The Improved Unet Semantic Segmentation Network for Remote Sensing Images

Hang Zhu, Ji Zhao

***Abstract*—With the development of artificial intelligence, deep learning has been increasingly used to achieve automatic detection of geographic information, replacing manual interpretation and improving efficiency. However, remote sensing images themselves have the issue of slight inter-class variance and significant intra-class variance, making it challenging to extract valuable information. Additionally, the increasing resolution and size of remote sensing images in recent years have introduced more complexity in the types of information, further increasing the difficulty of extracting valuable data. This paper proposes an improved Unet semantic segmentation network (referred to as RAUnet). First, in the encoder, continuous convolutional blocks are enhanced to extract features. At the same time, the EMAM multi-scale attention module is employed for cross-channel learning, capturing information from different feature channels of the target and using the surrounding feature information to assist in distinguishing target information. To capture multi-directional long-range dependencies, the Lo2 module is used for long-range modeling, which captures not only local contextual information but also long-range dependencies. In the decoder, a Dysample upsampling module is used to restore feature details, and in the skip connection layer, features are added for feature fusion. Experimental results show that compared to mainstream models, the proposed method achieves superior segmentation results on the Potsdam and Vihingen datasets.**

***Index Terms*—Attention Mechanisms, Deep Learning, Remote Sensing Images, Semantic Segmentation.**

I. Introduction

In recent times, an increasing number of satellites capable of capturing high-resolution images have been launched, allowing researchers to easily access a vast amount of high-quality remote sensing imagery. Surface information extracted from high-resolution remote sensing images plays a crucial role in land planning, construction, disaster prevention, and other fields. However, semantic segmentation of remote sensing images is often affected by the inherent characteristics of the images themselves, such as the complexity of ground objects, scale variability, occlusions, noise, and class imbalance, which can lead to poor segmentation results. Therefore, how to accurately and efficiently extract useful information from complex high-resolution remote sensing images has become a key focus of remote sensing image analysis.

Traditional semantic segmentation methods mainly include edge-based segmentation, region-based segmentation, threshold-based segmentation, and segmentation based on specific theories. These methods primarily rely on shallow semantic information such as color, texture, and gradients. While they perform well in extracting low-level semantic information, they struggle to meet the precision and efficiency requirements of modern intelligent remote sensing image analysis under more complex conditions.

Recently, the swift advancement in deep learning techniques has offered innovative support for image analysis. The advancements in convolutional neural networks (CNNs) have brought significant progress in computer vision, prompting researchers to develop the first end-to-end fully convolutional network (FCN) based on these principles. By substituting the completely connected layer in CNNs with convolutional layers, the FCN allows for handling input images of varying dimensions and has demonstrated encouraging performance. In the subsequent development, the encoder-decoder structure demonstrated strong capabilities and gradually became the mainstream network structure for semantic segmentation. In this structure, the encoder extracts feature information, and the decoder integrates and reconstructs different semantic features. For example, Unet uses skip connections in the encoder phase to integrate features, capturing more semantic information and

Fig. 1. Challenges in remote sensing images, including material color similarity and occlusion

Manuscript received September 12, 2024; revised March 23, 2025.

Hang Zhu is a postgraduate student at the School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China. (E-mail: igotit1998@163.com).

Ji Zhao is a professor of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China. (Corresponding author, e-mail: 319973500069@ustl.edu.cn).

improving segmentation accuracy, while the DeepLab series employs the atrous spatial pyramid pooling (ASPP) module for multi-scale feature extraction, and the encoder integrates these different scales of feature information for segmentation.

However, the inherent characteristics of remote sensing images, such as small inter-class variance, large intra-class variance, small object scales, scattered information, and occlusions, pose significant challenges to remote sensing semantic segmentation, as shown in Figure 1. CNN-based models, as the convolutional process continues, tend to lose small object information and blur object boundaries. Different ground semantic classes may share similar sizes, shapes, and colors, making segmentation difficult. Thus, rich contextual information and spatial features are needed for inference. CNNs succeed due to their inductive biases, but they also have limitations, particularly in capturing global context due to their localized nature. This is especially problematic in remote sensing semantic segmentation, where the useful information in remote sensing images is often scattered. Relying solely on CNN's local windows to extract information is clearly insufficient. We can often derive valuable information from the background of a target, as the environment surrounding the target is typically correlated with it.

With the success of Transformer on natural language, new ideas have been opened up for global relational modeling in the field of computer vision. ViT has surpassed mainstream CNN models in one fell swoop by virtue of its ability to extract global contextual information and its ability to model remote dependencies. Hlts, ignoring the image Transformer is originally designed for 1D results, ignoring the structure of 2D images, and does not have the inductive bias of CNNs, which leads to the fact that Transformer needs more training data to gowever, the current ViT still has deficiencies. The Transformer is designed based on the self-attention mechanism, and its computational complexity is quadratic with the size of the image, especially in the intensive prediction task of high-resolution remote sensing images; its computational complexity with a large-size image input is unacceptable. Transformer is initially designed for the input of the 1D resuet a better result; Transformer only focuses on spatial properties and ignores channel features, and often different channels contain important semantic information as well.

In order to address the above problems, this paper proposes a new network model structure, which is an Unet-like encoder-decoder structure, CNN used in the first three layers of the network's encoder, feature excitation module and Lo2 module used in the fourth layer of the encoder-decoder and bottleneck layer, and DySample module used in the decoder to perform the up-sampling, and the jump-junction layer to perform the feature fusion. Specifically, in remote sensing images, the feature information always appears in a specific environment; we use the multi-scale attention module in the third layer of the encoder so that the network can also capture the practical information of different feature channels; the information always appears sporadically in remote sensing images, and the range of information read by the local convolution is limited, so in the stage of encoding area, we use the global local module (Lo2) to extract the global contextual information; Facing the problem that small target information is easily lost during the convolution process, the encoder features of the same scale of the jump connection layer fusion module are used for information enhancement, and the fusion of low-level and high-level semantic information is used to assist in the segmentation of small targets.

II. Related work

A. CNN-Based Semantic Segmentation for Remote Sensing Images

With the promotion of various remote sensing image recognition competitions, CNN-based remote sensing image semantic segmentation has received widespread attention. Convolutional networks (CNNs) can effectively extract image features and perform precise pixel-level segmentation. Various model architectures, including fully convolutional networks, encoder-decoder models, and networks utilizing dilated convolutions, have gained significant popularity in semantic segmentation. In their analysis, Zhao et al. [1] investigated the use of convolutional neural networks (CNNs) for semantic segmentation. Subsequently, they integrated Conditional Random Fields (CRFs) to capture the relationships among the identified semantic regions. D Marcos et al. [2] A novel CNN architecture named the Rotation Equivariant Vector Field Network (RotEqNet) was introduced to embed rotational invariance directly within the network structure. Drawing inspiration from the UNet architecture, researchers Dong et al. [3] introduced DenseU-Net. This model emphasizes integrating features at a smaller scale through a tightly-knit fusion approach. K Nogueira et al. [4] employed a paradigm incorporating multiple contexts and training networks with various patch dimensions to allow the system to extract diverse contextual features from differing environments. As the network evaluates these varied patch sizes, it assigns scores to each, aiding in identifying the most suitable size for the problem at hand. Zhang et al. [5] To enhance the embedding of contextual information, multi-scale feature maps were produced by parallel convolutional structures organized in distinct branches of HRNet. At the same time, an adaptive spatial pooling module was crafted to consolidate more localized context information. Chen et al. [6] introduced a convolutional network with an adaptive receptive field, effectively balancing extracting features related to large and small-scale objects. The study known as FactSeg [7] introduced a dual-branch decoder with a symmetrical structure, consisting of a branch for foreground activation and another for enhancing semantics, which together utilize multi-scale feature integration via skip connections to boost the precision of segmenting small objects in remote sensing images.

B. Attention Mechanism

In recent years, self-attention mechanisms have been widely applied in computer vision tasks. Zhao et al. [8] proposed region-level attention to encode visual-text features in video captioning. SENet [9] uses a global average pooling layer to represent the relationships between channels, automatically learning the importance of different channels. ECA-Net [10] improves SENet by avoiding dimensionality reduction and introducing appropriate cross-channel interaction to enhance segmentation accuracy. CBAM [11] combines channel-level attention and spatial-level attention for adaptive feature refinement. LANet [12] developed

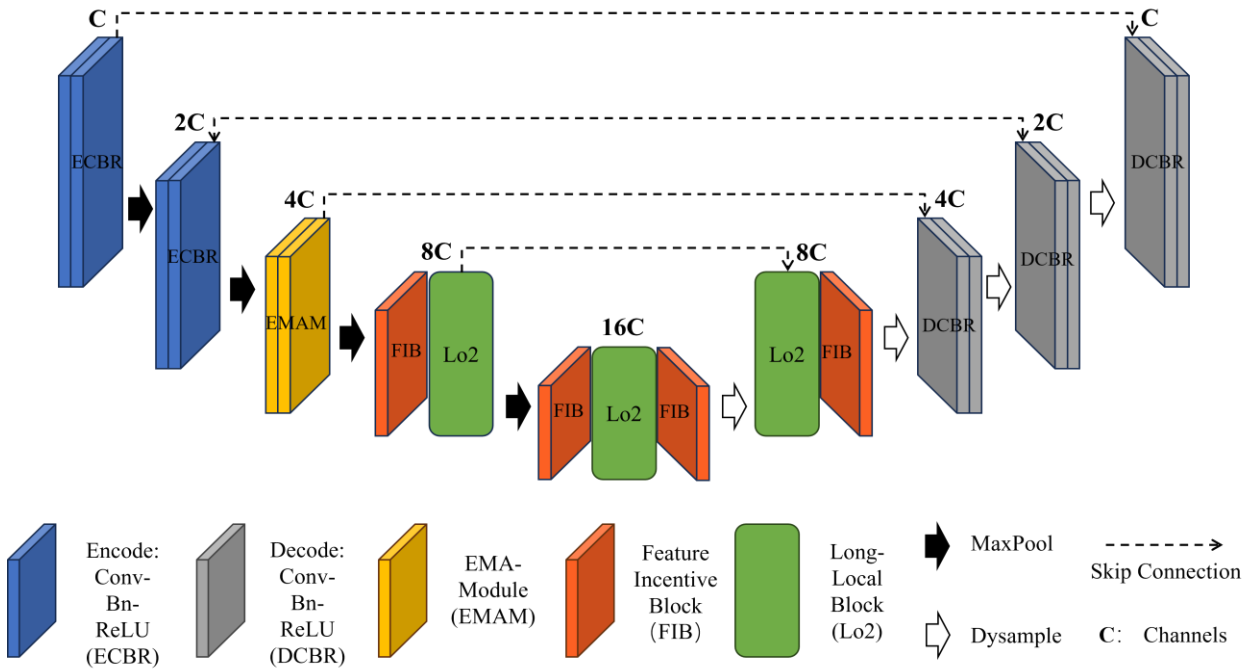

Fig. 2. Network architecture diagram

an attention-based module focusing on key feature map areas. The approach in MANet [13] focuses on capturing contextual relationships via various efficient attention mechanisms. It introduces a unique kernel attention method characterized by its linear complexity, which reduces the computational burden typically associated with attention mechanisms. Hou et al. [14] considered the information present across different channels and factors related to orientation and spatial position, thereby enhancing the model's capability to identify and locate objects precisely. Su et al. [15] examined groups of similar items within limited sets of images, employing a focused attention mechanism to discern distinctive features of comparable items from other images in the limited set.

C. Vision Transformer

In recent years, Transformer models have demonstrated powerful performance in the field of computer vision. Notably, the introduction of Vision Transformer (ViT) [16], which successfully applied the Transformer architecture to image classification tasks, has since sparked widespread research in image segmentation. For remote sensing image semantic segmentation, researchers have proposed various Transformer-based models. For example, SegFormer [17] uses a Transformer encoder to capture global contextual information of the image and generates high-quality segmentation results through a lightweight decoder. The design of SegFormer balances efficiency and accuracy, with the encoder effectively extracting multi-scale features and the decoder generating detailed segmentation maps through simple layer-by-layer upsampling and fusion operations. Additionally, the Swin Transformer [18] improves the model's efficiency and segmentation accuracy by dividing the input image into several non-overlapping windows and applying the self-attention mechanism within each window to extract features. Then, through shifting and merging windows, it captures global features. This method reduces computational complexity while retaining the advantage of Transformers in capturing long-range dependencies. Furthermore, TransUNet [19] combines the strengths of UNet and Transformer, using CNN to extract local features and enhancing the ability to model global features through the Transformer module. The encoder part of TransUNet employs standard convolution operations to extract multi-scale features. At the same time, the decoder incorporates Transformer modules to capture long-range dependencies between features, thereby improving segmentation performance. He et al. [20] embedded the Swin Transformer into a dual-branch structure in CNN's Unet to capture both global and local contexts. These Transformer-based models have performed exceptionally well in remote sensing image semantic segmentation tasks, significantly enhancing both segmentation accuracy and efficiency.

III. Theory and Methodology

A. Network Model Overview

The structure of our network is shown in Figure 2, which continues to follow the classic encoder-decoder and skip connection design of UNet. The encoder and decoder each consist of four downsampling and DySample modules for upsampling. The first two layers of the decoder retain the same structure as the original UNet, where each layer contains two standard convolution blocks. In the third layer of the encoder, a multi-scale attention module (EMAM) is employed to capture information across spatial dimensions and different channels. In the fourth layer of both the encoder and decoder, the feature excitation module and Lo2 module are applied. The feature excitation module handles feature channel compression and expansion, while the Lo2 module captures dependencies between global and local contextual information. The decoder employs a highly efficient dynamic sampling mechanism (DySample) for upsampling. Skip connections use additive fusion to combine features from the encoder at the same scale. Finally, the primary segmentation head at the last layer of the encoder generates the main

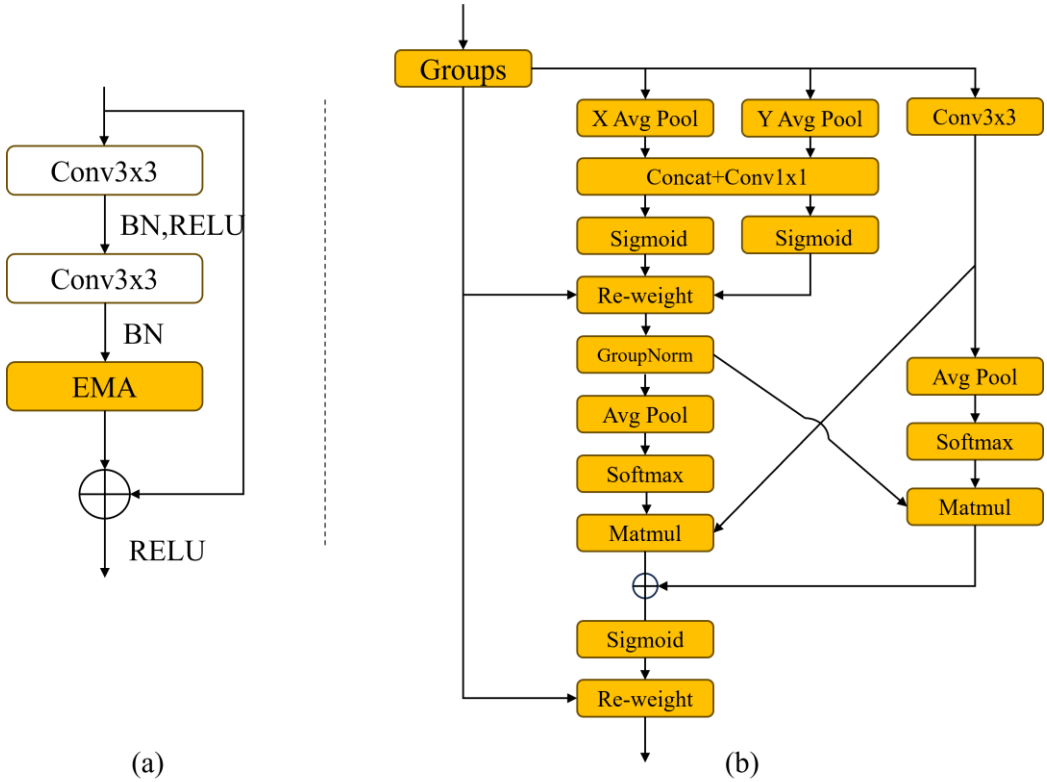

Fig. 3. Structural design of EAMA and EMA modules; (a) EAMA module; (b) EMA module as a sub-component of EAMA

output, which is used to compute the primary loss. An auxiliary segmentation head is applied in the penultimate layer of the encoder to assist the network in improving segmentation accuracy. The following sections provide a detailed explanation of each module.

B. EMAM Module

In computer vision tasks, channel and spatial attention mechanisms are effective in generating more distinguishable feature information. To balance the network's segmentation accuracy and parameter count, we use two consecutive multi-scale attention modules (EMAM) only in the third layer of the network encoder. The EMAM module enhances a residual network block. Figure 3 (a) shows the overall structure of EMAM. The input feature X first passes through a 3×3 convolution, batch normalization (BN), and ReLU activation layer. It is then processed by another 3×3 convolution, BN layer, and EMA [21] attention layer, as shown in Figure 3 (b). Finally, a residual connection is applied with the original features, followed by a ReLU activation output, as expressed in Formula 1.

$$EMAM(X) = RELU(EMA(\mathrm{Conv}_{BN}(Conv_{BN,RELU}(X))) + X) \quad (1)$$

For any input feature $X \in R^{C\times H\times W}$, the EMA (Efficient Multi-scale Attention) mechanism partitions X along the channel dimension into G groups of sub-features, as expressed by Formula $X = [X_0, X_1, ..., X_{G-1}]$, $X_i \in R^{C//G\times H\times W}$.

The parallel network structure effectively avoids extensive sequential structures and deep networks. Aftergrouping, two parallel 1×1 branches and one parallel 3×3 branch are used to capture dependencies across all channels. The two 1×1 branches employ $1D$ global average pooling in both horizontal and vertical directions to facilitate cross-channel information interaction. Their outputs are concatenated along the height dimension and undergo a 1×1 convolution to generate two vectors. After linear convolution, two Sigmoid functions approximate a $2D$ binary distribution. Matrix multiplication is applied to fuse features across different channels to promote cross-channel interaction between the parallel paths further. The 3×3 branch captures local cross-channel information through 3×3 convolution, preserving accurate spatial structure in the channels. To aggregate cross-spatial information from different spatial dimensions, the outputs of the 1×1 and 3×3 branches undergo $2D$ global average pooling and Softmax normalization. Subsequently, the features from the 1×1 branch and the 3×3 branch before pooling are multiplied and summed together. A Sigmoid function is then used for fitting. Finally, the output is multiplied with the original input X divided into G groups of sub-features, resulting in an output with the same dimensions as X. This mechanism enhances the network's ability to capture diverse channel and spatial features.

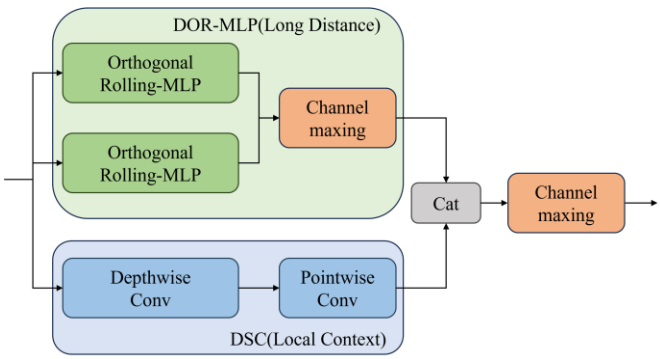

Fig.4. Lo2 module

C. Lo2 Module

In high-resolution remote sensing urban images, complexly shaped artificial objects frequently appear, making it challenging to achieve precise semantic segmentation without global context information. The Lo2 [22] module, as shown in Figure 4, consists of a global branch (DOR-MLP) and a local branch (DSC). This module uses a combination of CNN and MLP to capture both global and local contextual semantic information from feature images. In the global branch, we employ two parallel OR-MLP modules to capture the global context. To enhance performance, we use Depthwise Separable Convolution (DSC) to capture local information. As illustrated in Figure 4, the DOR-MLP and DSC modules are run in parallel. Their outputs are concatenated along the channel dimension, and then a channel mixing calculation is applied to reduce the channels to C. This process constitutes the Lo2 module, as expressed in Formula 2.

$$\mathrm{Lo2}(X) = CM(Concat[MLP_{DOR}(X), DSC(X)]) \quad (2)$$

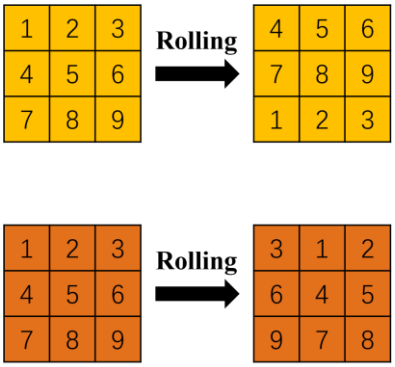

Fig. 5. Rolling operation

R-MLP Module

Given a feature matrix $X \in H \times W \times C$ with spatial resolution $H \times W$, feature channels C, height index $h_i (i \in [1, H])$, width index $w_j (j \in [1, W])$, and depth index $c_k (k \in [1, C])$, we perform a Rolling operation on each channel layer of the feature matrix. The Rolling operation consists of two steps: displacement and cropping. First, for the feature map at channel index C_k, we apply a rolling operation with a step size of k. Then, we crop the feature map to remove excess parts and fill in the missing areas. Assuming the feature matrix is denoted as $C, H, W = (3, 3, 3)$, as shown in Figure 5, a Rolling operation with a step size of 2 is performed along the height direction for channel $C = 2$, and a Rolling operation with a step size of 3 is performed along the width direction for channel $C = 3$. Finally, the feature matrix is adjusted to $X \in (H * W, C)$, and a weight-shared channel projection is executed at spatial positions (h_i, w_j) to capture contextual semantic information.

DOR-MLP Module

An R-MLP module can encode long-range dependencies either along the height or width direction. By first applying R-MLP along the height direction and then applying another R-MLP along the width direction, we form a diagonal receptive field that captures feature information from different positions. For the input X (as expressed in Formula 3), we first apply R-MLP in one direction, followedby GELU activation. Next, we apply another R-MLP in the perpendicular direction and perform a residual connection with the input X. This process creates a Diagonal-Oriented R-MLP (OR-MLP) module, as illustrated in Figure 6.

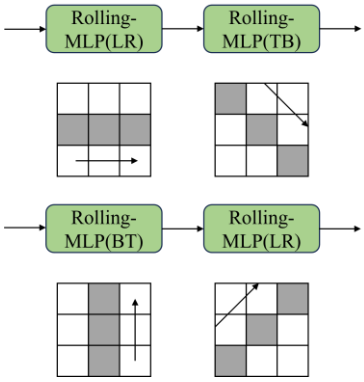

Fig.6. Controlling of different R-MLP to capture remote dependencies in different directions

$$MLP_{OR}(X) = (RMLP_2(GELU(RMLP_1(X)))) + X \quad (3)$$

To capture long-range dependencies in different directions, we use two parallel and complementary OR-MLP modules. The first module applies R-MLP from left to right and then from top to bottom. The second module applies R-MLP from bottom to top and then from left to right. This parallel design enables long-range information exchange in four directions: width, height, and both positive and negative diagonals. The input X (as shown in Formula 4) is fed into two parallel OR-MLP modules operating in different directions. The outputs of these two modules are then concatenated along the channel dimension. Standard normalization (LayerNorm) and channel mixing are applied to reduce the number of feature channels back to C. Finally, a residual connection is made with the input X, forming the Dual-Oriented Rolling-MLP (DOR-MLP) module, as illustrated in Figure 7.

$$MLP_{DOR}(X) = CM(LN(Concat[RMLP_1(X), RMLP_2(X)])) + X \quad (4)$$

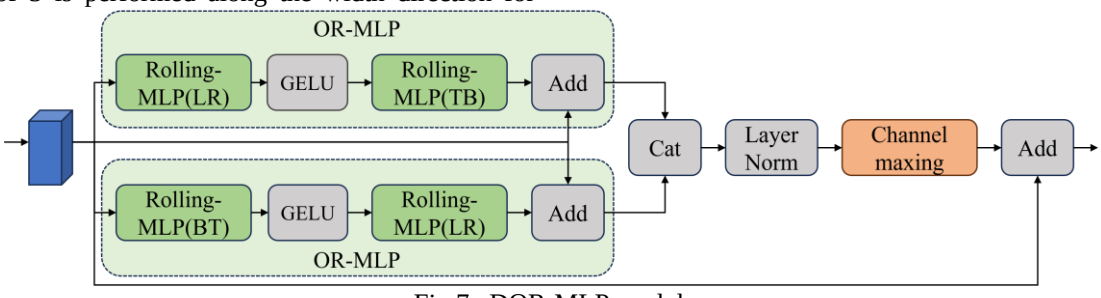

Fig.7. DOR-MLP module

D.DySample Upsampling Module

Feature upsampling is a critical factor in gradually restoring feature resolution in dense prediction models for remote sensing images. In semantic segmentation networks, the most commonly used upsamplers are nearest-neighbor and bilinear interpolation. However, these methods follow fixed rules for interpolation and may not be well-suited for complex tasks in remote sensing image segmentation. The DySample [23] module adopts a dynamic upsampling method using point sampling. As illustrated in Figure 8, a simple dynamic upsampling process is shown. It involves a feature map X with dimensions $C \times H_1 \times W_1$ along with a sampling set S that measures $2 \times H_2 \times W_2$. This grid_sample function leverages the coordinates from S to interpolate the feature map X into $X^{'}$, resulting in dimensions of $C \times H_2 \times W_2$. As shown in Equation 5.

$$X' = grid\_sample(X, S) \tag{5}$$

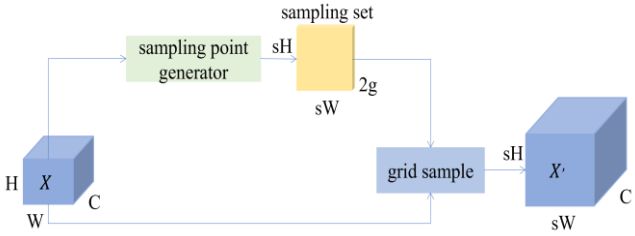

Fig.8. Sampling based dynamic upsampling

Considering a feature map X and a specified upsampling factor s , a fully connected layer featuring input and output dimensions of C and $2s^2$, respectively, produces an offset O of dimensions $2s^2 \times H \times W$. It undergoes a pixel transformation to reshape into dimensions of $2 \times sH \times sW$. The new sampling set S is created by adding the original sampling grid G to the offset O . If all sampling positions are fixed at the same initial position, the spatial relationships will be ignored. We change the initial positions to "bilinear initialization," separating the initial positions so they are evenly distributed. In this case, zero offset results in bilinear interpolation of the feature map. Due to the presence of the normalization layer, the sampling positions might overlap in their movement range, which affects the predictions near boundaries. To alleviate this, we multiply the offset by 0.25 to constrain the movement range of the sampling positions. Additionally, we divide the feature map into g groups along the channel dimension, generating g groups of offsets such that the features in each group share the same sampling set. As shown in Figure 9, this process is defined by the following equation 6 and equation 7.

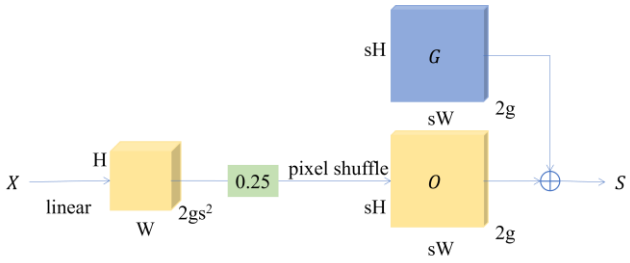

Fig.9. Sampling point generator in DySample

The process is defined by the following formula:

$$O = 0.25 Linear(X) \tag{6}$$

$$S = G + O \tag{7}$$

The reshaping operation is omitted, and the upsampled feature map $X^{'}$ with a size of $C \times sH \times sW$ can be generated using grid_sample. as shown in Equation 5.

IV. Experiment Process and Results Analysis

A. Experimental Data

The Vaihingen dataset, provided by ISPRS, includes 33 high-resolution ground images with a sampling distance of 9 centimeters. These images cover an area of 1.38 square kilometers in Vaihingen, Germany, including a small village, multiple individual buildings, and multi-story buildings. The dataset is annotated with 6 categories for semantic segmentation research: impervious surfaces, buildings, low vegetation, trees, cars, and background. We divided the dataset into 16 images for training, with image IDs 1, 3, 5, 7, 11, 13, 15, 17, 21, 23, 26, 28, 30, 32, 34, and 37, and the remaining 17 images for testing. Only RGB images are used for training and testing, and these original images are cropped to 512x512 pixels.

The Potsdam dataset, also provided by ISPRS, includes 38 high-resolution ground images of 6000×6000 pixels, with a sampling distance of 5 centimeters. These images cover an area of 3.42 square kilometers in the city of Potsdam, including numerous building clusters, narrow streets, and dense cluster structures. The dataset is annotated with 6 categories for semantic segmentation research: impervious surfaces, buildings, low vegetation, trees, cars, and background. We divided the dataset into 14 images for testing, with image IDs 2_13, 2_14, 3_13, 3_14, 4_13, 4_14, 4_15, 5_13, 5_14, 5_15, 6_13, 6_14, 6_15, and 7_13, and the remaining 24 images for training. Only RGB images are used for training and testing, and these original images are cropped to 512x512 pixels.

TABLE I
Ablation Experiments on the Potsdam Dataset

Model	IoU					Evaluation index	
	Impervious Surface	Building	Low	Tree	Car	mIoU	aAcc
Baseline	77.87	84.12	67.61	70.23	76.4	75.24	78.56
Baseline+Lo2	80.3	86.76	70.63	73.11	79.41	78.04	80.39
Baseline+Lo2+EMAM	81.01	87.45	71.43	73.39	79.33	78.52	80.8
Baseline+Lo2+EMAM+Dysample	81.22	87.75	71.42	73.35	80.03	78.75	81.83

B. Experimental Settings

The experiments in this study are conducted on an NVIDIA GeForce GTX 3090 GPU (24GB memory) using a Python deep learning framework. The training input images are 3-channel 512×512 pixels, with a batch size set to 8 and the training run for 100 epochs. The initial learning rate is set to 0.01, and the SGD optimizer is used with a momentum of 0.9 and a weight decay of 0.0005. Additionally, image scaling and flipping augmentation techniques are applied during the training process.

C. Loss Function

In different datasets, class imbalance leads to model training being concentrated on the classes with larger proportions, while "ignoring" the classes with smaller proportions. During the training phase, we use not only the main feature refinement head but also construct an additional auxiliary head to assist in optimizing the network, as shown in Figure 2. Previous research has demonstrated the effectiveness of such a multi-head segmentation architecture. Based on this multi-head design, we train the entire network using the main loss L_p and the auxiliary loss L_{aux}.

Both the main loss and auxiliary loss utilize a combined loss function that includes cross-entropy loss and Dice loss. The cross-entropy loss and Dice loss are defined by Formula 8 and Formula 9, respectively.

$$L_{CE} = -\frac{1}{N}\sum_{N=1}^{N}\sum_{K=1}^{K} y_k^{(n)} \log \hat{y}_k^{(n)} \tag{8}$$

$$L_{dice} = 1 - \frac{2}{N}\sum_{n=1}^{N}\sum_{k=1}^{K} \frac{\hat{y}_k^{(n)} y_k^{(n)}}{\hat{y}_k^{(n)} + y_k^{(n)}} \tag{9}$$

Where N and K denote the number of samples and the number of classes, respectively, $y^{(n)}$ and $\hat{y}^{(n)}$ represent the one-hot encoded ground truth labels and the corresponding softmax outputs from the network, respectively, and $n \in [1,...,N]$ and $\hat{y}_k^{(n)}$ represent the confidence scores of sample n belonging to class k. The combined loss for the main loss is defined by Formula 10.

$$L_p = L_{ce} + L_{dice} \tag{10}$$

Thus, the overall total loss can be represented by Formula 11, where α is set to 0.4 in this experiment.

$$L = L_p + \alpha \times L_{aux} \tag{11}$$

D. Evaluation Metrics

To evaluate our model's performance on the Potsdam and Vaihingen datasets, we adopt Intersection over Union (*IoU*), Mean Intersection over Union (*MIoU*), and Overall Accuracy (*aAcc*) as evaluation metrics. The *IoU* calculation method is the ratio of the intersection (the overlapping part between the prediction and the ground truth) to the union (the combined part of the prediction and ground truth), as shown in Equation (12). *MIoU* is the average of the *IoU* values for all classes. *aAcc* represents the overall average accuracy, as shown in Equation (13). In these equations, TP, FP, and FN represent True Positives, False Positives, and False Negatives, respectively, while C represents the total number of classes.

$$IoU = \frac{TP}{TP + FP + FN} \tag{12}$$

$$\text{aAcc} = \frac{\sum_{i=1}^{C} TP_i}{\sum_{i=1}^{C} (TP_i + FN_i)} \tag{13}$$

E. Ablation Study

To validate the effectiveness of each module, we conducted a series of evaluations using UNet as the baseline model on the Potsdam dataset, as shown in Table I. The addition of the Lo2 module to the baseline network yields the highest improvement in performance. The EMAM and DySample modules are built upon the Lo2 module, with the EMAM module providing better overall segmentation performance compared to the DySample module. However, both modules positively impact the network. The experiments demonstrate that each module effectively enhances the segmentation accuracy of the network.

F. comparative experiment

To further demonstrate the superiority of our model, we compared different network architectures on the Potsdam and Vaihingen datasets and visualized the experimental results.

The comparison experiments conducted on the Potsdam dataset are shown in Table II. Table II lists the *mIoU* , Ave $F1$, and *IoU* values for each category of each model. Compared to other advanced semantic segmentation networks, the proposed RAUnet outperforms other classic segmentation models overall. It achieves an *mIoU* improvement of 4.08%, 0.95%, 0.73%, 3.8%, and 3.83% over Pspnet, UNet++, Danet, Deeplabv3+, and UNetformer models, respectively. Additionally, it leads in Ave $F1$ by 3.9%, 4.1%, 1.13%, 4.12%, and 3.95%, respectively. Figure 10 shows the segmentation results of different models on the Potsdam dataset.

TABLE II
COMPARISON OF EXPERIMENTS ON THE POTSDAM DATASET

Model	IoU					Evaluation index	
	Impervious Surface	Building	Low	Tree	Car	MIoU	aAcc
Pspnet	78.49	83.84	68.16	68.11	74.74	74.67	77.39
Unet++	79.77	85.85	69.68	73.32	80.39	77.8	77.73
Danet	81.08	86.91	72.23	73.67	76.24	78.02	80.7
Deeplabv3+	79.15	86.5	68.26	67.93	72.92	74.95	77.71
Unetformer	78.72	86.06	68.15	67.2	74.51	74.92	77.88
Ours	81.22	87.75	71.42	73.35	80.03	78.75	81.83

TABLEIII
COMPARISON EXPERIMENTS ON THE VAIHINGEN DATASET

Model	IoU					Evaluation index	
	Impervious Surface	Building	Low	Tree	Car	MIoU	aAcc
Pspnet	72.79	74.82	62.05	73.72	28.33	62.34	64.79
Unet++	79.66	84.54	67.83	79.14	53.78	72.99	69.18
Danet	74.8	77.14	60.75	72.49	31.98	63.43	63.61
Deeplabv3+	74.49	76.63	58.56	72.1	29.86	62.32	63.54
Unetformer	74.7	78.91	62.66	74.29	28.39	63.79	62.48
Ours	80.95	84.53	68.4	78.3	47.9	72.01	70.43

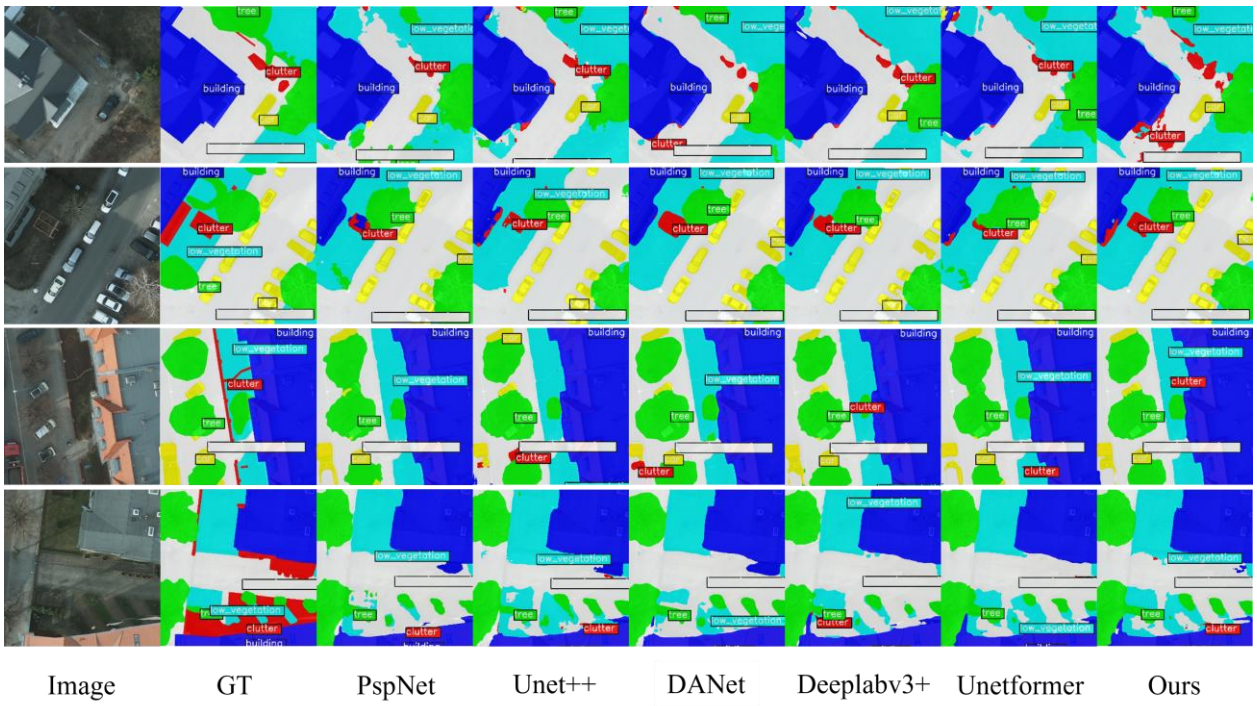

Fig.10. Segmentation results on the Potsdam dataset

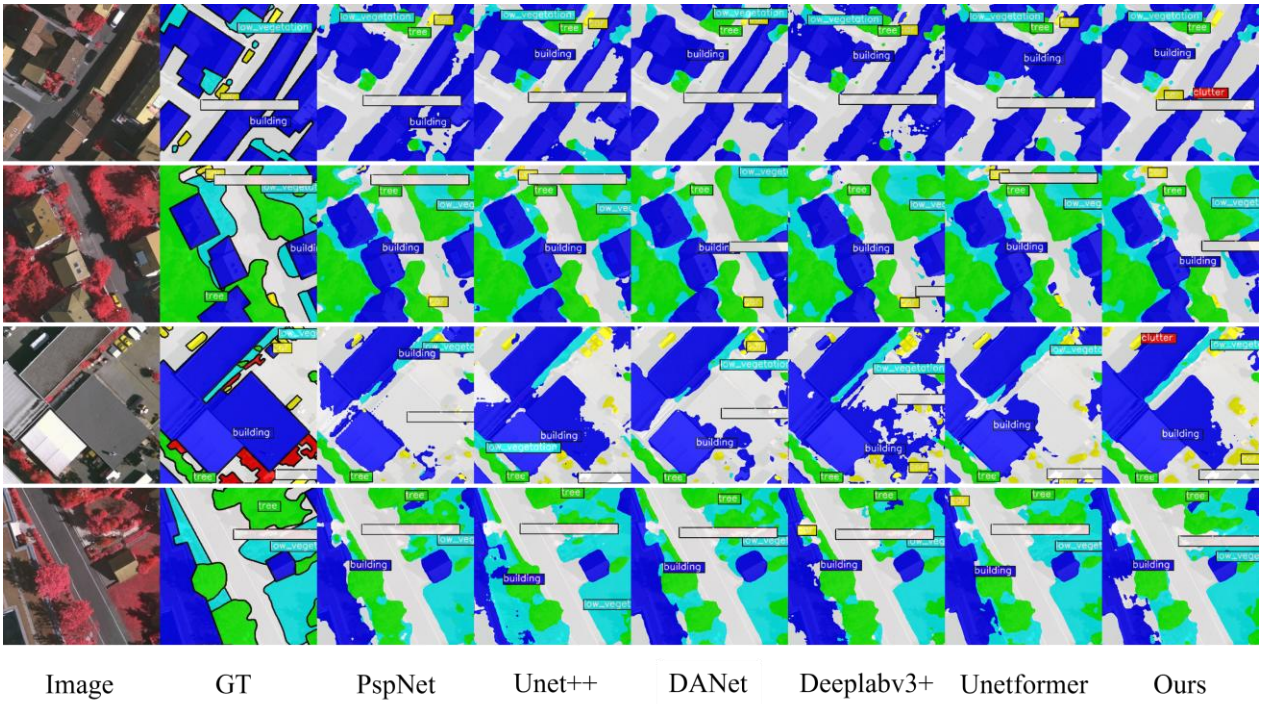

Fig.11. Segmentation results on the Vaihingen dataset

The comparison experiments conducted on the Vaihingen dataset are shown in Table III. Table III lists the *mIoU* , Ave $F1$, and *IoU* values for each category of each model. Compared to other advanced semantic segmentation networks, the proposed RAUnet outperforms other classic segmentation models overall. It achieves an *mIoU* improvement of 9.67%, -0.98%, 8.58%, 9.69%, and 8.22% over Pspnet, UNet++, Danet, Deeplabv3+, and UNetformer models, respectively. Additionally, it leads in Ave $F1$ by 5.64%, 1.25%, 6.82%, 6.98%, and 7.95%, respectively. Figure 11 shows the segmentation results of different models on the Vaihingen dataset. Overall, the proposed model demonstrates superior segmentation performance on both the Potsdam and Vaihingen datasets and shows strong competitiveness.

V. Conclusion

With the advancement of artificial intelligence, intelligent image analysis can significantly alleviate the workload for practitioners. This is especially true in the field of remote sensing image analysis, which has garnered substantial attention. Intelligent analysis of remote sensing images can be applied to numerous real-world scenarios. Significant progress has been made in the semantic segmentation of remote sensing images using deep learning, with improvements in segmentation accuracy and efficiency achieved through the optimization of model algorithms, architectures, and prior information. However, high-resolution remote sensing images present challenges such as sizeable intra-class variance, slight inter-class variance, and the complex and varied structures of objects. This paper proposes a feasible solution to address these issues by improving the classic UNet network as the base model. To ensure the network captures information from different feature channels, we use the EMAM module to focus on channel-specific information. The Lo2 module is employed to model long-range dependencies in multiple directions while maintaining parameter efficiency, capturing both local context and remote dependencies. The Dysample upsampling module is used in the encoder to restore features and preserve spatial details. Experiments conducted on the Potsdam and Vaihingen datasets demonstrate that the proposed modules are practical and competitive compared to current mainstream models.

Considering practical applications, especially deployment on mobile platforms, future work will focus on reducing model parameters and improving segmentation efficiency while maintaining segmentation accuracy.

References

[1] Zhao W, Du S, Wang Q, et al. Contextually guided very-high-resolution imagery classification with semantic segments[J]. ISPRS Journal of Photogrammetry and Remote Sensing, 2017, 132(oct.):48-60.DOI:10.1016/j.isprsjprs.2017.08.011.

[2] Marcos D, Volpi M, Kellenberger B, et al. Land cover mapping at very high resolution with rotation equivariant CNNs: towards small yet accurate models[J]. Isprs Journal of Photogrammetry & Remote Sensing,2018,145PA(NOV.):96-107.DOI:10.1016/j.isprsjprs.2018.01.021.

[3] Dong R, Pan X, and Li F. DenseU-Net-based Semantic Segmentation of Small Objects in Urban Remote Sensing Images[J]. IEEE Access, 2019, PP(99):1-1.DOI:10.1109/ACCESS.2019.2917952.

[4] Nogueira K, Mura M D, Chanussot J, et al. Dynamic Multicontext Segmentation of Remote Sensing Images Based on Convolutional Networks[J]. IEEE Transactions on Geoscience & Remote Sensing, 2019, PP(99):1-18.DOI:10.1109/TGRS.2019.2913861.

[5] Zhang J, Lin S, Ding L, et al. Multi-Scale Context Aggregation for Semantic Segmentation of Remote Sensing Images[J]. Remote Sensing, 2020, 12(4):701.DOI:10.3390/rs12040701.

[6] Chen X, Li Z, Jiang J, et al. Adaptive Effective Receptive Field Convolution for Semantic Segmentation of VHR Remote Sensing Images[J]. IEEE Transactions on Geoscience and Remote Sensing, PP(99):1-15[2024-07-09].DOI:10.1109/TGRS.2020.3009143.

[7] Ma A, Wang J, Zhong Y, et al. FactSeg: Foreground Activation Driven Small Object Semantic Segmentation in Large-Scale Remote Sensing Imagery[J]. IEEE Transactions on Geoscience and Remote Sensing, 2021.DOI:10.1109/TGRS.2021.3097148.

[8] Zhao, Bin, X. Li, and X. Lu . "CAM-RNN: Co-Attention Model Based RNN for Video Captioning." IEEE Transactions on Image Processing 28.99(2019):5552-5565.

[9] Jie,Shen, Samuel, et al. Squeeze-and-Excitation Networks.[J].IEEE transactions on pattern analysis and machine intelligence, 2019.DOI:10.1109/TPAMI.2019.2913372.

[10] Wang Q, Wu B, Zhu P, et al. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2020.DOI:10.1109/CVPR42600.2020.01155.

[11] Woo S, Park J, Lee J Y, et al. CBAM: Convolutional Block Attention Module[J].Springer, Cham, 2018.DOI:10.1007/978-3-030-01234-2_1.

[12] Ding L, Tang H, Bruzzone L LANet: Local Attention Embedding to Improve the Semantic Segmentation of Remote Sensing Images[J]. IEEE Transactions on Geoscience and Remote Sensing, 2021, 59(1):426-435. DOI:10.1109/TGRS.2020.2994150.

[13] Li R, Zheng S, Duan C, et al. Multi-Attention-Network for Semantic Segmentation of Fine Resolution Remote Sensing Images[J]. 2020.DOI:10.13140/RG.2.2.28977.81761.

[14] Hou Q, Zhou D, and Feng J. Coordinate Attention for Efficient Mobile Network Design[J]. 2021.DOI:10.48550/arXiv.2103.02907.

[15] Su Y, Wu Y, Wang M, et al. Semantic segmentation of high resolution remote sensing image based on batch-attention mechanism[C]//IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium. IEEE, 2019: 3856-3859.

[16] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale[C]//International Conference on Learning Representations.2021.

[17] Xie E, Wang W, Yu Z, et al. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers[J]. 2021.DOI:10.48550/arXiv.2105.15203.

[18] Liu Z, Lin Y, Cao Y, et al. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows[J]. 2021.DOI:10.48550/arXiv.2103.14030.

[19] Chen J, Lu Y, Yu Q, et al. TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation[J]. 2021.DOI:10.48550/arXiv.2102.04306.

[20] He X, Zhou Y, Zhao J, et al. Swin Transformer Embedding UNet for Remote Sensing Image Semantic Segmentation[J]. IEEE Transactions on Geoscience and Remote Sensing, 2022, 60.DOI:10.1109/TGRS.2022.3144165.

[21] Ouyang D, He S, Zhang G, et al. Efficient multi-scale attention module with cross-spatial learning[C]//ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2023: 1-5.

[22] Liu Y, Zhu H, Liu M, et al. Rolling-unet: Revitalizing mlp's ability to efficiently extract long-distance dependencies for medical image segmentation[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2024, 38(4): 3819-3827.

[23] Liu W, Lu H, Fu H, et al. Learning to upsample by learning to sample[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 6027-6037.