High-Resolution Remote Sensing Image Semantic Segmentation via Multiscale Context and Linear Self-Attention

Peng Yin, Dongmei Zhang, Wei Han, Jiang Li, and Jianmei Cheng

***Abstract*—Remote sensing image semantic segmentation, which aims to realize pixel-level classification according to the content of remote sensing images, has broad applications in various fields. Thanks to the superiority of deep learning (DL), the semantic segmentation model based on the convolutional neural network (CNN) dramatically promotes the development of remote sensing image semantic segmentation. Due to the high resolution, comprehensive coverage, extensive data, and sizeable spectral difference of high-resolution remote sensing images (HRRSI), the existing GPU is not suitable for directly semantic segmentation of the whole image. Cutting the image into small patches will lead to the loss of context information, resulting in the decline of accuracy. To address this issue, we propose the multiscale context self-attention network (MSCSANet). It combines the benefits of the self-attention mechanism with CNN to improve the segmentation quality of various remote sensing images. The MSCSANet extracts multiscale features from multiscale context images to solve the problem of feature loss caused by image segmentation. In addition, in order to make use of the feature of large-scale context, the multiscale context patches are used to guide the local image patch to focus on different fine-grained objects to enhance the feature of the local image patch. Moreover, considering the limited computing resources, we designed a linear self-attention module to reduce the computational complexity. Compared with other DL models, our proposed model can enhance the ability of multiscale features in complex scenes, and realizes improvements of 1.56% mean intersection over union (MIoU) on the Gaofen Image Dataset and 1.93% MIoU on the ISPRS Potsdam Dataset, respectively.**

***Index Terms*—Context, remote sensing, self-attention, semantic segmentation.**

I. Introduction

SEMANTIC segmentation that achieves pixel-level classification is a critical content in remote sensing research. It is critical in land resource management [1], urban planning [2], production estimation [3], [4], and economic evaluation [5].

Deep learning (DL), a powerful approach to learning the internal regularity and representation levels of sample data [6], [7], has made significant progress in the research of high-resolution remote sensing images (HRRSI). The semantic segmentation model [8] based on convolutional neural network (CNN) shows excellent performance in pixel-level classification of HRRSI. However, due to the image's high resolution, the semantic segmentation of the whole image will consume a lot of computing resources, which leads to the contradiction between the memory occupation and the segmentation effect.

Consequently, to segment the HRRSI, the standard method is to segment it into local image patches, classify the local image patches at the pixel level, and finally merge all patches. However, as the same object can be segmented into different local image patches, it is difficult to ensure the integrity of feature, which reduces the quality of segmentation, thereby remaining an intractable problem. In order to improve the segmentation quality, Chen et al. [9] proposed collaborative global-local networks (GLNet) to fuse whole image features with local features. This network uses local image patches to obtain the fine structural features of the target and uses global image patches to get context correlation. However, this method does not fully use the correlation between global features and local features, resulting in the wrong classification of some scale features. The self-attention mechanism [10] has a powerful ability to capture long-term dependence and feature correlation. In this article, the self-attention mechanism is introduced to calculate the feature correlation between local patch and mutiscale context patch, which solves the above problem.

In recent years, based on the self-attention mechanism, the transformer [10] has shown remarkable performance in most tasks of natural language processing. With the improvement of dot-product attention in computer vision, the nonlocal module [11] is widely used in image classification, object detection, semantic segmentation, and panoramic segmentation. The time and memory complexity of the dot-product attention mechanism is $o(N^2)$, which makes it challenging to deal with remote sensing images with increasing image resolution. In order to reduce the time and memory overhead of the attention mechanism,

Manuscript received 18 April 2022; revised 20 June 2022, 6 August 2022, and 14 September 2022; accepted 12 October 2022. Date of publication 17 October 2022; date of current version 2 November 2022. This work was supported by the Key projects supported by NSFC joint fund under Grant U1911205 and Grant U19B6003, in part by the science and technology project of Hubei Provincial Department of Natural Resources under Grant ZRZY2022KJ07, and in part by the Hubei Natural Science Foundation of China under Grant 2019CFA023. *(Corresponding author: Dongmei Zhang.)*

Peng Yin, Dongmei Zhang, Wei Han, and Jianmei Cheng are with the China University of Geosciences, Wuhan 430074, China (e-mail: pengyin@cug.edu.cn; cugzdm@foxmail.com; weihan@cug.edu.cn; janiemcheng@163.com).

Jiang Li is with the Department of Natural Resources of Hubei Province, Information Center, Wuhan 430074, China (e-mail: johnlee1124@126.com).

Digital Object Identifier 10.1109/JSTARS.2022.3214889

Child et al. [12] performed a sparse decomposition of the attention matrix, reducing its complexity to $O(N\sqrt{N})$. In addition, Kitaev et al. [13] used locality-sensitive hashing to reduce the storage of the $Q * K$ matrix while maintaining accuracy and finally reduced the complexity to $O(N\log N)$. In addition, Katharopoulos et al. [14] used kernel functions to simplify the attention calculation process and replaced softmax to reduce the complexity to $O(N)$. In addition, Shen et al. [15] proposed the concept of resource efficiency, which can flexibly integrate effective attention modules into the neural network, reducing the complexity to $O(N)$.

In this article, we explore the research of using the self-attention mechanism to calculate the correlation between multiscale context patches and local image patches. Moreover, we propose a multiscale context linear self-attention mechanism network model called MSCSANet. This model still follows the idea of cutting small pieces of the original image, and it can calculate the correlation between context patches and local patches at different scales. The multiscale context patches are used to guide the local image patch to focus on different fine-grained objects to enhance the feature of the local image patch. Moreover, in order to reduce the time complexity of the algorithm, this article achieves the linear time complexity computed for correlation with local patches using a linear self-attention mechanism. The kernel function is processed by derivation to reduce the space and time complexity to $O(N)$.

In the experimental part, the Potsdam dataset and the Gaofen Image Dataset (GID) are selected to verify the effectiveness of this method. We compare the experimental results with U-Net [16], DeeplabV3+[17], DANet [18], PSPNet [19], RefineNet [20], SegNet [21], ACFNet [22], and OCRNet [23]. The accuracy of the proposed model significantly outperformed the comparison models, which shows the advance of the multiscale context linear self-attention mechanism in the proposed model.

Our main contributions are as follows:

1) We propose a new kernel function with linear complexity to reduce the complexity of the self-attention mechanism to $O(N)$.
2) We propose a multiscale context model to calculate the correlation between different scale contexts and local image patches and strengthen the feature of local image patches.
3) Our proposed multiscale contextual linear self-attention module to learn representations from HRRSI shows significant performance improvements on boundary regions on GID and Potsdam datasets.

In Section II, the related work of remote sensing image semantic segmentation is introduced, and some studies on the self-attention mechanism are reviewed. In Section III, we introduce the proposed MSCSANet framework method in detail, including linear self-attention push process, multiscale context self-attention mechanism module, and implementation details. The Section IV introduces the experimental datasets and experimental results. Section V concludes this article.

II. Related Works

A. Semantic Segmentation of Remote Sensing Images

HRRSI plays an essential role in many application fields such as land resource management, natural disaster detection, urban planning, and production estimation. The accuracy of these applications is strongly correlated with the accuracy of image semantic segmentation [1], [2], [3], [4], [5]. Traditional methods utilize spectral information and texture information, which consume enormous human and material resources. Due to the rapid development of DL, significant breakthroughs have been made in the semantic segmentation of HRRSI. DL-based semantic segmentation networks significantly improve the segmentation of HRRSI. Full convolutional network (FCN) [8] is the first network structure of CNN for semantic segmentation. FCN uses skip structure and full convolution method to achieve pixel-level classification of images. However, the FCN is not sensitive to image details and does not consider the relationship between pixels, lacking spatial consistency. U-Net [16] uses skip connections to fuse low-level features with high-level features and has both a shrinking path that captures contextual information and a symmetric expanding path that allows precise localization. PSPNet [19] uses pyramid pooling technology to fuse multiscale features, which improves prediction accuracy. CascadePSP [24] uses a cascaded segmentation optimization model to achieve high-resolution image segmentation. However, the above model is relatively complex and consumes many resources. In response to the problem of resource consumption, ICNet [23] and ENet [25] significantly reduce the model parameters and calculation amount by improving the model structure. However, these models cannot extract HRRSI's spatial contextual information well due to solid spatial dependencies, resulting in poor segmentation quality for many delicate objects.

B. Multiscale and Context Aggregation

Multi-scale information has effectively improved segmentation accuracy in semantic segmentation tasks. ParseNet [26] combines global pooling into contexts at different aggregation levels, introducing global information and expanding the receptive field. RefineNet [20] proposes the multiresolution fuse module and the chained residual pooling module, which use multilevel abstraction for high-resolution semantic segmentation. However, it still leads to the problem of partial loss of boundary information. The Deeplab series [17], [27], [28], [29] is proposed to obtain multiscale image information through image pyramid, atrous convolution, and atrous spatial pyramid pooling. It can extract dense image features, increase the receptive field, and use the conditional random field for structure prediction, which improves the spatial accuracy of the segmentation results. However, this method has a slow segmentation speed and is inadequate for small-scale object segmentation. ENCNet [30] uses the context encoding module to capture the global context information, highlight the class information associated with the scene, and use SE-Loss to improve the segmentation effect of small objects. FastFCN [31] proposes a joined pyramid upsampling called

JPU. The JPU is extracted from multiple high-resolution feature images to replace the time and memory consumption in whole convolution. Auto-deeplab [32] first introduces neural architecture search (NAS) into the field of semantic segmentation to automatically search network results. DetectoRS [33] proposes a recursive pyramid and switchable ASPP, which significantly improves detection performance. Although multiscale context information can better represent the relationship between objects, HRRSI contains significant differences in the scale of ground features. Preprocessing operations such as cropping of remote sensing images are usually required, which will limit the extraction of contextual information. GLNet [9] uses global and local branches for deep feature sharing to address this problem, which improves segmentation quality by reducing GPU memory consumption. In addition, Qi Li et al. [34] proposed a multicontext local segmentation model based on GLNet, combining local results into high-definition results through a context refinement model. In recent studies [9], [35], [36], [37], combining deep and shallow branches has also achieved good results.

C. *Self-Attention Mechanismn*

The self-attention mechanism was first inspired by the human attention mechanism and has been dramatically developed in the DL wave. The attention mechanism is developing in the wave of DL and is widely used in natural language processing, image detection, speech recognition, and other fields. Bahdanau et al. [38] introduced a dot-product attention mechanism for simultaneous translation and alignment. Transformer [17] extracts the internal feature of images and the correlation between features by self-attention, proving the superiority of the attention mechanism in natural language processing. Wang et al. [11] proposed a nonlocal module and applied the dot-product attention mechanism to the computer vision domain. In addition, there are other lightweight attention mechanisms, such as the convolutional block attention module (CBAM) [39]. CBAM summarizes the attention information from both spatial and channel aspects by constructing two submodules (spatial attention module (SAM) and channel attention module (CAM), respectively, and integrates the information to a certain extent. Zhang et al. [40] proposed an effective channel attention module (ECA), which can significantly improve the accuracy only by adding a few parameters.

These attention mechanisms are entirely different in principle and purpose from the dot-product attention mechanism. This article mainly studies the dot-product attention mechanism. Many models appeared after the nonlocal module was proposed, such as DANet [18], PSANet [41], and OCNet [42]. Similar to CBAM, the idea of DANet [18] is to integrate the attention information of the channel and space. The difference is that the acquisition of dual attention information in CBAM is serial, while dual attention information in DANet [18] is parallel. Furthermore, DANet [18] uses position and channel attention modules to learn spatial and channel interdependencies. These models obtain attention information differently and achieve satisfactory results in semantic segmentation tasks.

However, the dot-product attention mechanism usually consumes many GPU resources, especially for high-resolution images. In order to reduce the memory consumption of the dot product attention mechanism, many pieces of research have been done from different perspectives. For example, sparse attention [13], [43], [44], [45] introduces sparsity bias into the attention mechanism to reduce complexity. Linear attention uses kernel eigenmaps to solve the attention matrix and then computes the attention in reverse order to obtain linear complexity. Prototype [46], [47] and memory compaction [48], [49], [50] reduce the size of the attention matrix by reducing the number of queries or key-value memory pairs. Low-level attention [51], [52], [53] studies the low-level features of attention. However, these methods are designed for natural scenes and cannot achieve the same effect when applied to HRRSI.

III. Methodology

In this section, we will introduce the proposed network in four sections. First, the overall structure of the MSCSANet network is introduced in Section A (as shown in Fig. 1), then the multiscale context is introduced in Section B. In addition, the derivation process of the linear self-attention mechanism is introduced in Section C. Finally, in Section D, multiscale context linear self-attention is introduced.

A. *MSCSANet Overall Architecture*

It is essential to improve the accuracy of remote sensing image segmentation by making full use of semantic information and spatial context information. The typical problems of remote sensing image semantic segmentation are feature loss and the limitation of GPU resources. In order to solve these problems, a multiscale feature extraction model with context is designed. We use ResNet [54] with training weights as the feature extraction network. The multiscale context linear self-attention mechanism model improves the feature representation ability of the model by fusing multiscale information.

As shown in Fig. 1, two context image patches of different sizes are cropped for each patch, and the context image patches are reshaped to the size of the local image patch to reduce the computational overhead. First, the processed context image patches are fed into the multiscale context feature extraction network to extract the context feature. Then, through the position attention module (PAM) and channel attention module (CAM), the correlation within the feature map and the correlation the between multiscale context feature map and the local feature map are calculated to strengthen or weaken some features. Finally, the feature image is changed to the original size with up-sampling. The multiscale context self-attention module strengthens the feature of local images and dramatically reduces the misclassification caused by incomplete features.

B. *Context of Local Patch*

Image down-sampling or segmentation into small blocks will cause feature loss. Therefore, this article proposes a multiscale context model, which cuts the image into small pieces. Then, the

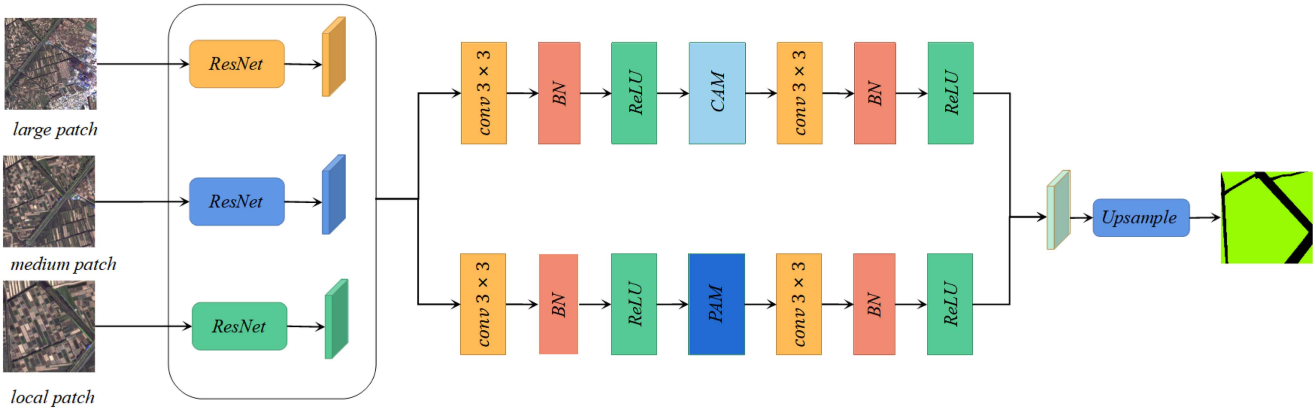


Fig. 1. Overall structure of MSCSANet model.

multiscale context patch (include large patch and middle patch) is used to enhance the feature of local patches to improve the model's effectiveness.

Suppose a high-resolution image m has a height of H and a width of W. First, the image is evenly divided into local image patches with width w and height h, in which any two patches do not intersect. Then, a magnification parameter α $(\alpha > 1)$ is designed to intercept the mutliscale context patch with the width of αw and the height of αh $(w < \alpha w < W, h < \alpha h < H)$ from the high-resolution image. The multiscale context patch contains local image patch, and the large context image patches will provide more information and have a better effect on large objects. The middle context image patches provide detailed information, effective for small object feature extraction. Before entering the network, all context patches are scaled to the same size as the local image patch.

C. Linear Self-Attention Mechanism

This section first introduces the standard self-attention mechanism, but the complexity of the standard self-attention mechanism is $O(N^2)$. In order to reduce the complexity, we deduce the standard self-attention into linear self-attention and introduce the derivation process of the linear self-attention mechanism in the later part.

Assume N and D_X represent the length and number of channels of the input sequence, where $N = w \times h$ (w and h represent the width and length of the input image). Given the eigenvector $X = [x_1, \ldots, x_N] \in R^{N\times D_x}$, self-attention uses three weight matrices $M_q \in R^{D_x \times D_k}$, $M_k \in R^{D_x \times D_k}$, and $M_v \in R^{D_x \times D_v}$ to generate Q, K, and V.

$$\begin{aligned} Q &= XM_q \\ K &= XM_k \\ V &= XM_v. \end{aligned} \tag{1}$$

The dimensions of Q and K are the same.

The self-attention function (Figs. 2 and 3) is used to calculate the weighted average of each location and other location features. The weight is the similarity between Q and V. Output $A(x)$ at all positions is calculated as follows:

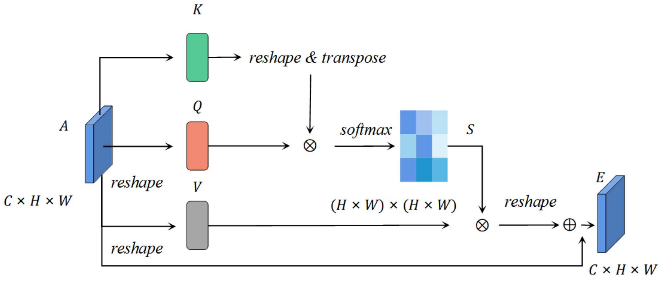


Fig. 2. Standard PAM.

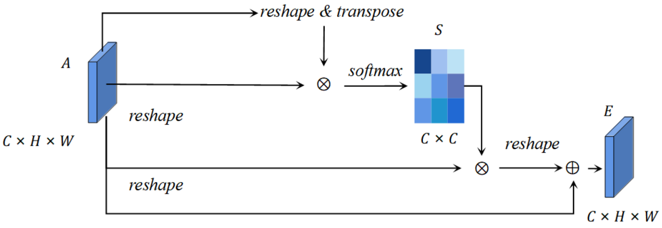


Fig. 3. Standard CAM.

$$A(x) = \text{softmax}\left(\frac{QK^T}{\sqrt{D}}\right)V. \tag{2}$$

The softmax function is used to evaluate the similarity between Q and K, and the softmax function is replaced by sim to represent the similarity function. Eq. (2) can be expressed as follows:

$$A(x) = \text{sim}(QK^T)V \tag{3}$$

where $\text{sim}(QK^T)$ represents the similarity between any positions. From the above equation, we can easily find that $Q \in R^{N\times D_k}$ and $K^T \in R^{D_k \times N}$. Therefore, the result obtained by multiplying Q and K is $R^{N\times N}$, and the time and space complexity are $O(N^2)$. The larger the feature map is, the larger the memory consumption of GPU is and the longer the calculation time is. We solve this problem by improving the sim function.

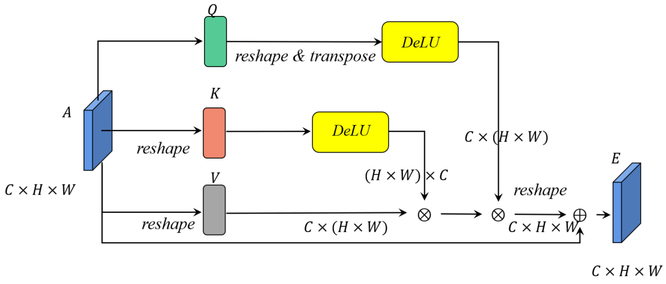


Fig. 4. Linear attention mechanism.

For line i, the corresponding definition is as follows:

$$A_i(X) = \frac{\sum_{j=1}^{N} e^{Q_i^T K_j} V_j}{\sum_{j=1}^{N} e^{Q_i^T K_j}}. \tag{4}$$

Equation (4) can be replaced by any normalization function and rewritten as follows:

$$A_i(X) = \frac{\sum_{j=1}^{N} \text{sim}\left(Q_i, K_j\right) V_j}{\sum_{j=1}^{N} \text{sim}\left(Q_i, K_j\right)}. \tag{5}$$

Here, a constraint is added to $\text{sim}(\cdot)$. For the self-attention function of equation (5), the value of $\text{sim}(\cdot)$ needs to be non negative. When the normalization function is softmax, $\text{sim}(Q_i, K_j) = e^{Q_i^T K_j}$. The generalization of $\text{sim}(Q_i, K_j)$ shows that $\text{sim}(Q_i, K_j) = \phi(Q_i)^T \phi(K_j)$. Bring it into equation (5) to obtain

$$A_i(X) = \frac{\sum_{j=1}^{N} \phi\left(Q_i\right)^T \phi\left(K_j\right) V_j}{\sum_{j=1}^{N} \phi\left(Q_i\right)^T \phi\left(K_j\right)}. \tag{6}$$

In (6), the summation symbol variable is j, which will not affect $\phi(Q_i)^T$. Therefore, $\phi(Q_i)^T$ is proposed to obtain

$$A_i(X) = \frac{\phi\left(Q_i\right)^T \sum_{j=1}^{N} \phi\left(K_j\right) V_j^T}{\phi\left(Q_i\right)^T \sum_{j=1}^{N} \phi\left(K_j\right)}. \tag{7}$$

The molecular vector of (7) is obtained

$$A_i(x) = \frac{\phi(Q)^T(\phi(K)V^T)}{\phi\left(Q_i\right)^T \sum_{j=1}^{N} \phi\left(K_j\right)}. \tag{8}$$

We construct a function close to $ReLU$ as the kernel function, which is named as derivable linear units ($DeLU$). The linear self-attention is shown in Fig. 4. The $DeLU$ function is as follows:

$$y_i = \begin{cases} a_i x_i + 1, & \text{if } x_i \geq 0 \\ e^{a_i x_i}, & \text{if } x_i < 0 \end{cases} \tag{9}$$

where a_i is a fixed number ($a_i \in (1, +\infty)$).

$ReLU$ function is not selected here, because the gradient of $ReLU$ function equals zero when it is negative, and the gradient does not exist when it equals zero. Therefore, the weight value will fluctuate violently during training. The $DeLU$ function solves the problem of $ReLU$, which is differentiable at 0, and the function curve is close to $ReLU$. Therefore, the similarity function can be expressed as follows:

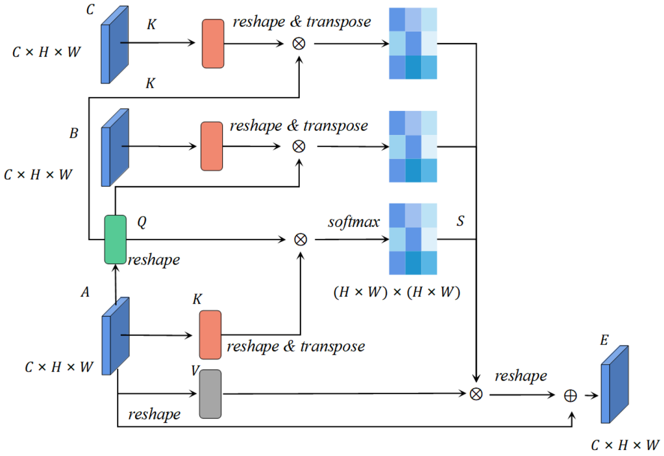


Fig. 5. Position context multiscale attention mechanism.

$$\text{sim}\left(Q_i, K_j\right) = F\left(\text{DeLU}\left(Q_i\right) \text{DeLU}\left(K_j\right)\right). \tag{10}$$

Equation (7) can be rewritten as follows:

$$A_i(X) = \frac{\text{DeLU}\left(Q_i\right)^T \sum_{j=1}^{N} \text{DeLU}\left(K_j\right) V_j^{\ T}}{\text{DeLU}\left(Q_i\right)^T \sum_{j=1}^{N} \text{DeLU}\left(K_j\right)}. \tag{11}$$

Equation (11) can be vectorized as follows:

$$A(x) = \frac{\text{DeLU}(Q)^T \sum_{j=1}^{N} \text{DeLU}\left(K_j\right) V_j^{\ T}}{\text{DeLU}(Q)^T \sum_{j=1}^{N} \text{DeLU}\left(K_j\right)}. \tag{12}$$

The time and space complexity of the linear attention mechanism obtained by equation (11) is $O(N)$, because it can be calculated once when calculating $\sum_{j=1}^{N} \text{DeLU}(K_j) V_j$ and $\text{DeLU}(K_j)^T$, and can be reused in the query vector Q.

D. Multiscale Context With Linear Self-Attention

The original self-attention mechanism calculates the similarity between the global pixels of the image. The pixel resolution of HRRSI is enormous, which will occupy much more memory when calculating the similarity feature map. Therefore, this article proposes a multiscale context self-attention mechanism that requires multiple feature maps as input, including feature maps of local patch and mutliscale context patch. The local patch uses a small segmented image. The objects on edge may be segmented into multiple patches, so the role of the multiscale context patch is to supplement these features and minimize the number of misclassification of local patch segmentation.

Figs. 5 and 6 show that the local image patch and the multiscale context patch are input into the backbone network to obtain the local feature map A and the multiscale context feature map B and C. The three feature maps obtain the output results through the context self-attention mechanism. When the HRRSI is cropped into multiple local image patches, the same object's features are cropped into multiple image patches. The lack of object features may lead to the wrong segmentation. The multiscale context image patches can enhance the feature

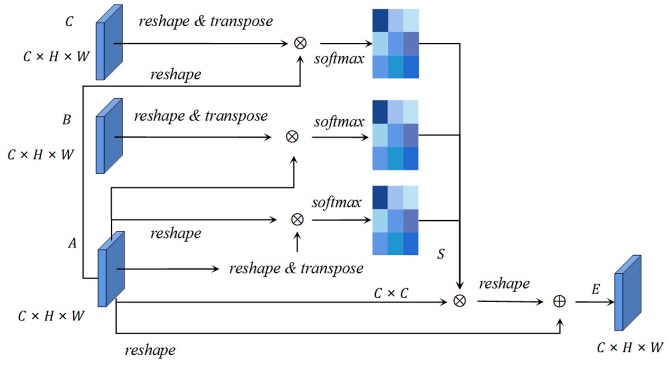


Fig. 6. Channel context multiscale attention mechanism.

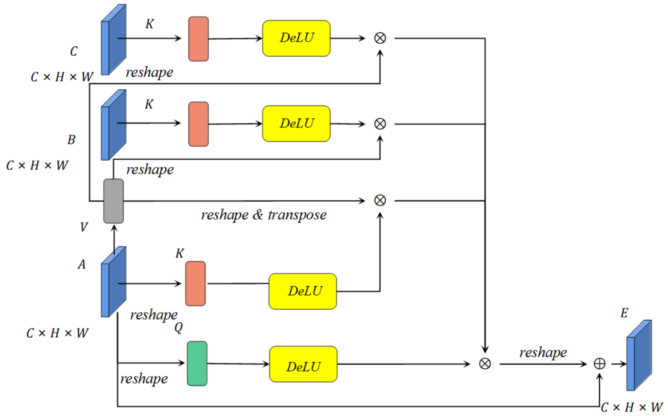


Fig. 7. Position context multiscale linear attention mechanism.

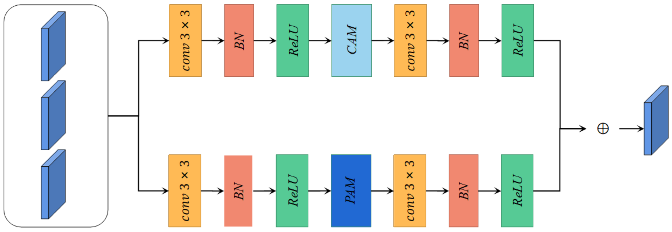


Fig. 8. Position and channel linear self-attention module.

TABLE I
EXPERIMENTAL RESULTS OF POTSDAM DATASET

Method	PA	CPA	Kappa	MIoU	FWIoU
U-Net [16]	85.59	80.83	81.03	70.24	75.19
DeeplabV3+ [28]	88.19	86.03	84.51	74.55	79.33
DANet [18]	89.42	86.08	86.1	76.98	81.2
PSPNet [19]	88.34	84.16	84.62	75.1	79.45
RefineNet [20]	88.27	84.07	84.38	74.44	79.35
SegNet [21]	86.77	83.06	82.5	72.31	77.11
ACFNet [22]	86.84	82.23	82.46	74.23	77.07
OCRNet [23]	88.36	86.16	84.46	74.56	79.59
MSCSANet(Ours)	**90.75**	**86.92**	**87.84**	**78.91**	**83.39**

The bold data represents the optimal result of each column.

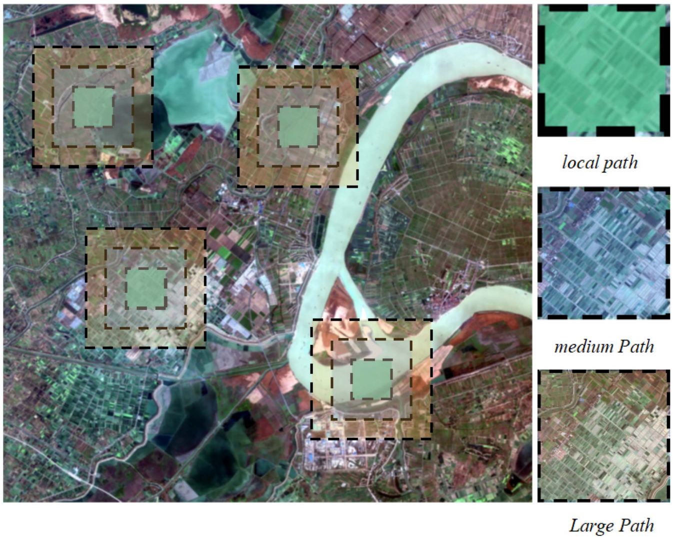


Fig. 9. Schematic diagram of multiscale context image patch acquisition.

of local image patches. Assuming that there are M feature maps of multiscale context patch are used, the equation of the context self-attention mechanism is as follows:

$$A(x) = \sum_{i=1}^{M} \text{softmax}\left(\frac{Q_A K_i^T}{\sqrt{D}}\right) V_A. \tag{13}$$

Taking the local image L as the query vector, the similarity between the key vector of B and C and the query vector Q is calculated, respectively, and then multiplied by the value vector of A. In this way, the B and C vectors will affect the feature map of A and supplement the missing feature.

The computational complexity of Fig. 5 is $O(N^2)$, so we use the linear self-attention kernel function $DeLU$ instead of softmax. As shown in Fig. 7, the time complexity and spatial complexity are reduced to $O(N)$. After replacing the kernel function, the equation of position context linear self-attention is defined as follows:

$$A(x) = \frac{\text{DeLU}(Q)^T \sum_{i=1}^{M} \sum_{j=1}^{N} \text{DeLU}\left(K_j^i\right) V_j^T}{\text{DeLU}(Q)^T \sum_{i=1}^{M} \sum_{j=1}^{N} \text{DeLU}\left(K_j^i\right)}. \tag{14}$$

Position attention mechanism (PAM) and channel attention mechanism (CAM) are used to model the long-term dependence of location and channel, respectively. An attention patch is designed to improve the extraction ability of each layer of the feature map (see Fig. 8).

IV. EXPERIMENTS

This chapter will describe the datasets, parameter settings, training parameter settings, and experimental results on each dataset.

A. Dataset

In order to demonstrate the effectiveness of the proposed model, the ISPRS Potsdam dataset and the GID is used. In this

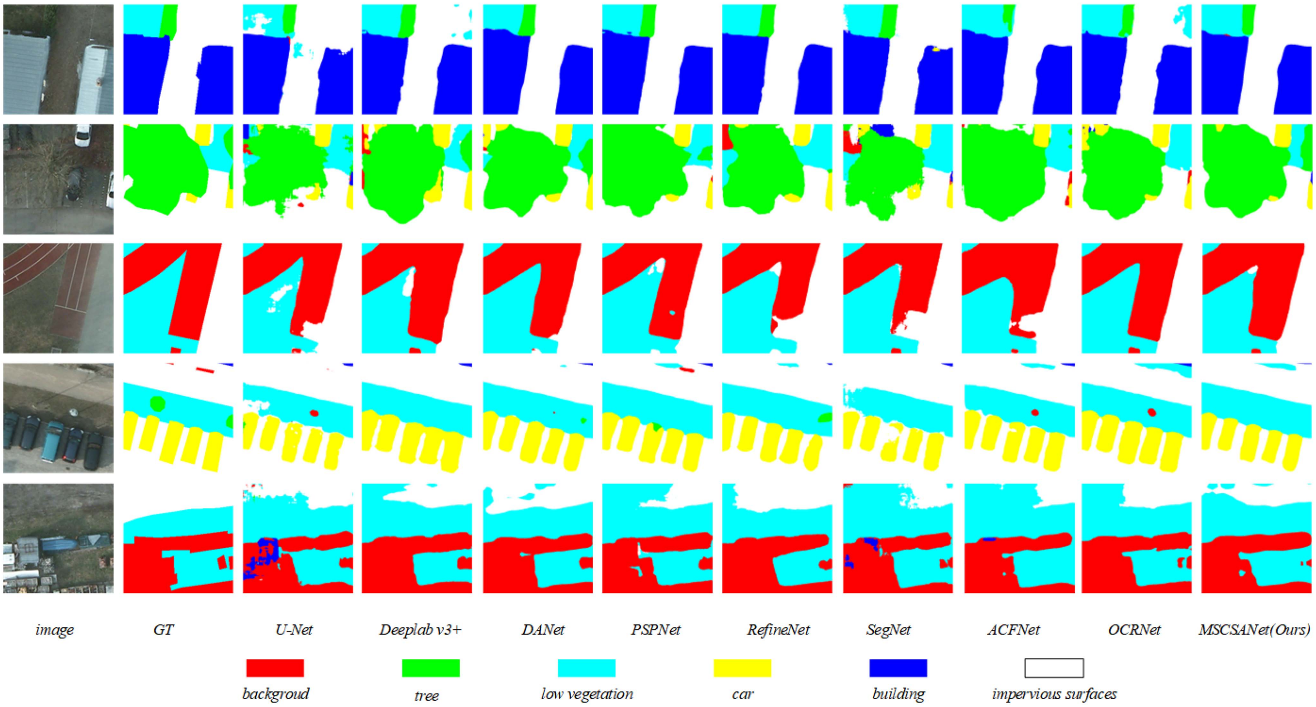


Fig. 10. Experimental effect of the benchmark models on Potsdam dataset.

TABLE II
MIoU Experimental Results of Each Kind on Potsdam Dataset

Method	Impervious surfaces	Backgroud	Tree	Low vegetation	Car	Building	MIoU
U-Net [16]	75.68	45.7	68.98	71.39	73.48	86.24	70.24
DeeplabV3+ [28]	80.08	55.8	71.74	74.93	74.98	91.21	74.91
DANet [18]	82.29	**60.02**	73.05	76.24	77.98	92.27	77.08
PSPNet [19]	82.71	56.36	72.45	76.12	79.31	91.88	76.47
RefineNet [20]	80.62	54.92	71.50	74.43	74.57	91.04	74.51
SegNet [21]	78.57	51.77	70.01	73.18	74.01	88.89	72.76
ACFNet [22]	80.35	53.37	70.92	74.97	75.90	89.83	74.23
OCRNet [23]	81.11	54.29	71.29	74.91	74.74	90.94	74.56
MSCSANet(Ours)	**87.49**	58.40	**75.09**	**77.39**	**81.15**	**93.63**	**78.91**

The bold data represents the optimal result of each column.

TABLE III
Experimental Results on GID Dataset

Method	PA	CPA	Kappa	MIoU	FWIoU
U-Net [16]	80.05	69.71	74.46	57.01	67.05
DeeplabV3+ [28]	80.57	69.98	75.50	59.35	67.79
DANet [18]	82.78	75.66	79.13	65.53	70.90
PSPNet [19]	81.56	75.45	77.17	63.08	69.23
RefineNet [20]	81.59	74.14	77.59	63.09	69.18
SegNet [21]	81.10	69.62	75.98	58.72	68.54
ACFNet [22]	82.64	76.79	77.57	64.58	70.84
OCRNet [23]	80.08	79.21	75.45	62.96	67.16
MSCSANet(Ours)	**83.76**	**80.30**	**81.00**	**67.09**	**72.51**

The bold data represents the optimal result of each column.

section, we first briefly describe the data set and introduce the details of the experiment.

1) Potsdam: ISPRS Potsdam dataset [55] contains 38 high-resolution images of 6000 × 6000 pixels, of which 24 are training sets and 14 are test sets. The dataset provides four channels: near-infrared, red, green, and blue. RGB three-channel data is selected in the experiment. The experiment only uses images for training, which are RGB three-channel images. We cut the image into patches of 256 × 256 and obtain large and medium patches centered on each patch. The size of medium and large patches are 512 × 512 and 768 × 768, respectively. The large and middle patches are scaled to the same size as the original image. Moreover, we increase the amount of data by rotating, resizing, flipping, and adding random noise (see Fig. 9).

2) GID: GID [56] is a large-scale high-resolution remote sensing image dataset taken by Gaofen-2. The dataset includes the large-scale classification set (GID-5) and the fine land cover set (GID-15), which contains ten images with a resolution of 7200 × 6800 pixels. GID dataset has the characteristics of a wide distribution of land cover information and is close to the natural distribution of ground objects. In terms of data processing, we cut the dataset into 256 × 256 image patches and discarded the redundant parts. We obtained a total of 7280 samples,

TABLE IV
MIoU Experimental Results of Each Kind on GID Dataset

Method	U-Net	DeeplabV3+	DANet	PSPNet	RefineNet	SegNet	ACFNet	OCRNet	MSCSANet(Ours)
Backgroud	65.8	66.39	68.67	67.04	66.89	67.11	68.84	62.57	**69.55**
Irrigated land	77.19	78.69	79.93	78.74	78.19	77.68	79.84	77.99	**81.13**
Traffic land	57.83	53.99	59.92	55.16	56.32	58.37	**61.5**	57.41	60.74
Industrial land	52.35	56.53	**57.06**	56.05	54.53	53.08	55.31	54.42	56.28
Rural residential	52.36	51.07	**55.43**	53.81	51.99	49.29	54.72	50.58	51.92
Pond	60.51	59.16	66.94	60.39	65.68	60.88	65.8	63.68	**67.8**
River	78.03	82.34	91.23	88.77	90.58	88.86	92.39	89.78	**92.48**
Garden land	24.1	31.87	34.94	28.87	33.82	25.77	32.67	38.25	**39.83**
Dry cropland	59.72	57.43	64.36	64.31	63.37	60.55	59.99	59.56	**69.75**
Urban residential	63.91	66.51	66.75	67.6	66.32	65.36	67.88	66.76	**68.06**
Arbor forest	75.68	71.45	75.19	74.97	74.99	73.51	**76.42**	73.98	76.14
Shrub land	0	32.74	40.45	44.22	34.85	0	43.18	37.93	**45.23**
Lake	63.24	64.97	81.71	79.7	80.25	75.37	83.57	80	**86.27**
Paddy field	67.21	56.82	74.89	69.98	72.63	71.44	71.04	69.48	**75.98**
Natural meadow	58.73	60.74	63.64	62.18	62.63	63.88	59.12	64.51	**66.75**
Artificial meadow	55.51	64.65	**67.43**	57.82	56.35	50	60.97	60.49	65.46
MIoU	57.01	59.35	65.53	63.1	63.09	58.82	64.58	62.96	**67.09**

The bold data represents the optimal result of each column.

and the large patch and medium patch centered on each patch, where the size of the medium patch is 512×512, and the size of the large patch is 768×768. Scale the large and medium patches to the same size as the original image. We first scrambled the cut dataset and then made the training set, validation set, and test set in the ratio of $6:2:2$.

B. Evaluation Metric

Pixel accuracy (PA), mean pixel accuracy (MPA), kappa coefficient (K), and mean intersection over union (MIoU) were used to evaluate the performance of MSCSANet on two datasets. PA is not sensitive to a few classes in unbalanced datasets, while MIoU is excessively sensitive to a few classes. Therefore, we use frequency weighted intersection over union (FWIoU) to evaluate the performance further. PA, MPA, K, MIoU, and FWIoU metrics are calculated as follows:

PA is the proportion of correctly classified pixels of all total image pixels, as shown as follows:

$$\mathrm{PA} = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k}\sum_{j=0}^{k} py}. \tag{15}$$

The mean PA is the proportion of correctly classified pixels of each class in all pixels of this class. Take the average value of the results, as shown as follows:

$$\mathrm{MPA} = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{i=0}^{k} p_{ij}}. \tag{16}$$

MIoU calculates the ratio of the intersection and union of two sets of actual and predicted values. See equation as follows:

$$\mathrm{mIoU} = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}}. \tag{17}$$

Kappa coefficient is an index used for consistency tests and can also measure the effect of classification. Kappa's calculation result is between - 1 and 1, but it usually falls between 0 and 1. See equation as follows:

$$K = \frac{p_o - p_e}{1 - p_e}. \tag{18}$$

FWIoU sets the weight according to the frequency of each class, and the weight is multiplied by the intersection over union (IoU) of each class and summed. It is formulated as follows:

$$\mathrm{FWIoU} = \frac{\sum_{i=0}^{k}\frac{P_{ii}}{\sum_{j=0}^{k} P_{ij}+\sum_{j=0}^{k} P_{ji}-p_{ii}'}}{\sum_{i=0}^{k}\sum_{j=0}^{k} P_{ij}}. \tag{19}$$

C. Training Configuration

In order to comprehensively evaluate the performance of MSCSANet, contrast models such as U-Net [16], deeplabV3+[28], RefineNet [20], PSPNet [19], ACFNet [22], SegNet [21], DANet [18], and OCRNet [23] are used for comparison. To ensure the fairness of each model, all models use ResNet50 as the backbone. All models are implemented by PyTorch. Moreover, an SGD optimizer is used, and the learning rate is set to 0.003 and attenuated by a poly polynomial. For the loss function setting/selection, the cross-entropy loss function is selected as the quantitative evaluation method. (9) is set to 10. The batch size is eight, and the training algebra is set to 300. All experiments were conducted on the Tianhe-I platform. NVIDIA Tesla V100 GPU is used for experiments on the Tianhe-I platform, and the GPU memory is 16 GB.

We complete the comparison experiments of each model on two datasets to compare the performance of different semantic segmentation models. In addition, ablation experiments are designed to verify the significance of the selected context size and number.

D. Experimental Results of Potsdam Dataset

We compare the experimental results of our proposed MSCSANet network with some other state-of-the-art networks (U-Net, DeeplabV3+, DANet, PSPNet, RefineNet, SegNet, ACFNet, OCRNet) over the same dataset. Moreover, we use the same dataset for comparison. We set the same parameters on

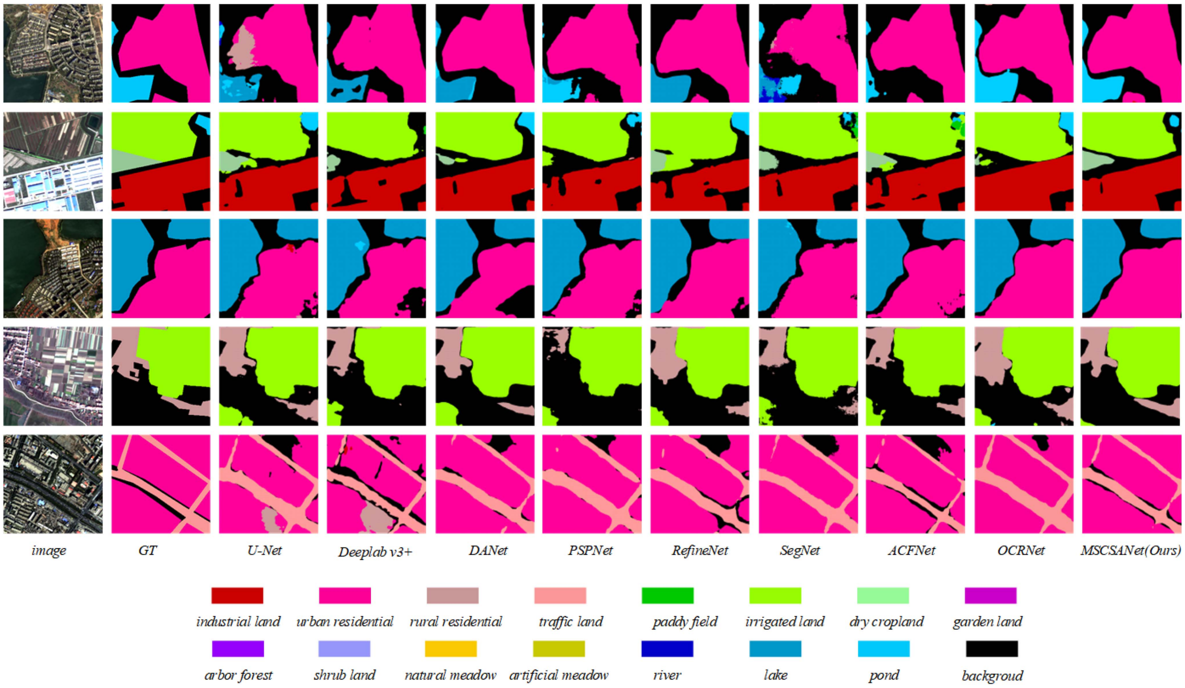


Fig. 11. Experimental effect of benchmark model on GID dataset.

TABLE V
EFFICACY OF CONTEXTS FOR LOCAL SEGMENTATION

Context			MIoU	
Local	Medium	Large	GID	Potsdam
✓			65.76	77.53
✓	✓		65.39	76.8
✓		✓	65.17	76.9
✓	✓	✓	**67.07**	**78.87**

The bold data represents the optimal result of each column.

the Potsdam dataset. The experimental results of the abovecentralized model are shown in Tables I and II.

As shown in Table I, the experimental results of U-Net and SegNet are poor, with MIoU of 70.24% and 72.31%, respectively. The main reason is that the original U-Net network is too simple to extract deep features, while SegNet does not consider the spatial context of the image. Although the Deeplabv3+ network combines multiscale information, the segmentation accuracy and MIoU of each feature class are not high. Due to the introduction of a dual attention module, DANet improves the segmentation accuracy of various ground objects. Compared with several other advanced networks, DANet's various evaluation indicators are the best, and its MIoU reaches 76.98%, which is 2.43% higher than Deeplabv3's 74.55%. PSPNet introduced the pyramid pooling module and added an auxiliary loss function, and the effect was quite good. Its MIoU reached 75.10%. The remaining experimental results of RefineNet, ACFNet, and OCRNet are relatively close, with MIoU of 74.44%, 74.23%, and 74.56%, respectively. RefineNet uses a residual convolution module, multiresolution fusion module, and chain residual pooling module, which is helpful for semantic segmentation of HRRSI. ACFNet utilizes the relationship between pixels in the same class to achieve class-level context, consistent with high-resolution images. OCRNet has also achieved good results by enhancing its pixel representation with object region representation. Using multiscale contextual image patches and a double-ended linear self-attention module, our model achieves the best experimental results among these models, with MIoU reaching 78.91%, which is still a 1.93% improvement compared to the best DANet above.

The visual structure of these models is shown in Fig. 10. We select five typical images from the prediction results for comparison. In scene 1, U-Net, deeplabV3, and OCRNet easily confuse low vegetation and impervious surfaces because the features of these two types of ground objects are relatively close, and our proposed model has an excellent ability to distinguish these easily confused features. In scene 2, U-Net, DeeplabV3+, RefineNet, SegNet, and ACFNet are challenging to recognize the features of the image boundary. However, by using multiscale context features, our model can well solve the problem of feature loss caused by image cutting. In scene 3, each class is also easy to confuse because the background and low vegetation have similar features. Only DANet, PSPNet, OCRNet, and our model can be well distinguished. In scene 4, some models have poor discrimination of car contours, while the models using the attention mechanism have better effects. In the image with complex features, such as scene 5, all models do not get outstanding results because it is complicated to distinguish between low vegetation and impervious surfaces.

Overall, our model has a good recognition effect in each scene. Especially for boundary regions and easily confused regions, our model can use multiscale information to complement the features. Second, small objects such as cars have few available

TABLE VI
SCALES OF CONTEXTS FOR LOCAL SEGMENTATION

GID				Potsdam			
Contest size			MIoU	Contest size			MIoU
256×256	768×768	1280×1280	65.6	**256×256**	**768×768**	**1280×1280**	**78.91**
256×256	512×512	1024×1024	64.21	256×256	512×512	1024×1024	78.82
256×256	**512×512**	**768×768**	**67.09**	256×256	768×768	1024×1024	78.48
256×256	768×768	1024×1024	63.23	256×256	512×512	768×768	78.87
256×256	512×512	1280×1280	64.29	256×256	512×512	1280×1280	78.9
256×256	1024×1024	1280×1280	64.46	256×256	1024×1024	1280×1280	78.27

The bold data represents the optimal result of each column.

features compared with large objects, and small objects adjacent to the aggregation area are difficult to distinguish. Benefiting from the attention mechanism, our model also has advantages in small object detection.

E. Experimental Results on GID Dataset

Unlike the ISPRS Potsdam dataset, the GID dataset contains 15 land-use types, which makes the segmentation of the GID dataset more difficult. Similar to the experiment in the previous section, we prove the superiority of our model by comparing our model with other advanced models.

Table III shows that the GID dataset cannot reach the accuracy of the Potsdam dataset because the GID dataset has many types of objects and more complex data. In Table III, our model achieves the best result on each evaluation index. Comparing the optimal results of each model in each evaluation index, MSCSANet improves PA by 0.98%, CPA by 1.09%, Kappa coefficient by 1.87%, MIoU by 1.56%, and FWIoU by 1.61%.

In order to compare the segmentation effects of these different types of networks, we selected five different scenes from the dataset and marked them as scene 1 to scene 5 in Fig. 11. scene 1 can be used to compare the segmentation effects of different networks on images with large differences in size characteristics because it includes most urban residential areas and a small part of lakes. Scene 2 can test the segmentation effect of different networks in complex scenes because it includes complex scenes such as irrigation land, industrial land, rural residential areas, and ponds. The main body of scene 3 is industrial land adjacent to a lake, which is a typical urban area. Scene 4 is a scene in a rural area consisting of irrigated land and rural residential. Scene 5 is a scene with a large amount of traffic land in the city, the roads are slender, and it is challenging to identify the wide roads. The feature of these different scenes are easy to confuse, all of which are very suitable for verifying the segmentation effect of different networks. Using these five scenes, we can compare the segmentation performance of each network on high-resolution images from different perspectives. These scenes and different feature types are shown in Fig. 11.

In scene 1 of Fig. 11, U-Net, SegNet, Deeplabv3+, PSPNet, and ACFNet have poor segmentation effects on lakes, and lakes are not entirely detected. However, DAnet, OCRNet, and MSCSANet have enhanced the long-term dependent feature due to the introduction of the attention module. Hence, the segmentation of large patches of land such as lakes is better. In scene 2, there are many types of ground objects. These models have better segmentation effects on industrial land and irrigated land with prominent feature, but the segmentation effects on rural residential and pond are pretty different. Only ACFNet and MSCSANet have a relatively complete segmentation of rural residential. Deeplabv3+, PSPNet, SegNet, and ACFNet have poor segmentation effects on the pond. In scene 3, only SegNet, OCRNet, and MSCSANet have better segmentation performance for urban residential in the lower right corner than other models. In scene 4, rural residential and irrigated land have similar features, which is very easy to confuse. In these models, U-Net, Deeplabv3+, DANet, RefineNet, and MSCSANet can better distinguish between rural residential and irrigated land because these models use context information. In scene 5, OCRNet and MSCSANet classify roads better. In the segmentation results of other models, roads will be disconnected. This is mainly because our model introduces an attention module, which can obtain long-distance dependencies.

Furthermore, as shown in Table IV, we show the MIoU for each feature class. U-Net and SegNet in classifying Shrub land are 0, mainly because there are few samples of Shrub land, so it is difficult for these two models to learn the feature of shrub land during training. Deeplabv3+ combines multiscale features, but the accuracy of ground object segmentation is not satisfactory. In contrast, DANet improves the segmentation accuracy of various ground feature classes. The segmentation accuracy of industrial land and artificial grassland is the highest because DANet uses the self-attention mechanism. The MIoU of our proposed model is as high as 67.09%, which is due to the combination of multiscale context features and self-attention mechanism.

F. Ablation Studies

In this section, we will prove the superiority of the model design through the experimental results of different parameter and network structure settings.

Efficacy of contexts. We first demonstrate the effectiveness of context through this experiments. We use the following four experimental schemes for comparison: local image patch, local image patch and mesoscale image patch, local image patch and large-scale patch, and three-scale image patches. The final experimental results are shown in Table V. Overall, multiscale context input improves the segmentation accuracy of the dataset. Compared with only using the linear self-attention mechanism, the accuracy of the GID dataset is improved by 1.33%, and

1.34% improves that of the Potsdam dataset. Because the probability of the local image patch's contour is part of some objects' semantic information, it is difficult to infer the semantics of objects close to the edge even though the autocorrelation of attention mechanism. Therefore, the use of medium and large-scale image patches can provide context information, which will be conducive to segmentation. However, based on experiments, we find that only using a certain scale of context image patch does not necessarily produce a better experimental effect. The experimental effect of using only a multiscale image patch on two datasets is worse than that of using local image patches. Therefore, using the complementary information of multiple contextual image patches with different scales can bring better experimental results.

Context scale. This part studies the impact of different context scales on the performance of the model. As shown in Table VI, we use multiple sets of context scales to study the segmentation effect on GID and Potsdam datasets. Theoretically, if the context scale is close to the local image patch size, it will cause information redundancy and will not bring much performance improvement. Therefore, in order to fully study the influence of different scales of context, we choose the multiple of local image patches as the context scale. From the experimental results, the optimal context of GID dataset is 256×256, 512×512 and 768×768, and the optimal context of Potsdam dataset is 256×256, 768×768 and 1280×1280. The data's features and position structures are different, leading to the inconsistency of their optimal context.

V. Conclusion

In this article, we propose a model that integrates contextual multiscale and linear self-attention. The contextual multiscale input convolution model can fully obtain the coarse-grained and fine-grained feature information of different image patches. The extracted large-scale image features will supplement the missing feature of local image patches. The multiscale linear self-attention mechanism takes the output feature of the multiscale convolution as the input. The global semantic correlation of multiscale input images is modeled to reduce the impact of the lack of target features at different scales caused by image segmentation. The prediction results of our MSCSANet model on Potsdam and GID datasets are competitive with other advanced models. The self-attention mechanism still has great potential for many applications in remote sensing.

In the future, we will study the self-attention mechanism in solving the problem of fuzzy edges of different categories of targets in semantic segmentation.

References

[1] C. Zhang et al., "Joint deep learning for land cover and land use classification," *Remote Sens. Environ.*, vol. 221, pp. 173–187, 2019.

[2] M. Volpi and D. Tuia, "Dense semantic labeling of subdecimeter resolution images with convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 881–893, Feb. 2017.

[3] J. Zhang, L. Feng, and F. Yao, "Improved maize cultivated area estimation over a large scale combining modis–evi time series data and crop phenological information," *ISPRS J. Photogrammetry Remote Sens.*, vol. 94, pp. 102–113, 2014.

[4] D. Sulla-Menashe, J. M. Gray, S. P. Abercrombie, and M. A. Friedl, "Hierarchical mapping of annual global land cover 2001 to present: The modis collection 6 land cover product," *Remote Sens. Environ.*, vol. 222, pp. 183–194, 2019.

[5] C. Zhang, P. A. Harrison, X. Pan, H. Li, I. Sargent, and P. M. Atkinson, "Scale sequence joint deep learning (SS-JDL) for land use and land cover classification," *Remote Sens. Environ.*, vol. 237, 2020, Art. no. 111593.

[6] Z. Zhang, C. Tian, H. X. Bai, Z. Jiao, and X. Tian, "Discriminative error prediction network for semi-supervised colon gland segmentation," *Med. Image Anal.*, vol. 79, 2022, Art. no. 102458.

[7] Z. Zhang et al., "Collaborative boundary-aware context encoding networks for error map prediction," *Pattern Recognit.*, vol. 125, 2022, Art. no. 108515.

[8] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[9] W. Chen, Z. Jiang, Z. Wang, K. Cui, and X. Qian, "Collaborative global-local networks for memory-efficient segmentation of ultra-high resolution images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8924–8933.

[10] A. Vaswani et al., "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, vol. 30, pp. 6000–6010, 2017.

[11] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7794–7803.

[12] R. Child, S. Gray, A. Radford, and I. Sutskever, "Generating long sequences with sparse transformers," 2019, *arXiv:1904.10509*.

[13] N. Kitaev, Ł. Kaiser, and A. Levskaya, "Reformer: The efficient transformer," 2020, *arXiv:2001.04451*.

[14] A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret, "Transformers are RNNS: Fast autoregressive transformers with linear attention," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 5156–5165.

[15] Z. Shen, M. Zhang, H. Zhao, S. Yi, and H. Li, "Efficient attention: Attention with linear complexities," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2021, pp. 3531–3539.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Interv.*, 2015, pp. 234–241.

[17] 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.*, 2018, pp. 801–818.

[18] J. Fu et al., "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3146–3154.

[19] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2881–2890.

[20] G. Lin, A. Milan, C. Shen, and I. Reid, "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1925–1934.

[21] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[22] F. Zhang et al., "ACFNet: Attentional class feature network for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6798–6807.

[23] Y. Yuan, X. Chen, and J. Wang, "Object-contextual representations for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 173–190.

[24] H. K. Cheng, J. Chung, Y.-W. Tai, and C.-K. Tang, "Cascadepsp: Toward class-agnostic and very high-resolution segmentation via global and local refinement," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8890–8899.

[25] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," 2016, *arXiv:1606.02147*.

[26] W. Liu, A. Rabinovich, and A. C. Berg, "Parsenet: Looking wider to see better," 2015, *arXiv:1506.04579*.

[27] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," *Comput. Sci.*, vol. 4, pp. 357–361, 2014.

[28] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[29] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.

[30] H. Zhang et al., "Context encoding for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7151–7160.

[31] H. Wu, J. Zhang, K. Huang, K. Liang, and Y. Yu, "Fastfcn: Rethinking dilated convolution in the backbone for semantic segmentation," 2019, *arXiv:1903.11816*.

[32] C. Liu et al., "Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 82–92.

[33] S. Qiao, L.-C. Chen, and A. Yuille, "Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10213–10224.

[34] Q. Li, W. Yang, W. Liu, Y. Yu, and S. He, "From contexts to locality: Ultra-high resolution image segmentation via locality-aware contextual correlation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 7252–7261.

[35] D. Mazzini, "Guided upsampling network for real-time semantic segmentation," 2018, *arXiv:1807.07466*.

[36] R. P. Poudel, U. Bonde, S. Liwicki, and C. Zach, "Contextnet: Exploring context and detail for semantic segmentation in real-time," 2018, *arXiv:1805.04554*.

[37] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 325–341.

[38] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*.

[39] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[40] Q.-L. Zhang and Y.-B. Yang, "Sa-net: Shuffle attention for deep convolutional neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 2235–2239.

[41] F. Li et al., "PSANet: Pyramid splitting and aggregation network for 3D object detection in point cloud," *Sensors*, vol. 21, no. 1, 2021, Art. no. 136.

[42] Y. Yuan, L. Huang, J. Guo, C. Zhang, X. Chen, and J. Wang, "Ocnet: Object context for semantic segmentation," *Int. J. Comput. Vis.*, vol. 129, no. 8, pp. 2375–2398, 2021.

[43] A. Roy, M. Saffar, A. Vaswani, and D. Grangier, "Efficient content-based sparse attention with routing transformers," *Trans. Assoc. Comput. Linguistics*, vol. 9, pp. 53–68, 2021.

[44] X. Li, Y. Meng, M. Zhou, Q. Han, F. Wu, and J. Li, "Sac: Accelerating and structuring self-attention via sparse adaptive connection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 16997–17008.

[45] Y. Tay, D. Bahri, L. Yang, D. Metzler, and D.-C. Juan, "Sparse sinkhorn attention," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 9438–9447.

[46] A. Vyas, A. Katharopoulos, and F. Fleuret, "Fast transformers with clustered attention," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 21665–21674.

[47] H. Zhou et al., "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 11106–11115.

[48] P. J. Liu et al., "Generating wikipedia by summarizing long sequences," 2018, *arXiv:1801.10198*.

[49] J. Lee, Y. Lee, J. Kim, A. Kosiorek, S. Choi, and Y. W. Teh, "Set transformer: A framework for attention-based permutation-invariant neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 3744–3753.

[50] S. Wang, B. Z. Li, M. Khabsa, H. Fang, and H. Ma, "Linformer: Self-attention with linear complexity," 2020, *arXiv:2006.04768*.

[51] Q. Guo, X. Qiu, X. Xue, and Z. Zhang, "Low-rank and locality constrained self-attention for sequence modeling," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 12, pp. 2213–2222, Dec. 2019.

[52] W. Chen, B. Chen, Y. Liu, Q. Zhao, and M. Zhou, "Switching poisson gamma dynamical systems," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, C. Bessiere, Ed. 2020, vol. 7, pp. 2029–2036, doi: 10.24963/ijcai.2020/281.

[53] Y. Xiong et al., "Nyströmformer: A nystöm-based algorithm for approximating self-attention," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, 2021, pp. 14138–14148.

[54] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[55] F. Rottensteiner, G. Sohn, M. Gerke, and J. D. Wegner, "ISPRS semantic labeling contest," *ISPRS: Leopoldshöhe, Germany*, vol. 1, 2014, Art. no. 4.

[56] X.-Y. Tong et al., "Land-cover classification with high-resolution remote sensing images using transferable deep models," *Remote Sens. Environ.*, vol. 237, 2020, Art. no. 111322.

Peng Yin received the B.S. degree in spatial information and digital technology from the China University of Geosciences, Wuhan, China, in 2020. He is currently working toward the M.S. degree in computer science and technology with the School of Computer Science and Technology, China University of Geosciences, Wuhan, China.

His research interests include remote sensing and deep learning.

Dongmei Zhang received the B.S. and M.S. degrees in computer application and the Ph.D. degree in earth exploration and information technology from the China University of Geosciences, Wuhan, China, in 1994, 1999, and 2007, respectively.

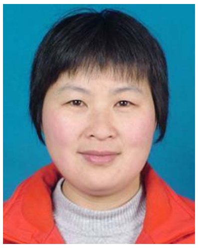
She is currently a Professor with the School of Computer Science, China University of Geosciences. In recent years, she has presided over more than ten provincial and ministerial projects, including the National Natural Science Foundation of China, the 13th Five-Year National Key Science and Technology Project, the local survey project of the China Geological Survey, and the Natural Resources Department of Hubei Province. Her research interests include in-depth learning, big geoscience data, and landslide disaster prediction.

Wei Han received the B.S. degree in network engineering, the M.S. degree in computer science and technology, and the Ph.D. degree in geographic information engineering from the China University of Geosciences, Wuhan, China, in 2015, 2018, and 2021, respectively.

He is currently an Associate Professor with the School of Computer Science, China University of Geosciences. His research interests include data management, high-performance computing, and high-resolution remote sensing image processing.

Jiang Li received the B.S. and M.S. degrees in computer science and the Ph.D. degree in geographic information systems from the China University of Geosciences, Wuhan, China, in 1994, 2008, and 2016, respectively.

Meanwhile, he was a Visiting Scholar in East Carolina University, USA from 2012 to 2013. His research interests include Spatio-temporal data analysis and knowledge mining, intelligent computing and Semantic Web.



Jianmei Cheng received the B.S., M.S., and Ph.D. degrees in hydrogeology and engineering geology from the China University of Geosciences, Wuhan, China, in 1994, 1996, and 1999, respectively.

In 2002, she was a Research Assistant with the Department of Earth Sciences, University of Hong Kong, Hong Kong. Her research interests include groundwater flow, solute migration numerical simulation technology, GIS application, oilfield hydrogeology, and basin fluid-rock interaction.