on the ISPRS Potsdam and Vaihingen datasets. This article chose three different pooling size combinations for our experiments, namely, (1, 2, 3, 6), (1, 3, 6, 8), and (1, 4, 8, 12).

The OA, m$F1$, and mIoU values of the MSEONet model on the Potsdam and Vaihingen datasets for different pooling size combinations are presented in Tables I and II, respectively. It can be seen that in the Potsdam dataset, the three metrics are highest for the (1, 2, 3, 6) combination and lowest for the (1, 3, 6, 8) combination. In the Vaihingen dataset, the (1, 2, 3, 6) combination has the highest values of m$F1$ and mIoU, and the (1, 4, 8, 12) combination has the highest values of OA, but the m$F1$ and mIoU values are the lowest. Overall, the (1, 2, 3, 6) combination has the best performance on both datasets. As the pooling size combination increases, the performance of the model has a decreasing trend. This is due to the fact that the increase of the pooling size combination overemphasizes the local information, leading to a decrease in the model performance.

2) Effect of θ in CEE Module: In the CEE module, the pooling window size θ affects the width of the extracted edges, which further affects the number of extracted UPs. This section discusses the effect of different edge widths on model performance by changing the value of parameter θ (in this experiment, θ was taken as 3, 5, and 7) under the same experimental conditions.

Fig. 6 shows the extracted edges when θ is taken as 3, 5, and 7. It can be seen that as θ increases, the extracted edges become wider. The OA, m$F1$, and mIoU values of the MSEONet model on the Potsdam and Vaihingen datasets for different θ values are presented in Tables III and IV, respectively. Combining the results on the two datasets, it can be seen that the model obtained the optimal results at θ value was 5. As the value of θ increases, the model performance

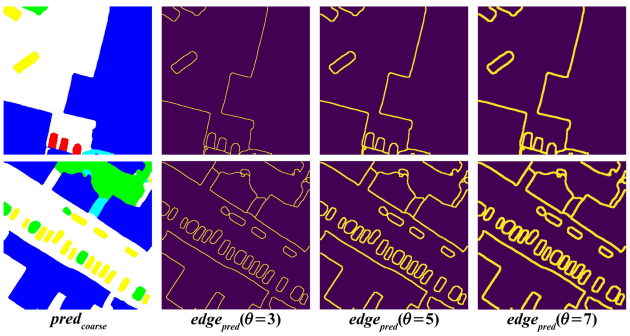

Fig. 6. Results of edge extraction ($edge_{pred}$) of the coarse classification results ($pred_{coarse}$) when taking different values of θ.

TABLE III
EVALUATION METRICS OF THE MSEONET MODEL FOR DIFFERENT θ VALUES ON THE POTSDAM DATASET

θ	OA (%)	mF1 (%)	mIoU (%)
3	88.52	83.83	74.18
5	**88.68**	**84.17**	**74.51**
7	88.57	83.89	74.22

TABLE IV
EVALUATION METRICS OF THE MSEONET MODEL FOR DIFFERENT θ VALUES ON THE VAIHINGEN DATASET

θ	OA (%)	mF1 (%)	mIoU (%)
3	87.57	80.68	69.39
5	**87.61**	**81.38**	**70.08**
7	87.58	80.68	69.38

metrics do not always increase but instead show a trend of first increasing and then decreasing. This is due to the fact that when the edge width is small, it is not possible to learn sufficient features to correctly classify difficult-to-distinguish samples. When the edge width is large, a large number of already correct pixels are used to train the EPFO module, which will cause the classifier to overfit these samples to a certain extent and reduce the classification accuracy of difficult-to-distinguish samples.

3) Effect of Different Point Sampling Ratios: In the EPFO module, N is not fixed and is automatically determined by (4), where the parameter $ratio$ influences the size of N. This section analyzes the effect on the performance of the proposed model from two aspects: whether N is fixed and the value of the parameter ratio. When N is not fixed, it is calculated by (4), and the specific point selection method is described in Section III-D. In this case, we conducted experiments on the model performance when the parameter $ratio$ was taken as 0.75 and 1.0. When N is fixed, referring to the research of Kirillov et al. [47], the N value is set to 2048 during the training process and 8096 during the inference process. The point selection method is consistent with the method when N is not fixed.

The OA, m$F1$, and mIoU values of the MSEONet model on the Potsdam and Vaihingen datasets for different point sampling ratios are presented in Tables V and VI, respectively. It can be seen that when the value of N is not fixed, all the

TABLE V
EVALUATION METRICS OF THE MSEONET MODEL FOR DIFFERENT POINT SAMPLING RATIOS ON THE POTSDAM DATASET

Number of uncertain points	$ratio$	OA (%)	mF1 (%)	mIoU (%)
Fixed	-	88.52	83.90	74.10
Unfixed	0.75	**88.68**	**84.17**	**74.51**
Unfixed	1.0	88.65	84.11	74.43

TABLE VI
EVALUATION METRICS OF THE MSEONET MODEL FOR DIFFERENT POINT SAMPLING RATIOS ON THE VAIHINGEN DATASET

Number of uncertain points	$ratio$	OA (%)	mF1 (%)	mIoU (%)
Fixed	-	87.53	80.68	69.32
Unfixed	0.75	**87.61**	**81.38**	**70.08**
Unfixed	1.0	87.41	80.64	69.34

metrics when $ratio$ is set to 0.75 are better than when $ratio$ is set to 1.0. This result indicates that using all the edge points does not necessarily give the optimal results. This is due to the fact that not all of the edge points are indistinguishable uncertainty points, and too many edge points will force the model to focus on some regions that are already classified correctly, leading to inaccurate prediction results. In addition, when the value of N is fixed, all the metrics are lower than those when the value of N is not fixed. This is because N selected in many samples is too small, it is easy to lead insufficient training of the model and unable to learn the features of these difficult points that are difficult to distinguish well.

B. Ablation Study

In order to validate the effectiveness of each module in the proposed method, we conducted relevant ablation experimental studies on the Potsdam and Vaihingen datasets. ResNet-101 was used as the base model, and the MSCA module for decoder 1 and the CEE-EPFO module for decoder 2 were gradually added to it. Due to the EPFO module depending on the CEE module, we conducted ablation experiments with these two modules as a whole.

1) Quantitative Analysis: Table VII shows the ablation experimental results on the Potsdam dataset. It can be seen that the base model only obtains 72.80% of mIoU, 88.28% of OA, and 82.35% of m$F1$, which is a poor performance. The addition of the MSCA module greatly improves the segmentation performance; the mIoU, OA, and m$F1$ are improved to 73.96%, 88.50%, and 83.63%, respectively. This is because multiscale pooling fully exploits the global contextual information. The addition of encoder 2 improves the mIoU and m$F1$ values by 0.6% and 0.5%, respectively, which indicates that the edge point optimization module can further supplement boundary details and alleviate the problem of missing information during the upsampling process. The improvement of the IoU metrics of the proposed model on the Potsdam dataset mainly focuses on impervious surfaces, low vegetation, trees, and backgrounds, with less improvement for cars. Building,

TABLE VII
EVALUATION METRICS OF ABLATION STUDY FOR THE MSEONET MODEL ON THE POTSDAM DATASET

Method	IoU (%)						OA (%)	mF1 (%)	mIoU (%)
	Impervious surface	Building	Low vegetation	Tree	Car	Background			
Base model	83.83	**91.85**	72.31	74.99	83.55	30.29	88.28	82.35	72.80
Base model + MSCA	84.01	91.55	72.67	75.00	**83.57**	36.94	88.50	83.63	73.96
MSEONet	**84.11**	91.45	**72.91**	**75.63**	83.37	**39.59**	**88.68**	**84.17**	**74.51**

as a more easily distinguishable object class, still maintains a relatively high IoU value.

Table VIII shows the ablation experimental results on the Vaihingen dataset. It can be seen that the mIoU, OA, and m$F1$ metrics are gradually increasing with the gradual addition of decoders 1 and 2. The addition of decoder 1 resulted in an increase of 2.0%, 0.03%, and 2.4% in the mIoU, OA, and m$F1$ metrics, respectively. The addition of encoder 2 resulted in an increase of 0.8%, 0.2%, and 0.7% in the mIoU, OA, and m$F1$ metrics, respectively. This is consistent with the experimental results on the Potsdam dataset. However, the Vaihingen dataset has fewer data quantities and more complex scenes compared to the Potsdam dataset, so the performance improvement is more obvious with the addition of the decoder module. The improvement of the IoU metrics of the proposed model on the Vaihingen dataset mainly focuses on the low vegetation, tree, car, and background, and the improvement of the impervious surface is relatively small. Building, as a class that is easier to distinguish, still maintains a relatively high IoU value.

2) Visualization Effect: Fig. 7 shows some examples of the visualization results of the ablation experiment on the Potsdam dataset and the Vaihingen dataset. It can be seen that after adding the MSCA module, some indistinguishable features with similar characteristics are also correctly classified, indicating that the MSCA module can effectively extract multiscale contextual information. After adding the CEE and EPFO modules, the optimization of edge point features cannot only reduce the problem of missing edge information in the upsampling process but also the joint loss function of L_{CE} and L_{PCE} can alleviate the performance degradation caused by the overemphasis on local information in MSCA to some extent. For correctly classified features, the EPFO module can optimize its boundaries to a certain extent, but for incorrectly classified features, the feature optimization module seems to have limited effectiveness.

To further demonstrate the effectiveness of the proposed EPFO module, this section randomly selects a portion of edge points for visualization. Fig. 8(a) and (b) shows the visualization results on the Vaihingen and Potsdam datasets, respectively. In Fig. 8, 1–3 represent the true label of the point, the $pred_{coarse}$ before the EPFO module, and the final classification result after the EPFO module, respectively. Different colors represent different object classes, and each column represents a different edge point. It can be seen that most of the initially misclassified edge points are correctly classified after the optimization by the EPFO module. This directly indicates that the proposed EPFO module is effective and able to optimize the edge points. However, there are still some edge points that fail to be correctly classified, and this phenomenon is inevitable.

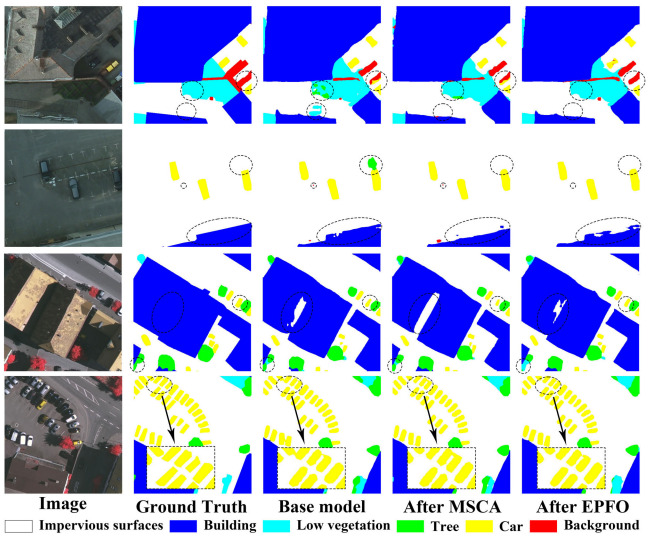

Fig. 7. Some examples of the visualization results of the ablation study on the Potsdam dataset and the Vaihingen dataset.

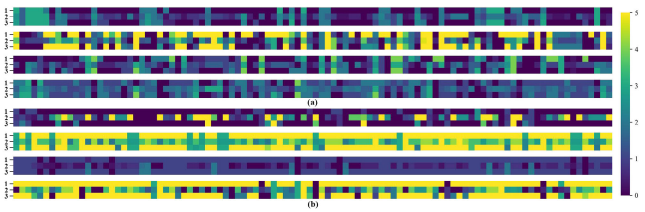

Fig. 8. Visualization of the effect of the EPFO module on some edge points on different datasets. (a) Vaihingen dataset and (b) Potsdam dataset, 1-true labels, 2-coarse results before EPFO, 3-fine results after EPFO, color bar, and object classes.

C. Comparison With Other Similar Modules

1) Multiscale Approach: To highlight the innovation of the proposed MSCA module, this section uses ResNet-101 as the base model and conducts experiments on different multiscale models proposed in previous studies as decoders. These multiscale models used for comparison include the pooling pyramid module (PPM, [20]), atmospheric spatial pyramid pooling (ASPP, [50]), and multiscale strip convolutional attention module (Strip-MSA, [27]). Table IX presents the experimental results on the ISPRS Potsdam and Vaihingen

TABLE VIII
EVALUATION METRICS OF ABLATION STUDY FOR THE MSEONET MODEL ON THE VAIHINGEN DATASET

Method	IoU (%)						OA (%)	mF1 (%)	mIoU (%)
	Impervious surface	Building	Low vegetation	Tree	Car	Background			
Base model	81.00	**87.85**	65.57	76.37	67.61	25.06	87.45	78.26	67.24
Base model + MSCA	81.21	87.56	65.42	76.27	68.49	36.66	87.48	80.60	69.27
MSEONet	**81.29**	87.43	**66.01**	**76.52**	**68.76**	**40.44**	**87.61**	**81.38**	**70.08**

TABLE IX
EXPERIMENTAL RESULTS OF DIFFERENT MULTISCALE METHODS ON POTSDAM AND VAIHINGEN DATASETS

Method	Potsdam			Vaihingen		
	OA (%)	mF1 (%)	mIoU (%)	OA (%)	mF1 (%)	mIoU (%)
PPM	84.55	83.30	73.69	87.55	80.80	69.46
ASPP	88.56	83.22	73.62	87.47	80.51	69.12
Strip-MSA	88.49	83.23	73.61	87.49	79.87	68.63
Our	88.50	83.63	73.96	87.48	80.60	69.27

TABLE X
EXPERIMENTAL RESULTS OF DIFFERENT EDGE OPTIMIZATION METHODS ON POTSDAM AND VAIHINGEN DATASETS

Method	Potsdam			Vaihingen		
	OA (%)	mF1 (%)	mIoU (%)	OA (%)	mF1 (%)	mIoU (%)
CRF	88.48	83.44	73.77	87.44	80.54	69.16
BL	88.62	83.78	74.17	87.57	81.09	69.84
pointrend	88.52	83.90	74.10	87.53	80.68	69.32
Our	88.68	84.17	74.51	87.61	81.38	70.08

datasets. It can be seen that the PPM method has the best performance among previous methods, while the Strip-MSA method has the worst performance. The results of the proposed MSCA method in this article are not significantly different from those of the PPM method. The MSCA method performs better on the Potsdam dataset.

2) Edge Optimization Approach: CRFs ([43]), boundary loss function (BL, [45]), and PointRend [47] are the methods proposed in previous studies to optimize the coarse segmentation results. In order to demonstrate the performance of the proposed EPFO module, this section is based on the base model + MSCA model, and adds the CEE + EPFO module, CRF module, BL module, and PointRend module, respectively. Relevant experiments were conducted on the Vaihingen and Potsdam datasets, and the experimental results are shown in Table X. It can be seen that the proposed CEE + EPFO module has the highest OA, m$F1$, and mIoU metrics on both datasets, while the CRF method has the worst performance in all metrics. This shows that the CEE + EPFO method has the best effect on edge optimization.

D. Comparison With the State-of-the-Art Methods

In this section, we compared the proposed method with some state-of-the-art semantic segmentation methods to demonstrate its advantages. These methods include

TABLE XI
EVALUATION METRICS OF THE MSEONET MODEL AND THE STATE-OF-THE-ART METHODS ON THE POTSDAM DATASET

Method	Backbone	auxiliary loss	OA (%)	mF1 (%)	mIoU (%)
DANet	ResNet-101	✓	88.54	83.40	73.77
CCNet	ResNet-101	✓	88.57	83.48	73.87
ACFNet	ResNet-101	✓	88.38	83.66	73.89
GCNet	ResNet-101	✓	88.67	83.88	74.28
DNLNet	ResNet-101	✓	**88.70**	83.78	74.18
LANet	ResNet-101	×	88.37	82.97	73.28
A2FPN	ResNet-101	×	88.55	83.92	74.24
CGRSeg	ResNet-101	×	88.41	83.57	73.80
MSEONet (ours)	ResNet-101	×	88.68	**84.17**	**74.51**

DANet [51], CCNet [52], ACFNet [53], GCNet [54], DNLNet [55], LANet [56], A2FPN [57], and CGRSeg [58]. All methods use ResNet-101 as the backbone network and conduct comparison experiments under the same implementation details, following their respective optimal parameters and loss function settings.

1) Comparison on the Potsdam Dataset: To validate the performance of the proposed methods, we compared them with these state-of-the-art methods on the Potsdam dataset. The evaluation metrics of each method are shown in Table XI. Compared to other methods, the proposed model has the second highest OA value after the DNLNet model, but both mIoU and m$F1$ metrics are optimal and higher than the DNLNet model. Meanwhile, compared with other models that added an auxiliary head loss function, the proposed method in this article also achieved optimal results without auxiliary head loss. The Potsdam dataset has abundant data samples and high object discrimination, so simpler models can already achieve good results. The advantages of more complex models such as DANet, CGRSeg, and ACFNet, on the contrary, are not obvious.

Fig. 9 shows some examples of the visualization results of each method. It can be seen that the proposed method has the best visualization results. In particular, the predictions of the object of impervious surface, low vegetation, and background are more accurate, the prediction of the object of the tree is slightly improved, and the predictions of the object of building and car are basically the same as those of other models. Due to low vegetation having similar color and texture to the background, it is highly similar to trees in appearance and always appears in adjacent locations. Most methods easily incorrectly predict low vegetation areas as

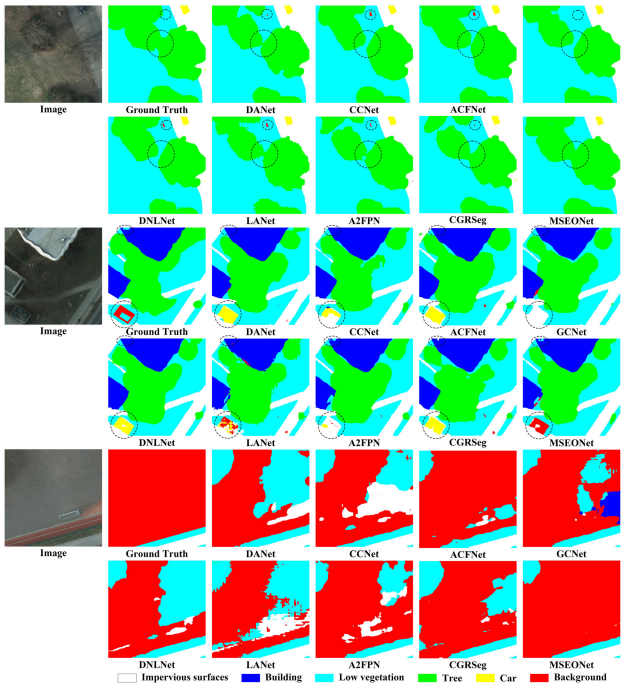

Fig. 9. Some examples of the visualization results of the MSEONet and other state-of-the-art methods on the Potsdam dataset.

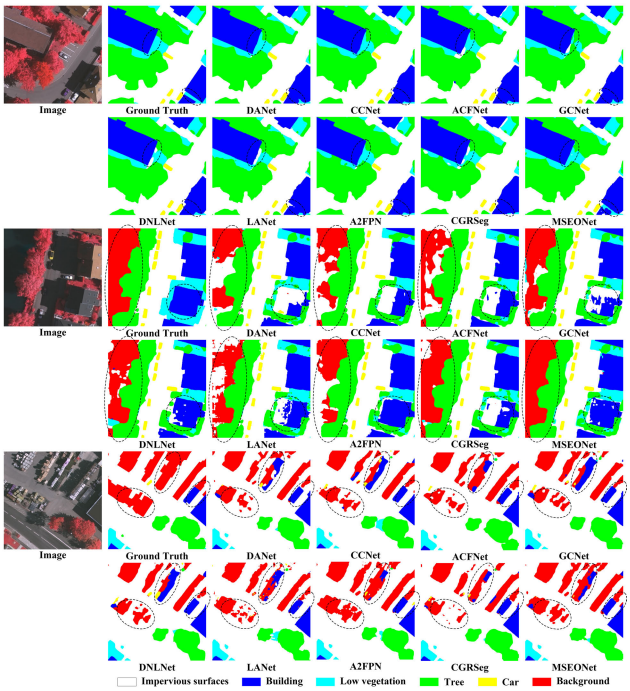

Fig. 10. Some examples of the results of the MSEONet and other state-of-the-art methods on the Vaihingen dataset.

TABLE XII
EVALUATION METRICS OF THE MSEONET MODEL AND THE STATE-OF-THE-ART METHODS ON THE VAIHINGEN DATASET

Method	Backbone	auxiliary loss	OA (%)	mF1 (%)	mIoU (%)
DANet	ResNet-101	✓	87.68	80.31	69.11
CCNet	ResNet-101	✓	87.56	80.04	68.83
ACFNet	ResNet-101	✓	87.56	80.06	68.80
GCNet	ResNet-101	✓	87.68	80.46	69.23
DNLNet	ResNet-101	✓	**87.72**	80.46	69.27
LANet	ResNet-101	×	87.58	79.88	68.59
A2FPN	ResNet-101	×	87.47	80.04	68.73
CGRSeg	ResNet-101	×	87.67	80.32	69.02
MSEONet (ours)	ResNet-101	×	87.61	**81.38**	**70.08**

TABLE XIII
EVALUATION METRICS OF THE MSEONET MODEL AND THE STATE-OF-THE-ART METHODS ON THE FLAIR #1 DATASET

Method	Backbone	auxiliary loss	OA (%)	mF1 (%)	mIoU (%)
DANet	ResNet-101	✓	72.74	69.38	55.09
CCNet	ResNet-101	✓	73.75	70.84	56.96
ACFNet	ResNet-101	✓	69.21	67.50	52.53
GCNet	ResNet-101	✓	**74.16**	71.82	57.87
DNLNet	ResNet-101	✓	73.18	69.83	55.70
LANet	ResNet-101	×	71.30	66.36	52.01
A2FPN	ResNet-101	×	72.05	70.41	55.75
CGRSeg	ResNet-101	×	72.48	70.26	55.63
MSEONet (ours)	ResNet-101	×	74.06	**72.69**	**58.72**

trees or backgrounds, but the proposed method is still able to accurately distinguish between them.

2) Comparison on the Vaihingen Dataset: To further validate the generality of the MSEONet model, we conducted experiments on the Vaihingen dataset, and the evaluation metrics of each method are shown in Table XII. The OA value of the proposed model is 87.61%, and the highest OA value is obtained from the DNLNet model. The proposed model has the highest mIoU and m$F1$ values of 70.08% and 81.38%, respectively. Due to the Vaihingen dataset containing relatively fewer training samples, lower image resolution, and more small-scale objects, the performance of most comparison methods on this dataset is not as good as the results on the Potsdam dataset.

Fig. 10 shows some examples of the visualization results of each method. It can be seen that there are many densely distributed miscellaneous classes in the background, which are adjacent to the impervious surface and have similar colors, making it easily misclassified as an impervious surface. However, the proposed method can alleviate this problem to some extent. In addition, many models easily confuse building shadow with impervious surfaces, resulting in building shadow areas being misclassified as impervious surfaces. These misclassifications make the building have incomplete segmentation results, but the proposed model is closer to the ground truth.

3) Comparison on the FLAIR #1 Dataset: To demonstrate that the MSEONet model is also applicable to scenarios with more classes, we conducted experiments on the FLAIR #1 dataset, and the experimental results are shown in Table XIII. It can be seen that the proposed model has the highest mIoU

and m$F1$ values, which are 72.69% and 58.72%, respectively. The proposed methods are much better than previous methods. Unlike typical datasets, the FLAIR #1 dataset contains richer class information for land cover domain adaptive semantic segmentation tasks. The dataset exhibits significant spatial and temporal heterogeneity within the same class, which is highly challenging. Therefore, the mIoU value of the model on this dataset is much lower than the other two datasets.

Fig. 11 shows some examples of the visualization results of each method. It can be seen that the proposed method has better segmentation performance on several land cover classes, including pervious surface, bare soil, coniferous, water, and brushwood. In addition, the shadows of herbaceous vegetation often cover buildings, resulting in the incomplete segmentation of buildings. In some areas, impervious surfaces, brushwood, and bare soil have similar colors and are difficult to distinguish. The proposed method can alleviate these problems to a certain extent.

The ISPRS Potsdam dataset is used to validate the performance of the model on the RGB bands with few classes, the ISPRS Vaihingen dataset is used to validate the performance of the model on the NIR, R, and G bands, and the FLAIR #1 dataset is used to validate the performance of the model on multiclasses data. Based on the above comparative analysis, the proposed model obtained optimal results on three datasets, and in particular, it has a better ability to distinguish between small objects and objects with consistent color. In addition, the advantages of our proposed model are more obvious when there are more classes in the dataset.

E. Efficiency Analysis

This section calculated the total number of parameters (Params) and floating point operations per second (FLOPs) for each part of the proposed model and some state-of-the-art methods mentioned in Section V-D under the same experimental conditions. The calculation results are shown in Table XIV. It can be seen that the proposed method in this article has lower Params than the DNLNet and ACFNet models. At the same time, FLOPs are lower than ACFNet models, and there is no significant increase compared to DANet, CCNet, A2FPN, and CGRSeg. Combined with the conclusion in Section V-D, it can be seen that the proposed method in this article has achieved better segmentation results and saved memory resources to a certain extent without significantly increasing computational complexity.

As can be seen from Table XIV, compared with the base model, the MSCA module in our proposed model takes up most of the FLOPs and Params. The total number of parameters in the CEE and EPFO modules is only 0.204 M, which is much smaller than the other modules. Meanwhile, the FLOPs of the CEE and EPFO modules are only 0.006 T, which is approximately (1/10) of the FLOPs of MSCA modules.

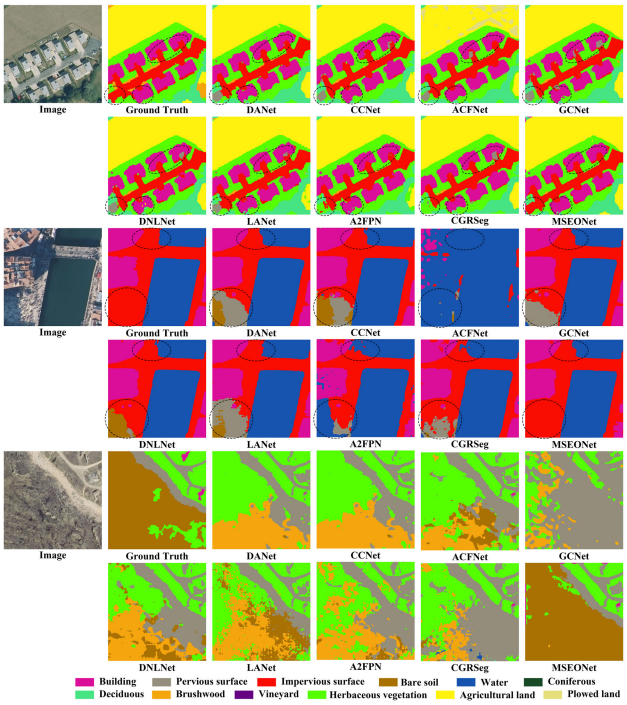

Fig. 11. Some examples of the results of the MSEONet and other state-of-the-art methods on the FLAIR #1 dataset.

TABLE XIV
EFFICIENCY ANALYSIS FOR EACH PART OF THE PROPOSED MODEL AND SOME STATE-OF-THE-ART METHODS

Methods	FLOPs (T)	Params (M)
DANet	0.289	66.454
CCNet	0.279	66.447
ACFNet	0.349	84.997
GCNet	0.18	42.812
DNLNet	0.077	376
LANet	0.231	55.506
A2FPN	0.276	66.251
CGRSeg	0.286	66.645
Base model	0.218	59.305
Base model + MSCA	0.287 (+0.069)	72.94 (+13.635)
MSEONet (ours)	0.293 (+0.006)	73.144 (+0.204)

VI. CONCLUSION

In this article, a multiscale remote sensing image semantic segmentation network based on edge optimization is proposed, namely, MSEONet. It can comprehensively extract contextual information from both global and multiscale perspectives and optimize edge segmentation results for high efficiency. In particular, we propose an optimization module based on edge points to solve the problem of boundary loss during upsampling through point rendering. In addition, in order to adaptively select UPs, the CEE module is proposed to extract multiclass edge information quickly. To validate the proposed approach, we conducted experiments on three high-resolution remote sensing image semantic segmentation datasets: the ISPRS Potsdam 2-D dataset, the ISPRS Vaihingen 2-D dataset, and the FLAIR #1 dataset. The results show that MSEONet

outperforms the other methods in most of the metrics, which demonstrates the effectiveness of the proposed method. The results of the model are mainly determined by the MSCA module, and the CEE and EPFO modules are only able to accurately adjust the pixel edges that are already correctly classified. In the future, the CEE and EPFO modules can be applied as separate decoders in other main network architectures to enhance the segmentation results without increasing excessive computational and parameters.

REFERENCES

[1] T. Xiao, Y. Liu, Y. Huang, M. Li, and G. Yang, "Enhancing multiscale representations with transformer for remote sensing image semantic segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5605116.

[2] N. Bagwari, S. Kumar, and V. S. Verma, "A comprehensive review on segmentation techniques for satellite images," *Arch. Comput. Methods Eng.*, vol. 30, no. 7, pp. 4325–4358, Sep. 2023.

[3] H. Chen et al., "Research on land cover type classification method based on improved MaskFormer for remote sensing images," *PeerJ Comput. Sci.*, vol. 9, p. e1222, Feb. 2023.

[4] Y. Li, Y. Zhou, Y. Zhang, L. Zhong, J. Wang, and J. Chen, "DKDFN: Domain knowledge-guided deep collaborative fusion network for multimodal unitemporal remote sensing land cover classification," *ISPRS J. Photogramm. Remote Sens.*, vol. 186, pp. 170–189, Apr. 2022.

[5] W. Li et al., "Joint semantic-geometric learning for polygonal building segmentation from high-resolution remote sensing images," *ISPRS J. Photogramm. Remote Sens.*, vol. 201, pp. 26–37, Jul. 2023.

[6] W. Zhou, H. Zhang, W. Yan, and W. Lin, "MMSMCNet: Modal memory sharing and morphological complementary networks for RGB-T urban scene semantic segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 12, pp. 7096–7108, Dec. 2023.

[7] S. Ji, D. Yu, C. Shen, W. Li, and Q. Xu, "Landslide detection from an open satellite imagery and digital elevation model dataset using attention boosted convolutional neural networks," *Landslides*, vol. 17, no. 6, pp. 1337–1352, Jun. 2020.

[8] J. Yang et al., "A generalized deep learning-based method for rapid coseismic landslide mapping," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 16970–16983, 2024.

[9] W. Huang et al., "Landslide susceptibility mapping and dynamic response along the sichuan-tibet transportation corridor using deep learning algorithms," *CATENA*, vol. 222, Mar. 2023, Art. no. 106866.

[10] Q. Liu, L. Xiao, J. Yang, and Z. Wei, "CNN-enhanced graph convolutional network with pixel- and superpixel-level feature fusion for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8657–8671, Oct. 2021.

[11] Y. Li et al., "Old landslide detection using optical remote sensing images based on improved YOLOv8," *Appl. Sci.*, vol. 14, no. 3, p. 1100, Jan. 2024.

[12] H. Zhang, L. Wang, and J. Sun, "Knowledge-based reasoning network for object detection," in *Proc. IEEE Int. Conf. Image Process.*, Aug. 2021, pp. 1579–1583.

[13] R. Xiao, C. Zhong, W. Zeng, M. Cheng, and C. Wang, "Novel convolutions for semantic segmentation of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5907313.

[14] S. Zhou et al., "DSM-assisted unsupervised domain adaptive network for semantic segmentation of remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5608216.

[15] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[16] F. I. Diakogiannis, F. Waldner, P. Caccetta, and C. Wu, "ResUNet-A: A deep learning framework for semantic segmentation of remotely sensed data," *ISPRS J. Photogramm. Remote Sens.*, vol. 162, pp. 94–114, Apr. 2020.

[17] H. Zhang et al., "ResNeSt: Split-attention networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 2735–2745.

[18] L. Luo, P. Li, and X. Yan, "Deep learning-based building extraction from remote sensing images: A comprehensive review," *Energies*, vol. 14, no. 23, p. 7982, Nov. 2021.

[19] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, Jan. 2018, pp. 833–851.

[20] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6230–6239.

[21] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI*. Munich, Germany: Springer, Oct. 2015, pp. 234–241.

[22] A. Kirillov, R. Girshick, K. He, and P. Dollár, "Panoptic feature pyramid networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6392–6401.

[23] X. Dong, C. Zhang, L. Fang, and Y. Yan, "A deep learning based framework for remote sensing image ground object segmentation," *Appl. Soft Comput.*, vol. 130, Nov. 2022, Art. no. 109695.

[24] J. Hou, Z. Guo, Y. Feng, Y. Wu, and W. Diao, "SPANet: Spatial adaptive convolution based content-aware network for aerial image semantic segmentation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 2192–2204, 2023.

[25] L. Huang, B. Jiang, and S. Lv, "Deep learning-based semantic segmentation of remote sensing images: A survey," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 8370–8396, 2023.

[26] K. Nogueira, M. D. Mura, J. Chanussot, W. R. Schwartz, and J. A. Dos Santos, "Dynamic multicontext segmentation of remote sensing images based on convolutional networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7503–7520, Oct. 2019.

[27] T. Wang et al., "MCAT-UNet: Convolutional and cross-shaped window attention enhanced UNet for efficient high-resolution remote sensing image segmentation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 9745–9758, 2024.

[28] B. Neupane, T. Horanont, and J. Aryal, "Deep learning-based semantic segmentation of urban features in satellite images: A review and meta-analysis," *Remote Sens.*, vol. 13, no. 4, p. 808, Feb. 2021.

[29] B. Zhu, H. Gao, X. Wang, M. Xu, and X. Zhu, "Change detection based on the combination of improved SegNet neural network and morphology," in *Proc. IEEE 3rd Int. Conf. Image, Vis. Comput. (ICIVC)*, Jun. 2018, pp. 55–59.

[30] S. Wei, S. Ji, and M. Lu, "Toward automatic building footprint delineation from aerial images using CNN and regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 2178–2189, Mar. 2020.

[31] D. Chai, S. Newsam, and J. Huang, "Aerial image semantic segmentation using DCNN predicted distance maps," *ISPRS J. Photogramm. Remote Sens.*, vol. 161, pp. 309–322, Mar. 2020.

[32] Z. Dong, J. Li, T. Fang, and X. Shao, "Lightweight boundary refinement module based on point supervision for semantic segmentation," *Image Vis. Comput.*, vol. 110, Jun. 2021, Art. no. 104169.

[33] Y. Ni, J. Liu, J. Cui, Y. Yang, and X. Wang, "Edge guidance network for semantic segmentation of high-resolution remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 9382–9395, 2023.

[34] X. Zhang, G. Chen, W. Wang, Q. Wang, and F. Dai, "Object-based land-cover supervised classification for very-high-resolution UAV images using stacked denoising autoencoders," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 7, pp. 3373–3385, Jul. 2017.

[35] M. Kampffmeyer, A.-B. Salberg, and R. Jenssen, "Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2016, pp. 680–688.

[36] R. Li, S. Zheng, C. Zhang, C. Duan, L. Wang, and P. M. Atkinson, "ABCNet: Attentive bilateral contextual network for efficient semantic segmentation of fine-resolution remotely sensed imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 181, pp. 84–98, Nov. 2021.

[37] T. K. Behera, S. Bakshi, M. Nappi, and P. K. Sa, "Superpixel-based multiscale CNN approach toward multiclass object segmentation from UAV-captured aerial images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 1771–1784, 2023.

[38] T. Li, Z. Cui, Y. Han, G. Li, M. Li, and D. Wei, "Enhanced multi-scale networks for semantic segmentation," *Complex Intell. Syst.*, vol. 10, no. 2, pp. 2557–2568, Apr. 2024.

[39] S. Pang, Y. Shi, H. Hu, L. Ye, and C. Jia, "PTRSegNet: A patch-to-region bottom-up pyramid framework for the semantic segmentation of large-format remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 3664–3673, 2024.

[40] Q. Bai, X. Luo, Y. Wang, and T. Wei, "DHRNet: A dual-branch hybrid reinforcement network for semantic segmentation of remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 4176–4193, 2024.
[41] W. Long, Y. Zhang, Z. Cui, Y. Xu, and X. Zhang, "Threshold attention network for semantic segmentation of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 4600312.
[42] D. Wu, Z. Guo, A. Li, C. Yu, C. Gao, and N. Sang, "Conditional boundary loss for semantic segmentation," *IEEE Trans. Image Process.*, vol. 32, pp. 3717–3731, 2023.
[43] X. Cheng and H. Lei, "Semantic segmentation of remote sensing imagery based on multiscale deformable CNN and DenseCRF," *Remote Sens.*, vol. 15, no. 5, p. 1229, Feb. 2023.
[44] S. Wei, T. Zhang, S. Ji, M. Luo, and J. Gong, "BuildMapper: A fully learnable framework for vectorized building contour extraction," *ISPRS J. Photogramm. Remote Sens.*, vol. 197, pp. 87–104, Mar. 2023.
[45] A. Bokhovkin and E. Burnaev, "Boundary loss for remote sensing imagery semantic segmentation," in *Proc. Int. Symp. Neural Netw.*, Jan. 2019, pp. 388–401.
[46] L. Chen, Z. Qu, Y. Zhang, J. Liu, R. Wang, and D. Zhang, "Edge-enhanced GCIFFNet: A multiclass semantic segmentation network based on edge enhancement and multiscale attention mechanism," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 4450–4465, 2024.
[47] A. Kirillov, Y. Wu, K. He, and R. Girshick, "PointRend: Image segmentation as rendering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9796–9805.
[48] L. Ding et al., "Pointnet: Learning point representation for high-resolution remote sensing imagery land-cover classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS*, Jul. 2021, pp. 4956–4959.
[49] A. Garioud, S. Peillet, E. Bookjans, S. Giordano, and B. Wattrelos, "FLAIR #1: Semantic segmentation and domain adaptation dataset," 2022, *arXiv:2211.12979*.
[50] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.
[51] J. Fu et al., "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 3141–3149.
[52] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 603–612.
[53] F. Zhang et al., "ACFNet: Attentional class feature network for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6797–6806.
[54] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "GCNet: Non-local networks meet squeeze-excitation networks and beyond," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 1971–1980.
[55] M. Yin et al., "Disentangled non-local neural networks," in *Proc. Eur. Conf. Comput. Vis., Cham*, Jan. 2020, pp. 191–207.
[56] L. Ding, H. Tang, and L. Bruzzone, "LANet: Local attention embedding to improve the semantic segmentation of remote sensing images," in *Proc. IEEE Trans. Geosci. Remote Sens.*, May 2020, vol. 59, no. 1, pp. 426–435.
[57] R. Li, L. Wang, C. Zhang, C. Duan, and S. Zheng, "A2-FPN for semantic segmentation of fine-resolution remotely sensed images," *Int. J. Remote Sens.*, vol. 43, no. 3, pp. 1131–1155, Feb. 2022.
[58] Z. Ni et al., "Context-guided spatial feature reconstruction for efficient semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 239–255.

Wubiao Huang received the B.S. degree from Xiamen University of Technology, Xiamen, China, in 2020, and the M.Sc. degree in photogrammetry and remote sensing from Chang'an University, Xi'an, China, in 2023. He is currently pursuing the Ph.D. degree with the School of Geodesy and Geomatics, Wuhan University, Wuhan, China.

His research interests include deep learning, landslide susceptibility mapping, semantic segmentation for remote sensing, and knowledge graphs.

Fei Deng received the B.S., M.S., and Ph.D. degrees in photogrammetry and remote sensing from Wuhan University, Wuhan, Hubei, China, in 1999, 2002, and 2006, respectively.

He is currently a Professor with Wuhan University. His research interests include 3-D reconstruction, parametric modeling, and machine learning for remote sensing images and point clouds.

Haibing Liu received the B.S. and M.Sc. degrees from Wuhan University, Wuhan, China, in 2020 and 2022, respectively, where he is currently pursuing the Ph.D. degree with the School of Geodesy and Geomatics.

His research interests include deep learning, 3-D reconstruction, 3-D generation, and computer vision.

Mingtao Ding (Member, IEEE) received the B.S. and Ph.D. degrees in applied mathematics from Northwestern Polytechnical University, Xi'an, China, in 2007 and 2010, respectively.

From 2011 to 2012, he was a Post-Doctoral Fellow with the Department of Electrical and Computer Engineering, Duke University, Durham, NC, USA. In 2013, he joined the College of Geological Engineering and Geomatics, Chang'an University, Xi'an, where he is currently an Associate Professor in remote sensing science and technology. His research focuses on landslide detection, mapping, and monitoring with interferometric synthetic aperture radar and multispectral remote sensing. He also specializes in machine learning, artificial intelligence, and applied statistics.

Qi Yao received the bachelor's degree in surveying and mapping engineering from Wuhan University, Wuhan, Hubei, China, in 2011.

Currently, he is a Senior Engineer with Ningxia Hui Autonomous Region Institute of Surveying and Mapping and Geographic Information, Ningxia, China. His main work and research interests are remote sensing image processing and application, 3-D construction, and application of real scenes.