Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.043524

ARTICLE

# A Lightweight Road Scene Semantic Segmentation Algorithm

**Jiansheng Peng**[1,2,*], **Qing Yang**[1] **and Yaru Hou**[1]

[1]College of Automation, Guangxi University of Science and Technology, Liuzhou, 545000, China

[2]Department of Artificial Intelligence and Manufacturing, Hechi University, Hechi, 547000, China

*Corresponding Author: Jiansheng Peng. Email: sheng120410@163.com

Received: 04 July 2023 Accepted: 20 September 2023 Published: 29 November 2023

**ABSTRACT**

In recent years, with the continuous deepening of smart city construction, there have been significant changes and improvements in the field of intelligent transportation. The semantic segmentation of road scenes has important practical significance in the fields of automatic driving, transportation planning, and intelligent transportation systems. However, the current mainstream lightweight semantic segmentation models in road scene segmentation face problems such as poor segmentation performance of small targets and insufficient refinement of segmentation edges. Therefore, this article proposes a lightweight semantic segmentation model based on the LiteSeg model improvement to address these issues. The model uses the lightweight backbone network MobileNet instead of the LiteSeg backbone network to reduce the network parameters and computation, and combines the Coordinate Attention (CA) mechanism to help the network capture long-distance dependencies. At the same time, by combining the dependencies of spatial information and channel information, the Spatial and Channel Network (SCNet) attention mechanism is proposed to improve the feature extraction ability of the model. Finally, a multi-scale transposed attention encoding (MTAE) module was proposed to obtain features of different resolutions and perform feature fusion. In this paper, the proposed model is verified on the Cityscapes dataset. The experimental results show that the addition of SCNet and MTAE modules increases the mean Intersection over Union (mIoU) of the original LiteSeg model by 4.69%. On this basis, the backbone network is replaced with MobileNet, and the CA model is added at the same time. At the cost of increasing the minimum model parameters and computing costs, the mIoU of the original LiteSeg model is increased by 2.46%. This article also compares the proposed model with some current lightweight semantic segmentation models, and experiments show that the comprehensive performance of the proposed model is the best, especially in achieving excellent results in small object segmentation. Finally, this article will conduct generalization testing on the KITTI dataset for the proposed model, and the experimental results show that the proposed algorithm has a certain degree of generalization.

**KEYWORDS**

Semantic segmentation; lightweight; road scenes; multi-scale transposition attention encoding (MTAE)

## 1 Introduction

In today's society, road scene segmentation has become a technology of great importance as urbanization and traffic demand continue to grow. Road scene segmentation aims to accurately separate and identify individual objects on the road from their surroundings in digital images or videos. This technology has a wide range of promising applications in areas such as autonomous driving, traffic monitoring, and intelligent transportation systems. The goal of road scene segmentation is to achieve a comprehensive understanding and perception of the traffic environment by accurately segmenting the vehicles, pedestrians, traffic signs, and other elements on the road. By effectively separating all objects on the road, road scene segmentation provides autonomous vehicles with the necessary environment awareness to ensure safe driving. At the same time, road scene segmentation can be used in traffic monitoring systems to monitor traffic flow in real-time, detect violations, and optimize traffic signal control, thereby improving road safety and traffic efficiency.

Semantic segmentation is one of the important tasks in the field of computer vision, aiming to classify each pixel in an image or video into different semantic classes accurately. With the continuous development of deep learning, semantic segmentation models have also evolved. From the original full convolutional network (FCN) [1] to various improved methods later, each generation of networks has introduced new ideas and techniques in feature extraction, contextual information, and multi-scale features to continuously improve the accuracy and efficiency of semantic segmentation. Common semantic segmentation models are based on convolutional neural networks to achieve pixel-level classification. The main feature of these models is the adoption of an encoder-decoder structure and the combination of improved strategies such as skip connections and contextual coding, which improve the accuracy of image semantic segmentation. In the ILSVRC2012 vision competition [2], the AlexNet network [3] achieved state-of-the-art results (SOTA) by improved the accuracy of the model through techniques such as nonlinear activation functions, Dropout layers, data augmentation, and multi-GPU training. These ideas and techniques have since been widely used in deep learning models. In 2014, Long et al. improved the convolutional neural network and proposed FCN, which was the first successful application of convolutional neural network to semantic segmentation tasks. The FCN network achieves pixel-level classification by removing the fully connected layer and mapping the feature map output from the convolutional neural network to the output image of the same size as the input image. In 2015, Ronneberger et al. proposed the U-Net network [4], which uses an encoder-decoder structure. The encoder is used to extract the features, while the decoder is used to gradually recover the position and size of the feature image pixels. This model structure is able to fuse shallow features of images with deep features to obtain highly accurate segmentation results. Chen et al. proposed DeepLab network [5], which mainly uses dilated convolutions and multiscale pyramid pooling to improve segmentation accuracy. Niu et al. proposed the hybrid multi-attention network HMANet [6], which employs a channel attention module and introduces a region random playback attention module to reduce feature redundancy and improve the efficiency of the self-attentive mechanism by representing it in a regional manner. Chen et al. proposed DeepLabv3+ network [7], which employs a series of technical improvements, including atrous spatial pyramid pooling (ASPP), encoder-decoder structure, and hybrid loss function. Zhao et al. proposed the PSPNet network [8], which uses a pyramid pooling module to capture the global contextual information of the input image and combine it with local information to obtain better semantic segmentation results.

However, the complexity of road scenes and the limitations of current semantic segmentation models lead to a series of challenges in road scene segmentation. (1) Road scenes have complex diversity, including different road types, changes in lighting conditions, effects of weather conditions, and vehicle occlusion and overlap. All these factors will lead to a decrease in the accuracy and stability of the semantic segmentation model. (2) There are objects of different scales in the road scene, such as pedestrians, vehicles, and traffic signs. Current semantic segmentation models face difficulties in handing scale variation, and it is difficult to accurately segment objects with different scales, especially small-scale targets. (3) In autonomous driving and traffic monitoring systems, real-time segmentation results of road scenes are highly demanding. However, the current semantic segmentation models cannot be applied on embedded devices to achieve real-time segmentation, considering both the number of parameters and inference time. (4) In road scenes, the boundaries between some semantic categories may be unclear or blurred, and it is challenging for current semantic segmentation techniques to accurately capture and segment these blurred boundary regions. Therefore, how to effectively solve these challenges and improve the accuracy and robustness of road scene segmentation has become a hot spot and focus of current research.

In this paper, we propose a new lightweight semantic segmentation model based on LiteSeg [9] to address these problems. The work we have done is as follows:

1. The original model backbone network is replaced by the MobileOne [10] network. MobileOne is an efficient neural network that can effectively reduce the number of parameters and computation of the network. Also, to ensure the feature extraction capability of the model, a lightweight and flexible CA attention mechanism [11] is introduced to obtain more efficient feature information.
2. A multi-scale transposition attention module is proposed. This module can acquire vector features at different resolutions and fuse the features. A Transformer encoder module is also incorporated to operate between channels with the help of covariance matrices of key and query values, which combine the accuracy of global transform networks with the scalability of convolutional structures.
3. The SCNet module is proposed. This module processes spatial attention and channel attention in parallel to obtain richer feature information.

## 2 Related Work

As deep learning models continue to evolve, there is an increasing need to apply these models to real-world problems, thus requiring more and more models to be deployed on mobile devices. In order to interact with the real environment in real-time, semantic segmentation models need to have real-time processing capabilities and meet accuracy requirements. In recent years, the state-of-the-art lightweight semantic segmentation models can be divided into three main types: encoder-decoder structures, two-branch structures, and multi-branch structures. The model with an encoder-decoder structure extracts the features of the input image through the encoder and then maps these features back to the original image size through the decoder to achieve pixel-level classification. This structure allows for relatively fast inference while maintaining high accuracy. The model with a two-branch structure divides the network into two branches, one for the extraction of global contextual information and the other for capturing local details. This structure can effectively balance global and local information, improve segmentation accuracy, and enhance inference speed to a certain extent. The model with a multi-branch structure processes feature information at different scales separately by using multiple parallel branching networks. These branching networks can extract features in different sense field ranges and fuse them to obtain more comprehensive contextual information, thus improving the accuracy of semantic segmentation. The classification of lightweight semantic segmentation models is shown in Table 1.

1. Encoder-decoder architecture. In 2016, Paszke et al. proposed the ENet [12] model, which is the first semantic segmentation model that takes real-time into account, but its segmentation accuracy is low. In the same year, Eduardo et al. improved on ENet by proposing the ERFNet [13] model to obtain more information by interleaving the use of null convolution

and ResNet blocks. The RGPNet [14] model consists of an asymmetric encoder-decoder and an adapter, which helps to preserve and refine the distributed representation of multiple levels and facilitates the flow of gradients between different levels. The LiteSeg model proposed by Emara et al. explores a deeper version of the ASPP module and applies short and long residual connections and deeply separable convolution to provide a faster and more efficient model. The MSCFNet [15] model uses decomposed convolution blocks and asymmetric residual blocks of dilated convolution to construct the encoder and uses deconvolution instead of the high computational cost of the FPANet [16] model extracts high-level semantic information by aggregating spatial pyramids with feature pyramids and uses a bidirectional directed feature pyramid network to fuse feature information at different levels. Besides, the LETNet [17] model combines U-shaped Convolutional Neural Networks (CNN) with Transformer, the ELANet [18] model designs an effective lightweight attention-guided network, and the ELUNet [19] model provides an efficient and lightweight U-shaped network architecture. The EACNet [20] model uses convolutional decomposition to enhance the feature representation capability and robustness to rotating objects by using depth-oriented convolutional decomposition as a basic feature layer and point-by-point convolution for fusion. The CFPNet [21] model combines the Inception module and null convolution to extract feature maps and contextual information of various sizes. MobileOne is an ultra-lightweight backbone network for mobile devices that achieves significant improvements in latency and accuracy through the introduction of linear branching.

2. Double branch structure. In 2018, Yu et al. proposed BiSeNet [22], a bilateral segmentation network containing spatial and contextual paths, and they introduced a feature fusion module and an attention refinement module to further improve accuracy at an acceptable cost. To handle communication between parallel branches, the authors proposed BiseNetV2 [23] by adding an effective fusion layer to the BiSeNet model, which enhances the connection between the two paths. Despite the significant progress in speed and accuracy of BiseNetV2, there are still some redundancies in the initial downsampling phase and the fusion layer, which limit the information exchange between spatial and semantic branches. To address this issue, Faster BiSeNet [24] adopts a cleaner design that reduces redundant network architecture and enhances the relationship between the two branches. Aerial-BiSeNet [25] proposes a feature attention module and a channel attention-based feature fusion module based on the channel attention mechanism, effectively refining and combining features to improve the model's performance. Additionally, Poudel et al. proposed Fast-SCNN [26], which introduces a learning downsampling module based on the existing two-branch fast segmentation method to compute low-level features of multiple resolution branches simultaneously and combine high-resolution spatial details with low-resolution depth features. This method is suitable for low-memory embedded devices with efficient computational power.
3. Multi-branch structure. In 2018, Zhao et al. proposed ICNet [27] with multi-scale input, using few convolutions at high resolution and a deeper network at low resolution, and finally features fusion. In 2019, Li et al. proposed DFANet [28] to aggregate discriminative features through a series of subsidiary stages. DFANet is based on multi-scale feature propagation, which reduces the model parameters while maintaining a good perceptual field and enhancing the learning ability of the model. In the same year, Liu et al. proposed FDDWNet [29], which uses decomposition-expanded depth-separable convolution to learn feature representations from different scale receptive fields. The MSFNet [30] model designs a multiscale feature network consisting of an enhanced diverse attention module and an upsampling phase fusion module that uses high-level semantic information to complement low-level detail information

to improve prediction. In 2021, Fan et al. proposed the Short-Term Dense Concate (STDC) network [31], which constructs the basic modules of the STDC network by reducing the dimensionality of the feature map and using feature map clustering for image representation. NDNet [32] eliminates redundant information through pruning and is suitable for real-time segmentation tasks with narrow width and large depth. To further optimize the output resolution of the segmentation network, NDNet uses point-by-point convolution to connect feature maps, facilitating the aggregation of information from two different levels. DFFNet [33] proposes a lightweight multiscale component of the semantic pyramid module, which improves the efficiency of context encoding through depth decomposition.

**Table 1:** Lightweight semantic segmentation model classification

| Structure | Network |
|---|---|
| Encoder-decoder architecture | ENet, ERFNet, RGPNet, LiteSeg, FPANet |
| Double branch structure | BiseNet, BiseNet V2, Faster BiseNet, Fast-SCNN, Aerial-BiseNet |
| Multi-branch structure | ICNet, DFANet, DFFNet, MSFNet, FDDWNet |

## 3 Proposed Method

LiteSeg is one of the current excellent lightweight semantic segmentation models. It is designed based on the encoder-decoder architecture, ASPP, dilated convolutions, and depth-wise separable convolutions. By employing depth-wise separable convolutions and ASPP, the model reduces parameter count while improving segmentation accuracy. The use of dilated convolutions expands the receptive field, enabling better capture of object information at different scales. On the Cityscapes dataset, LiteSeg achieves an impressive mIoU accuracy of 67.81%. It is a lightweight, efficient real-time semantic segmentation model. However, we noticed that the LiteSeg model does not consider the positional information of features and the semantic correlations of long-distance features, resulting in difficulties in accurately segmenting object boundaries and small objects. Therefore, in this paper, we propose a lightweight semantic segmentation model with higher accuracy based on LiteSeg while maintaining the model volume, and the structure is shown in Fig. 1. Building upon LiteSeg, this paper utilizes the MobileOne backbone network module and incorporates the CA attention mechanism and SCNet attention module to extract feature information. Additionally, the multi-scale transposition attention coding module is used to extract long-range global features.

### *3.1 Feature Extraction Based on CA Attention Mechanism*

The CA attention mechanism is used to enhance the receptive field of deep neural networks, primarily by weighting the feature maps of different channels in the network. Its structure diagram is shown in Fig. 2. The CA attention module encodes channel relationships and long-term dependencies through precise location information, and the specific operations are divided into two parts: Coordinate information embedding and Coordinate Attention generation.

The CA Attention module encodes channel relationships and long-term dependencies through precise location information, and the specific operations are divided into two parts: Coordinate information embedding and Coordinate Attention generation.

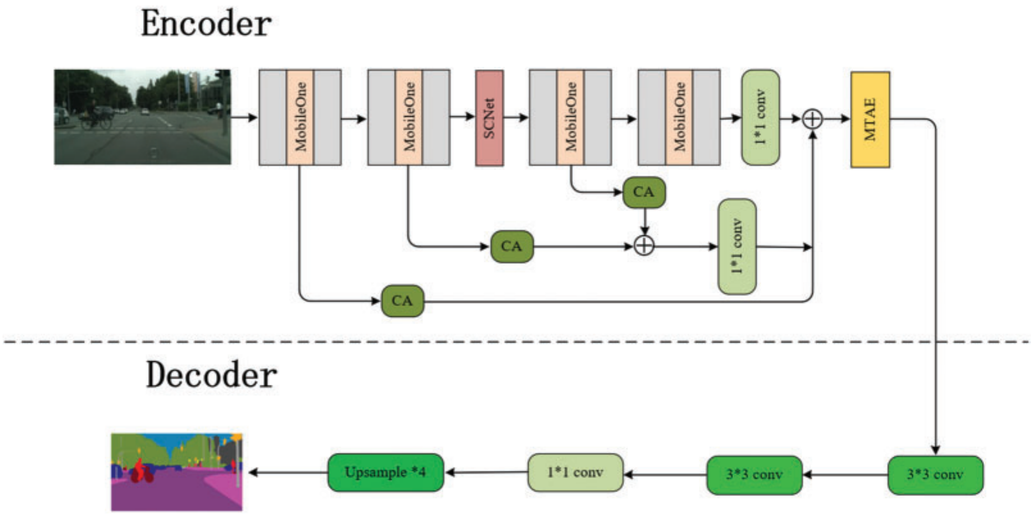

**Figure 1:** Improved LitSeg network structure

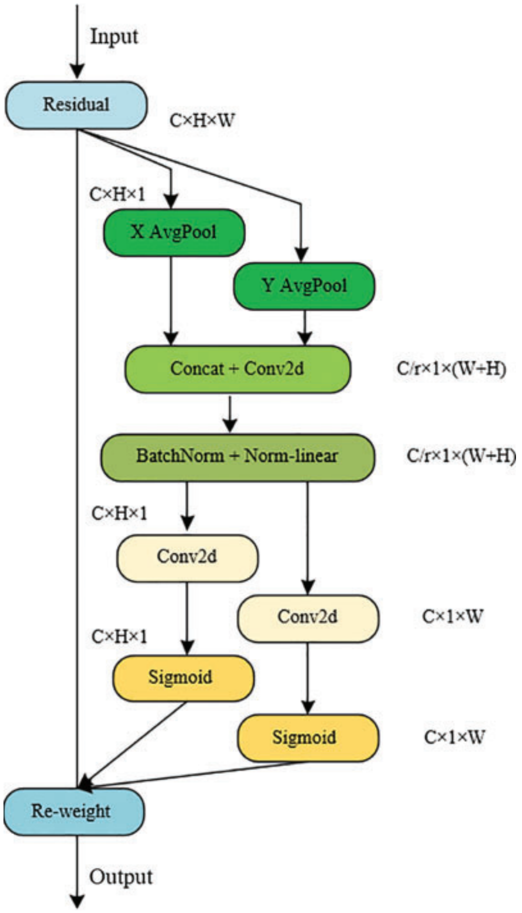

**Figure 2:** CA attention mechanism

Coordinate information embedding: the global pooling approach is used for the global encoding of spatial information for channel attention encoding, but it makes it difficult to maintain location information because it pushes global spatial information into the channel description. To induce the attention unit to capture distant spatial interactions with accurate location information, the global pooling is decomposed according to the formula in Eq. (1) and transformed into a one-to-one feature encoding operation.

$$Z_c = \frac{1}{H * W} \sum_{i=1}^{H} \sum_{j=1}^{W} x_c(i,j) \tag{1}$$

Specifically, given the input, each channel is encoded along the horizontal or vertical coordinates using pooling kernels of size $(H, 1)$ or $(1, W)$, respectively. This channel attention captures long-term dependencies along one spatial direction and preserves precise position information along the other spatial direction, which helps the network to locate the target of interest more accurately.

Coordinate Attention generation: After passing through the transformations in the information embedding, this part splices the above transformations and then uses the convolutional transform function to process them.

$$f = \delta\left(F_1\left([z^h, z^w]\right)\right) \tag{2}$$

Eq. (2) represents the concatenation operation along the spatial dimension, encoding spatial information in both the horizontal and vertical directions. It is then decomposed into two separate tensor sums along the spatial dimension. Utilizing two additional convolutional transforms, $z^h$ and $z^w$ are individually transformed into tensors with the same number of channels, as shown in Eqs. (3) and (4).

$$g^h = \sigma(F_h(f^h)) \tag{3}$$

$$g^w = \sigma(F_w(f^w)) \tag{4}$$

Finally, the output of the CA module can be expressed as shown in Eq. (5).

$$y_c(i,j) = x_c(i,j) * g_c^h(i) * g_c^w(j) \tag{5}$$

The global pooling layer in the network pools the input feature maps globally on average to obtain the global average of each channel. The channel weighting layer multiplies the original feature maps with the channel weights to obtain the weighted feature maps. Finally, the feature reconstruction layer reconstructs the weighted feature maps into the final feature maps. The addition of the CA module helps MobileOne extract more feature information at a very small additional computational cost.

### *3.2 SCNet Attention Module*

Combined attention modules have achieved wide application in the field of image processing, and in [34], a spatial, temporal, and channel attention module was proposed to achieve the extraction of spatio-temporal features using three attention modules. Since using three attention modules in series would add extra computation, this paper aims to balance the accuracy and speed of the model and thus designs the SCNet attention module. The SCNet attention module uses the spatial and channel attention mechanisms and performs feature extraction on the input separately in parallel, and finally fuses them using one-dimensional convolution. The structure of the SCNet attention module model is shown in Fig. 3.

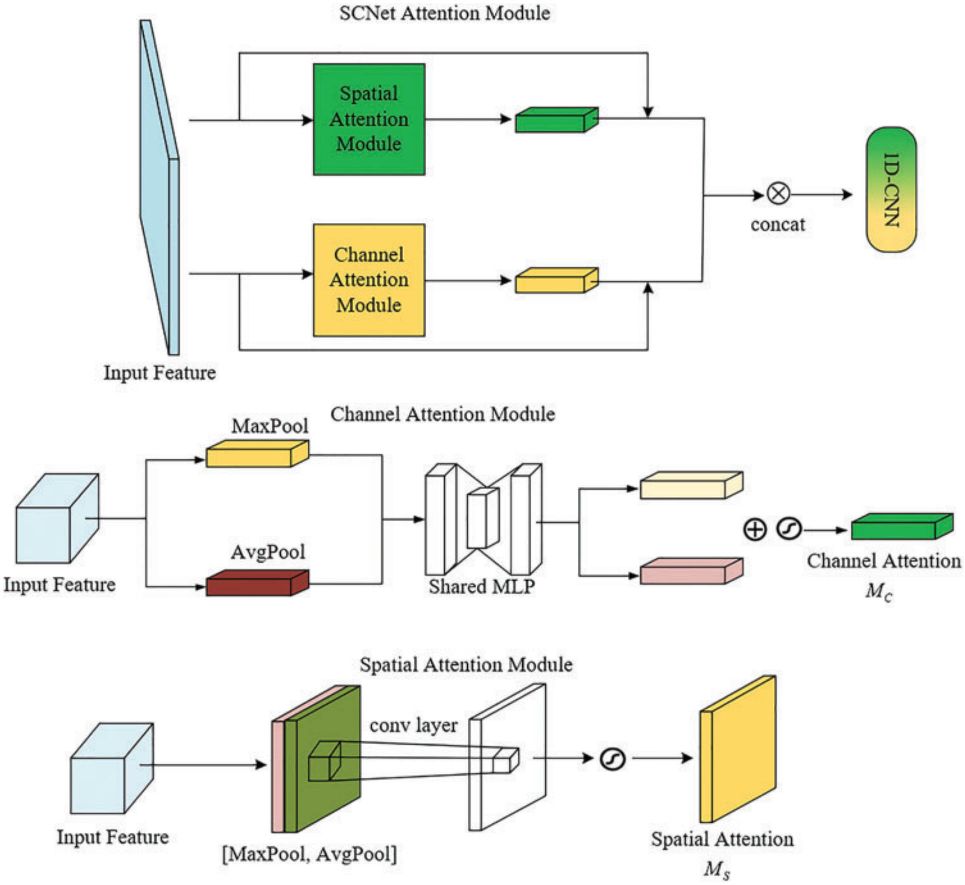

**Figure 3:** SCNet attention module

The SCNet attention module consists of a spatial attention module (SAM) and a channel attention module (CAM). The SAM module is capable of assigning different levels of attention to each region. The equations for this module are shown in Eq. (6).

$$M_s = \delta(MaxPool\,(f_{in}) + AvgPool(f_{in})) \tag{6}$$

where $f_{in}$ denotes the input features and $\delta(\cdot)$ denotes the Sigmoid activation function. The spatial attention map $M_s$ is multiplied with the input feature map to perform adaptive refinement in terms of residuals.

The channel attention module is used to extract channel features from the region feature map in the image frame. The formula for this module is shown in Eq. (7).

$$M_c = \delta(g_c(MaxPool\,(f_{in}) + AvgPool(f_{in}))) \tag{7}$$

where $\delta(\cdot)$ denotes the Sigmoid activation function, where $g_c$ denotes the multilayer perceptron (MLP), and $M_c$ denotes the channel attention map. The channel attention map $M_c$ is multiplied with the input feature map in a residual manner for adaptive refinement.

After the input features pass through the two attention modules separately, the concat function is used to concatenate the two feature maps in parallel, as shown in Eq. (8), where concat denotes the concatenation operation.

$$M = concat(M_s, M_c) \tag{8}$$

The improved SCNet attention module features the spatial and channel attention modules in parallel to extract the input features separately. The features of each module are multiplied with the input features in the form of residuals for adaptive refinement. The two features in parallel are then concatenated together and finally input to a 1 $D$ convolutional network for fusion.

### *3.3 Multiscale Transposed Attention Encoding Module*

In this paper, different regions of the feature map are segmented using the MTAE module, which encodes the feature map to extract multiscale features. The encoder model is used to extract long-distance global features. The encoder module is a component of the Transformer model, which can reduce the strict neighborhood constraint of graph convolution and focus on the connection between pixels that are physically far away, while also being able to focus on local information. The feature encoding module can associate more features between regions, establish long-distance feature dependencies, and extract richer features.

The traditional Transformer uses a self-attentive mechanism to interact with image blocks to model image data. However, the complexity of the self-attentive mechanism itself makes it difficult to handle high-resolution images. In this paper, the encoder module uses a transposed attention mechanism, which operates between channels of features with the help of covariance matrices of key and query values. The transposed attention mechanism combines the accuracy of traditional global transform networks with the scalability of convolutional structures with linear complexity in sequence length, thus allowing efficient processing of high-resolution images.

The mutual covariance attention mechanism is an improvement on the self-attentive mechanism, and the former is capable of handling high-resolution images. In the self-attentive mechanism, the input vectors can form a matrix $X$. The query matrix $Q$, the key matrix $K$, and the value matrix $V$ are obtained by multiplying $X$ with three learnable transformation matrices $W_q$, $W_k$ and $W_v$, respectively.

For each query vector $q_i$, calculate its dot product scores with all key vectors $k_j$ and input these scores into the softmax function to obtain a weight vector $w_i$, where each element represents the correlation between $q_i$ and the different key vectors $k_j$, as shown in Eq. (9).

$$w_i = softmax\left(\frac{QK^T}{\sqrt{d}}\right) \tag{9}$$

Due to the limitations of the self-attention mechanism, this paper adopts the mutual covariance attention mechanism to interact with image features. The mutual covariance attention is a transposed form of the self-attention mechanism, which improves the attention mechanism based on the mutual covariance matrix. In the self-attention mechanism, the attention score is first calculated using the query matrix $Q$, the key matrix $K$, and the value matrix $V$. Then, the values are weighted and summed by the attention scores to get the output. In contrast, in the mutual covariance attention, the mutual covariance matrix between features needs to be calculated first, as shown in Eq. (10).

$$K^T Q = W_k^T X^T X W_q \tag{10}$$

The formulation of the mutual covariance attention is shown in Eq. (11), where both $Q$ and $K$ are generated by the coding layer, $t$ denotes the learnable parameter, and $T$ denotes the transpose operation. The XCA module is shown in Fig. 4, and each mutual covariance attention is preceded by a LayerNorm layer, which serves to normalize all the data.

$$XC_{Attention}\left(Q, K, V\right) = V * softmax\left(\frac{K^{T}Q}{t}\right) \tag{11}$$

The multiscale perception module based on the transposed attention mechanism is shown in Fig. 5. From the figure, it can be seen that the module inserts residual connection structures with hierarchies in the residual units. The encoder structure is used as a filter in the module, while connecting different filter groups in a hierarchical residual-like manner. After chunking, the input is divided into three subsets: $x_1$, $x_2$, and $x_3$. Each feature has the same scale size, but the channels are 1/3 of the input features. The encoder first extracts features from a set of feature maps, and then the output features from the previous set are sent to the next set of encoder filters and another set of input feature maps. This process is repeated several times until all the input feature maps have been processed. Finally, all the feature maps are concatenated to obtain the fusion information. Due to the combination effect, many equivalent feature scales are obtained.

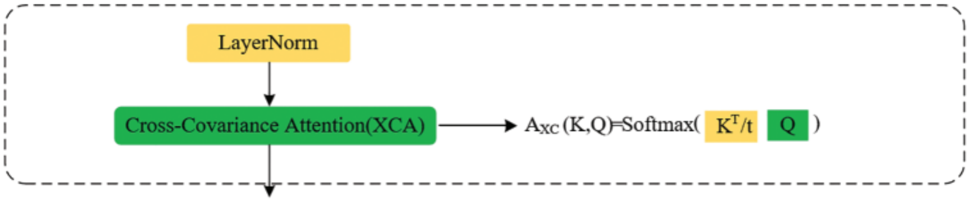

**Figure 4:** Reciprocal covariance attention

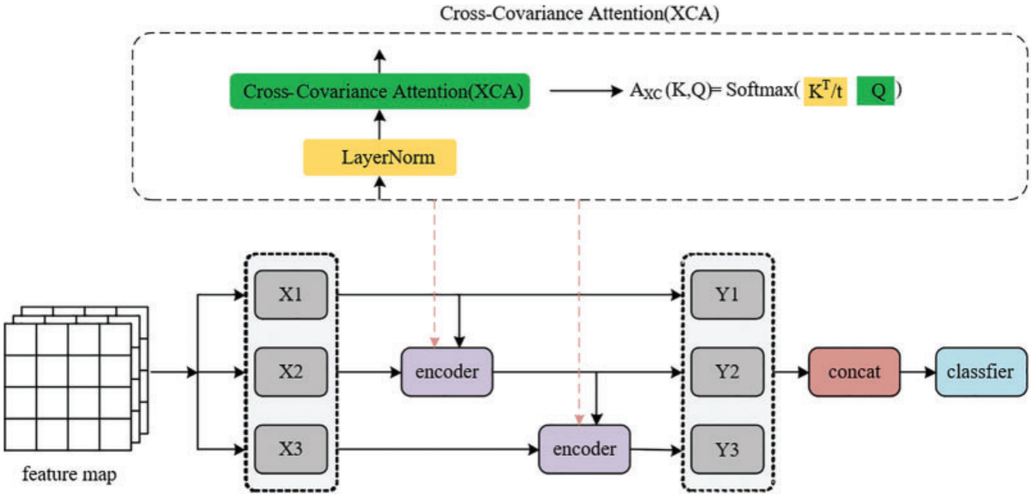
Cross-Covariance Attention(XCA)
Cross-Covariance Attention(XCA)
Axc (K,Q)= Softmax( K^T/t Q )
LayerNorm
feature map
X1
X2
X3
encoder
encoder
Y1
Y2
Y3
concat
classfier

**Figure 5:** Multi-level transposed attention encoding module

The module partitions the feature map into 3 subsets: $x_1$, $x_2$, and $x_3$, and then performs the operations using the transpose attention mechanism in the encoder, respectively. The formula is shown in Eq. (12).

$$y_i = \begin{cases} x_i, & i = 1 \\ A\left(x_i + y_{i-1}\right), & 2 \le i \le 3 \end{cases} \tag{12}$$

It is specified that each passing encoder is an operation $A_i$, and the output is $y_i$. Meanwhile, the output $y_{i-1}$ of $A_{i-1}$ is added to the feature subset $x_i$, and then input to $A_i$ to complete the feature extraction.

## 4 Experimental Results and Analysis

### *4.1 Experimental Environment Configuration*

The software and hardware environments for the experiments in this paper are shown in Table 2.

**Table 2:** Experimental environment configuration

| Software and hardware | Version and model |
|---|---|
| CPU | Intel(R) Core(TM) i7-8700K CPU @ 3.70 GHz |
| GPU | Nvidia GeForce RTX 1080Ti |
| Memory | 32 G |
| Operating system | Ubuntu20.04 |
| Frame | Pytorch1.7.1, CUDA10.1 |

Hyperparameter setting: Optimizer, SGD (stochastic gradient descent); momentum, 0.937; weight decay, 0.0005; learning rate, 0.001; epoch, 150.

### *4.2 Experimental Data and Evaluation Index*

In this paper, we use Cityscapes dataset as the experimental data. Cityscapes dataset is a large-scale dataset for computer vision, focusing on providing training and performance testing for autonomous driving environment perception models. It covers various street scenes, road scenes, and seasons, with a total of 5000 images. The dataset includes 2975 images in the training set, 500 images in the validation set, and 1525 images in the test set.

Before training, the Cityscapes dataset was divided into 19 classes, namely: road, sidewalk, building, wall, fence, pole, traffic light, traffic sign, vehicle, terrain, sky, person, rider, car, truck, bus, train, motorcycle, and bicycle, In the training process, the size of the images is scaled from $1024 * 2048$ to $321 * 512$ according to the server's computing power and time efficiency, and the number of images loaded in each batch is 8. The experiments use the pre-training weights of the official LiteSeg model to initialize the model parameters, and the other network parameters are kept constant during the training process.

The evaluation metric used in the experiment is mIoU, which is the average of the IoU of all categories. In semantic segmentation, the IoU value of a category is calculated as the ratio of the intersection of the set of pixels predicted by the semantic segmentation model as the set of pixels in that category to the real set of pixels in that category and the union set. The IoU is calculated as shown in Eq. (13). The mIoU is calculated as shown in Eq. (14).

$$IoU = \sum\nolimits_{i=0}^{k} \frac{p_{ii}}{\sum\nolimits_{i=0}^{k} p_{ij} + \sum\nolimits_{i=0}^{k} p_{ji} - p_{ii}} \tag{13}$$

$$mIoU = \frac{1}{k+1}\sum\nolimits_{i=0}^{k}\frac{p_{ii}}{\sum\nolimits_{j=0}^{k}p_{ij}+\sum\nolimits_{j=0}^{k}p_{ji}-p_{ii}} \tag{14}$$

where, $k$ is the number of categories, $k+1$ includes background categories, $i$ denotes the true value, $j$ denotes the predicted value, $p_{ij}$ denotes the prediction of $i$ to $j$, $p_{ii}$ denotes the prediction of $i$ to $i$, $p_{ji}$ denotes the prediction of $j$ to $i$.

Params refers to the number of parameters to be trained in the model, including all the weights and bias terms, and is used in this paper to measure the complexity of the model. Giga Floating Point Operations (GFLOPs) refers to the number of floating-point operations performed by the model when performing a single forward computation, in billions of operations per second, which is used in this paper to measure the computational complexity of the model.

### *4.3 Analysis of Experimental Results*

#### *4.3.1 Comparison of Different Algorithms*

To validate the effectiveness of the proposed algorithm in this paper, we compared it with LiteSeg, ICNet, ENet, ERFNet, BiSeNetV2, STDC1-Seg50, and SeaFormer [35] algorithms. The experimental results are shown in Table 3. The proposed algorithm in this paper increases the mIoU value by 2.46%, the Params value by 0.19M, and the GFLOPs value by 0.46 compared to the original LiteSeg, which proves that the improvements in this paper are effective. Compared to ICNet, the mIoU value increased by 11.57%, the Params value decreased by 4.86M, and the GFLOPs value decreased by 22.94. Compared to ENet, the mIoU value increased by 4.95%, the Params value increased by 1.40M, and the GFLOPs value decreased by 3.15. Compared to ERFNet, the mIoU value increased by 5.57%, and the Params value decreased by 0.25. Compared to BiSeNetV2, although the mIoU value decreased by 1.57%, our model size is much smaller, the Params value decreased by 39.43M, and the GFLOPs value decreased by 15.78. Compared to STDC1-Seg50, the mIoU value increased by 0.53%, the Params value increased by 6.58M, and GFLOPs value increased by 4.55. Compared to SeaFormer, although the mIoU value decreased by 0.07%, our model size is much smaller, the Params value decreased by 2.18M, and the GFLOPs value increased by 3.37. It can be seen that the proposed algorithm balances accuracy and model size and has the best overall performance among the eight algorithms.

**Table 3:** Comparison of metrics of mainstream algorithm

| Model | mIoU(%) | Params(M) | GFLOPs |
|---|---|---|---|
| LiteSeg | 68.57 | 1.63 | 4.91 |
| ICNet | 59.46 | 6.68 | 28.31 |
| ENet | 66.08 | 0.42 | 8.52 |
| ERFNet | 65.46 | 2.07 | 53.48 |
| BiSeNetV2 | 72.60 | 41.25 | 21.15 |
| STDC1-Seg50 | 70.50 | 8.40 | 0.82 |
| SeaFormer | 71.10 | 4.00 | 2.00 |
| Ours | 71.03 | 1.82 | 5.37 |

To further verify the validity proposed in this paper, the metrics on the 19 categories of datasets in Cityscapes were analyzed and compared, and the experimental results are shown in Table 4. The IoU values for sidewalk, building, wall, fence, pole, traffic light, terrain, car, train, motorcycle, and bicycle categories are higher than the other four models. The IoU values of the proposed algorithm in the road category are slightly lower than those of the ERFNet model. In the traffic sign and rider categories, the IoU values of the proposed algorithm are lower than those of the LiteSeg and ERFNet models. In the vehicle category, the IoU of the proposed algorithm is lower than the other four models. In the sky, bus, and track categories, the IoU value of the proposed algorithm is lower than that of the LiteSeg model. In the person category, the IoU of the proposed algorithm is lower than the LiteSeg and ENet models. The proposed model optimizes the feature extraction results by using attention mechanism and multi-scale structure and can extract higher quality feature information of low-resolution targets compared to the original model and other models. However, the resolution of targets such as vehicle, road, and traffic sign are higher, so the improvement of segmentation effect for them is relatively low.

**Table 4:** IoU comparison of dataset categories

| Category | Ours | LiteSeg | ICNet | ENet | ERFNet |
|---|---|---|---|---|---|
| Road | 0.9614 | 0.9591 | 0.9529 | 0.9588 | 0.9663 |
| Sidewalk | 0.8237 | 0.7787 | 0.7808 | 0.7725 | 0.8012 |
| Building | 0.8441 | 0.8186 | 0.8203 | 0.8256 | 0.8011 |
| Wall | 0.6120 | 0.4956 | 0.4177 | 0.5314 | 0.5463 |
| Fence | 0.4996 | 0.4728 | 0.3669 | 0.4860 | 0.4627 |
| Pole | 0.6478 | 0.5232 | 0.4215 | 0.5495 | 0.3945 |
| Traffic light | 0.5261 | 0.4817 | 0.4011 | 0.4650 | 0.4279 |
| Traffic sign | 0.6143 | 0.6475 | 0.5113 | 0.5652 | 0.6996 |
| Vegetation | 0.8411 | 0.8551 | 0.8509 | 0.8659 | 0.8457 |
| Terrain | 0.7198 | 0.6871 | 0.5958 | 0.6350 | 0.6422 |
| Sky | 0.9294 | 0.9297 | 0.8877 | 0.9123 | 0.8902 |
| Person | 0.7538 | 0.7952 | 0.6052 | 0.7656 | 0.7154 |
| Rider | 0.5169 | 0.5266 | 0.3622 | 0.4698 | 0.5620 |
| Car | 0.9225 | 0.8786 | 0.8359 | 0.8472 | 0.8120 |
| Truck | 0.6173 | 0.6306 | 0.4102 | 0.5649 | 0.5647 |
| Bus | 0.6918 | 0.7225 | 0.6198 | 0.6535 | 0.6215 |
| Train | 0.6243 | 0.6160 | 0.4157 | 0.5664 | 0.5480 |
| Motorcycle | 0.6164 | 0.5101 | 0.4594 | 0.4957 | 0.5940 |
| Bicycle | 0.7334 | 0.6987 | 0.5829 | 0.6245 | 0.5428 |
| mIoU | 0.7103 | 0.6857 | 0.5946 | 0.6608 | 0.6546 |

This article also analyzed and compared the Class mIoU metric of five algorithms, with 150 iterations in the experiment. The mIoU change curve is shown in Fig. 6, from which it can be seen that the proposed algorithm and LiteSeg present a stable growth trend (smooth curve fluctuation) with the increase of epoch and gradually stabilize when epoch reaches 140. ICNet, ENet and ERFNet show small fluctuations in mIoU values with the increase of epochs, and ENet has stabilized when epoch reaches 100, and ERFNet stabilizes only when epoch reaches 140, while the mIoU value of ICNet is not only the lowest but also always in oscillation during the training process.

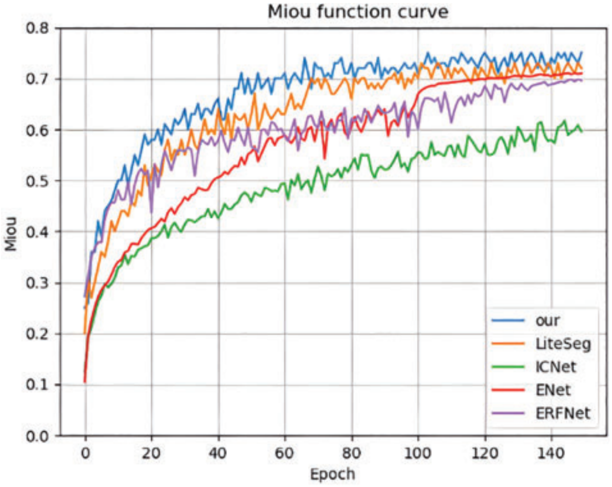

**Figure 6:** mIoU variation graph

### *4.3.2 Ablation Experiments*

To further verify the effectiveness of the added modules, they are compared with the original algorithm, and the experimental results are shown in Table 5. It can be seen that adding SCNet and MTAE can increase the mIoU value of LiteSeg by 4.96%. However, the Params and GFLOPs of the algorithm will also increase by 1.61M and 7.14, respectively. In this paper, considering the model size and computation, we use MobileOne instead of the original backbone network and introduce a lightweight and flexible CA attention mechanism. As a result, we achieve a 2.46% increase in the mIoU of the algorithm without significantly increasing the Params and GFLOPs.

**Table 5:** Ablation experiment

| Model | Class mIoU | Params | GFLOPs |
|---|---|---|---|
| LiteSeg | 68.57% | 1.63M | 4.91 |
| LiteSeg+SCNet+MTAE | 73.26% | 3.24M | 12.05 |
| LiteSeg-MobileOne | 69.53% | 1.12M | 2.43 |
| LiteSeg-MobileOne+CA+SCNet+MTAE | 71.03% | 1.82M | 5.37 |

### *4.3.3 Comparison of Visualization Results*

To directly validate the effectiveness of the proposed model in this paper, we compare the segmentation results of seven different algorithms on the Cityscapes dataset, as shown in Fig. 7. The yellow box indicates areas where other models have incomplete segmentation compared to our model.

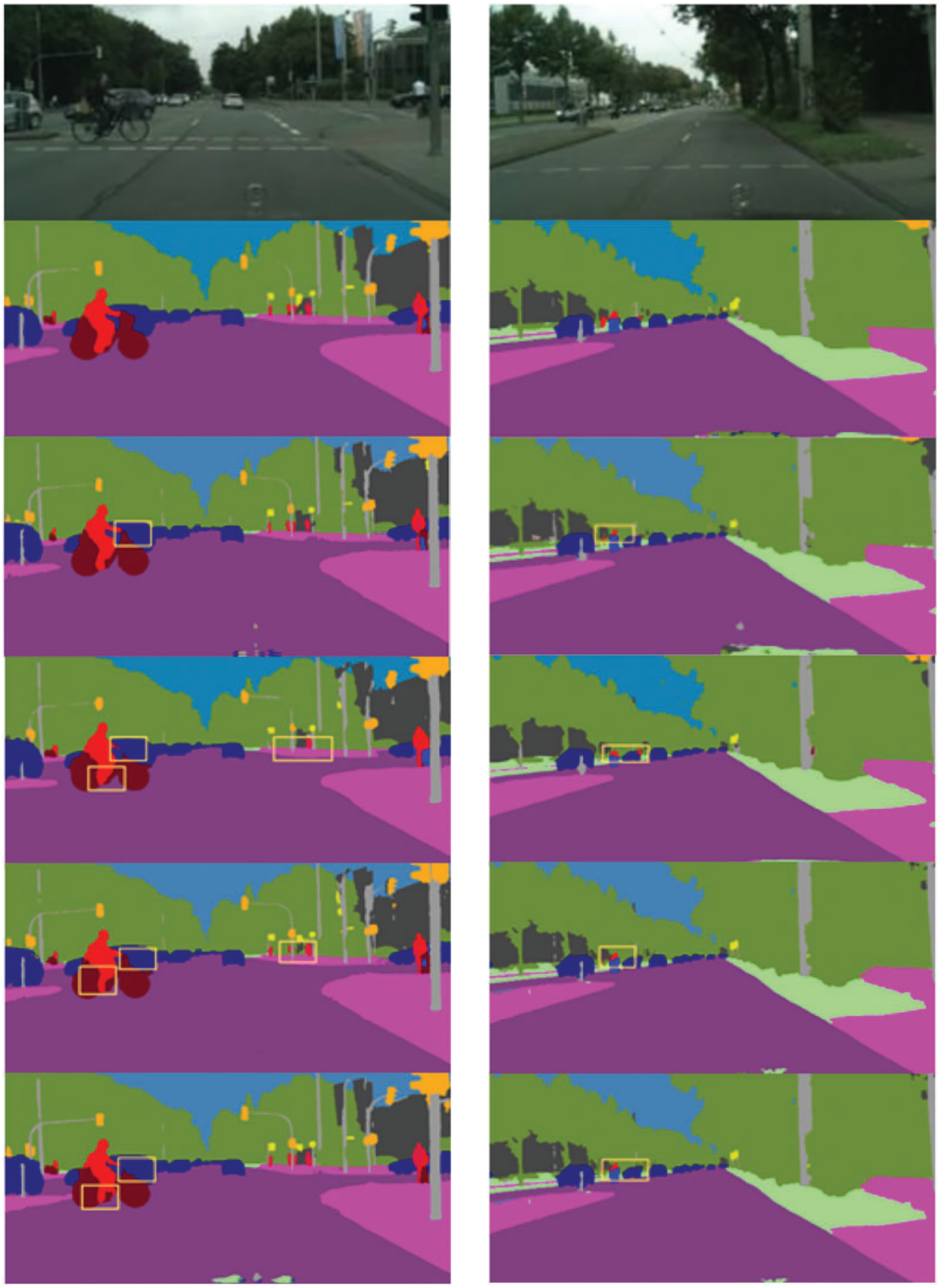

**Figure 7:** (Continued)

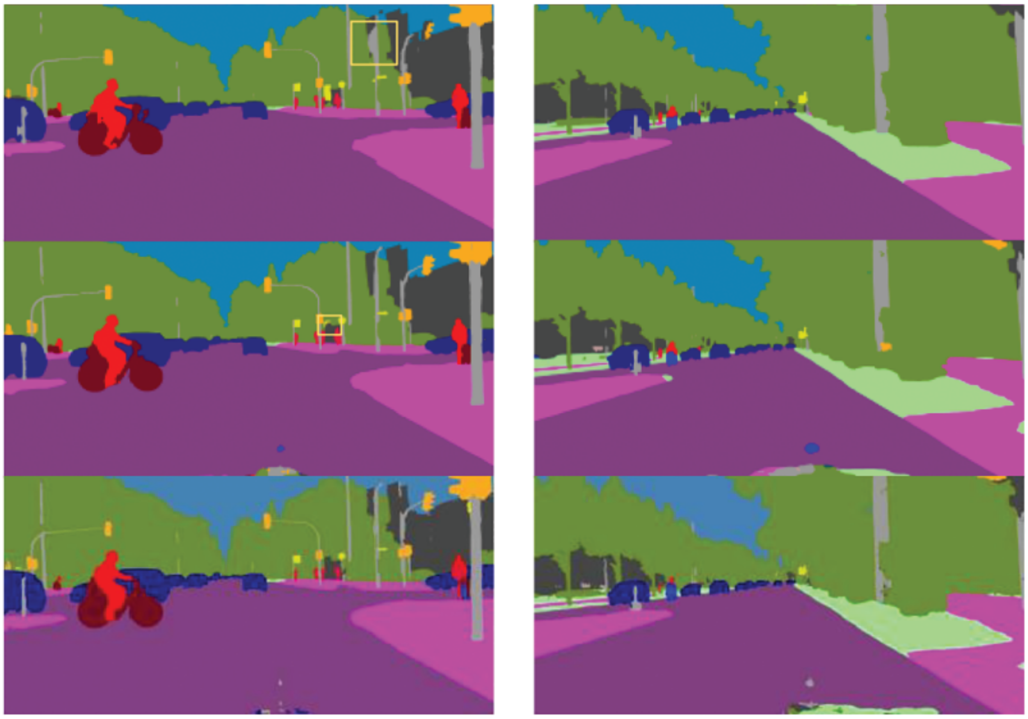

Figure 7: Visualization results of multiple models on Cityscapes validation set. From top to bottom 1-Input RGB image; 2-Our model; 3-LiteSeg model; 4-ICNet model; 5-ENet model; 6-ERFNet model; 7-BiSeNetV2; 8-STDC1-Seg50; 9-SeaFormer

To verify the generalization ability of the model proposed in this article, we conducted experiments on the KITTI dataset. We selected three models with similar segmentation accuracy as the proposed model in this article, BiSeNetV2, STDC1-Seg50, and SeaFormer, and visually compared their generalization ability, as shown in Figs. 8–10. The yellow box represents the part of the model that has not been segmented.

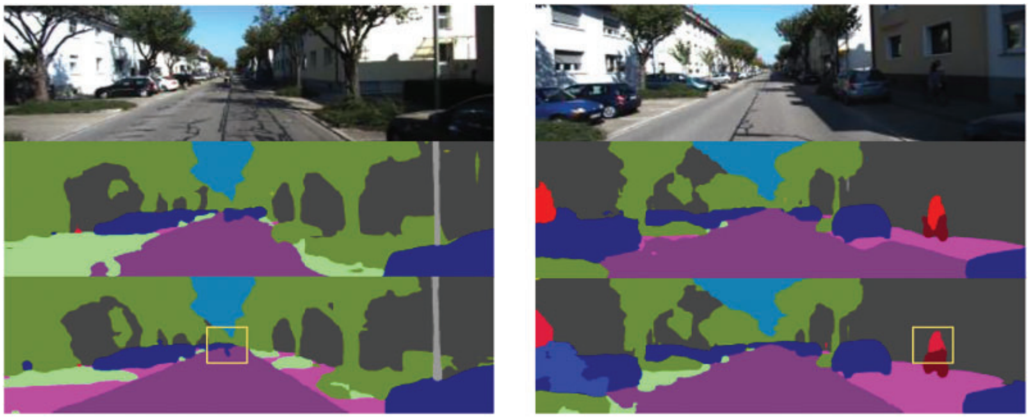

Figure 8: Visualization results of KITTI dataset. From top to bottom 1-Input RGB image; 2-Our model; 3-BiSeNetV2

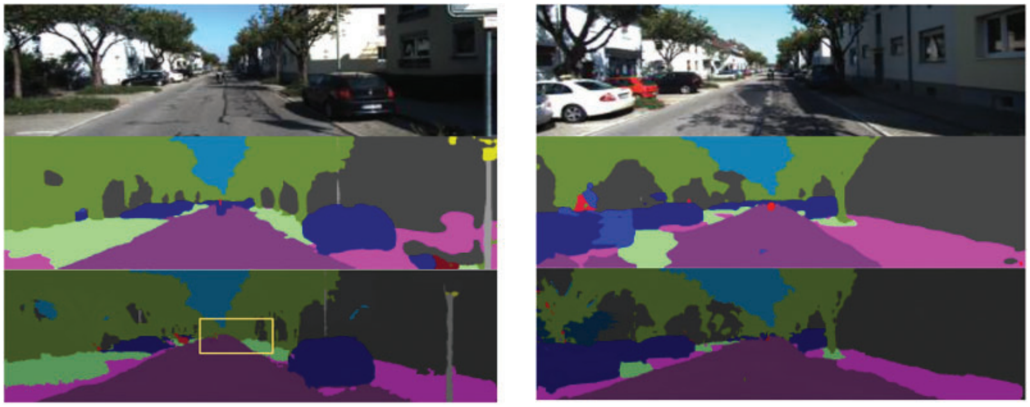

**Figure 9:** Visualization results of KITTI dataset. From top to bottom 1-Input RGB image; 2-Our model; 3- STDC1-Seg50

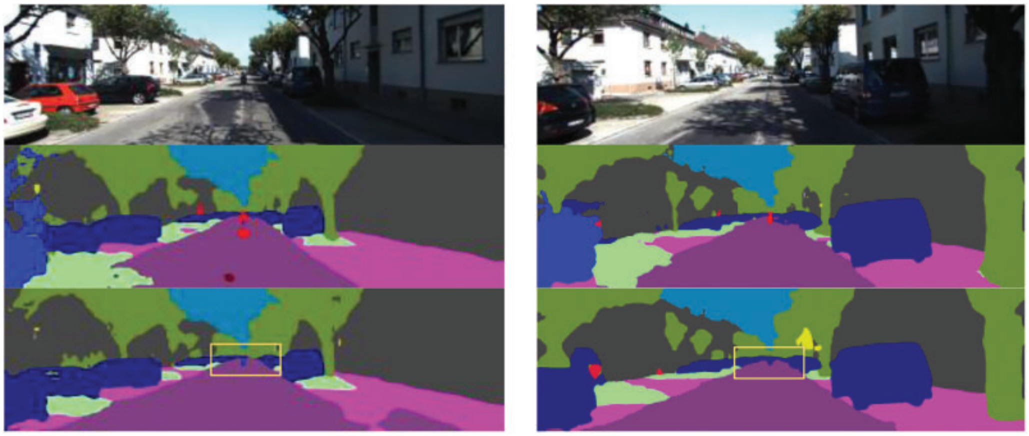

**Figure 10:** Visualization results of KITTI dataset. From top to bottom 1-Input RGB image; 2-Our model; 3-SeaFormer

## 5 Conclusion

To achieve the segmentation task of complex road scenes, this paper proposes a lightweight semantic segmentation algorithm based on LiteSeg. To reduce the size of the model, the MobileOne backbone network is used to extract features. A lightweight and efficient CA attention mechanism and SCNet module are used to enhance the feature extraction capability of the network, enabling it to focus on discriminative regions in the image and efficiently distinguish differences between different regions to achieve accurate segmentation. Furthermore, cross-dimensional feature fusion is achieved by adding the MTAE module, introducing the encoder module of Transformer in each scale space, and establishing jump connections in each dimensional space to fuse the features from different dimensional spaces. In this paper, the proposed algorithm is tested on the Cityscapes dataset, and the experimental results show that the algorithm improves the mIoU to 71.03% with only a slight increase in Params and GFLOPs compared to LiteSeg, while the IoU of all 12 categories on Cityscapes is higher than that of the LiteSeg algorithm, with only 7 categories having slightly lower IoU values. At the same time, this article will also test the generalization ability of the proposed model on the KITTI dataset, and the experimental results show that the proposed model has a certain degree of generalization ability. This demonstrates that the proposed algorithm meets the demand for accurate and fast segmentation of road images. However, the algorithm proposed in this paper also has some limitations. For example, for the compactness of the model, we abandon the downsampling process in the final stage, which makes the receptive field of the model insufficient to cover large target objects, resulting in limited improvement in segmentation accuracy for high-resolution objects. Due to the limitation of the computing power of the device, the image is cropped during the model training process, resulting in the loss of spatial details in the image, which leads to unsatisfactory segmentation of the boundary part of the image. Future work will conduct in-depth experiments on the power consumption of the model while improving segmentation accuracy. We will use knowledge distillation to further reduce the computational resources of the model and conduct experiments on different datasets captured on smart cars.

**Acknowledgement:** The authors are highly thankful to the National Natural Science Foundation of China, to the Innovation Fund of Chinese Universities Industry-University-Research, to the Research Project for Young and Middle-Aged Teachers in Guangxi Universities, and to the Special Research Project of Hechi University. This research was financially supported by the Project of Outstanding Thousand Young Teachers' Training in Higher Education Institutions of Guangxi, Guangxi Colleges and Universities Key Laboratory of AI and Information Processing (Hechi University), Education Department of Guangxi Zhuang Autonomous Region.

**Funding Statement:** The authors are highly thankful to the National Natural Science Foundation of China (No. 62063006), the Natural Science Foundation of Guangxi Province (No. 2023GXNSFAA026025), to the Innovation Fund of Chinese Universities Industry-University-Research (ID: 2021RYC06005), to the Research Project for Young and Middle-Aged Teachers in Guangxi Universities (ID: 2020KY15013), and to the Special Research Project of Hechi University (ID: 2021GCC028). This research was financially supported by the Project of Outstanding Thousand Young Teachers' Training in Higher Education Institutions of Guangxi, Guangxi Colleges and Universities Key Laboratory of AI and Information Processing (Hechi University), Education Department of Guangxi Zhuang Autonomous Region.

**Author Contributions:** The authors confirm contribution to the paper as follows: study conception and design: J. Peng; data collection: Y. Hou; analysis and interpretation of results: Q. Yang; draft manuscript preparation: Q. Yang. All authors reviewed the results and approved the final version of the manuscript.

**Availability of Data and Materials:** The data presented in this study are available upon request from the corresponding author.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

[1] J. Long, E. Shelhamer and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2015.

[2] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh *et al.,* "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[3] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

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

[5] 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 Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[6] R. Niu, X. Sun, Y. Tian, W. Diao, K. Chen *et al.,* "Hybrid multiple attention network for semantic segmentation in aerial images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–18, 2021.

[7] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. of the European Conf. on Computer Vision (ECCV)*, vol. 11211, pp. 801–818, 2018.

[8] H. Zhao, J. Shi and X. Qi, "Pyramid scene parsing network," in *2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp. 6230–6239, 2017.

[9] T. Emara, H. E. A. E. Munim and H. M. Abbas, "LiteSeg: A novel lightweight convnet for semantic segmentation," in *2019 Digital Image Computing: Techniques and Applications (DICTA)*, Perth, WA, Australia, pp. 1–7, 2019.

[10] P. K. A. Vasu, J. Gabriel and J. Zhu, "An improved one millisecond mobile backbone," arXiv preprint, arXiv:2206.04040, 2022. [Online]. Available: https://arxiv.org/abs/2206.04040

[11] Q. Hou, D. Zhou and J. Feng, "Coordinate attention for efficient mobile network design," in *2021 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, pp. 13708–13717, 2021.

[12] A. Paszke, A. Chaurasia, S. Kim and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," arXiv preprint, arXiv:1606.02147, 2016. [Online]. Available: https://arxiv.org/abs/1606.02147

[13] E. Romera, J. M. Alvarez, L. M. Bergasa and R. Arroyo, "Erfnet: Efficient residual factorized convnet for real-time semantic segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 263–272, 2017.

[14] A. Elahe, S. Marzban, A. Pata and B. Zonooz, "RGPNet: A real-time general purpose semantic segmentation," arXiv preprint, arXiv:1912.01394, 2019. [Online]. Available: https://arxiv.org/abs/1912.01394

[15] G. Gao, G. Xu, Y. Yu, J. Xie, J. Yang *et al.,* "MSCFNet: A lightweight network with multi-scale context fusion for real-time semantic segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 25489–25499, 2021.

[16] Y. Wu, J. Jiang, Z. Huang and Y. Tian, "FPANet: Feature pyramid aggregation network for real-time semantic segmentation," *Applied Intelligence*, vol. 52, no. 3, pp. 3319–3336, 2022.

[17] G. Xu, J. Li, G. Gao, H. Lu, J. Yang *et al.,* "Lightweight real-time semantic segmentation network with efficient transformer and CNN," in *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2023.

[18] Q. Yi, G. Dai, M. Shi, Z. Huang and A. Luo, "ELANet: Effective lightweight attention-guided network for real-time semantic segmentation," *Neural Processing Letters*, vol. 55, pp. 1–18, 2023.

[19] Y. Ai, J. Guo and Y. Wang, "ELUNet: An efficient and lightweight U-shape network for real-time semantic segmentation," *Journal of Electronic Imaging*, vol. 31, no. 2, pp. 023019, 2022.

[20] Y. Li, X. Li, C. Xiao, H. Li and W. Zhang, "EACNet: Enhanced asymmetric convolution for real-time semantic segmentation," *IEEE Signal Processing Letters*, vol. 28, pp. 234–238, 2021.

[21] A. Lou and M. Loew, "CFPNET: Channel-wise feature pyramid for real-time semantic segmentation," in *2021 IEEE Int. Conf. on Image Processing (ICIP)*, Anchorage, AK, USA, pp. 1894–1898, 2021.

[22] C. Yu, J. Wang and C. Peng, "BiSeNet: Bilateral segmentation network for real-time semantic segmentation," arXiv preprint, arXiv:1808.00897, 2018. [Online]. Available: https://arxiv.org/abs/1808.00897

[23] C. Yu, C. Gao and J. Wang, "BiSeNet V2: Bilateral network with guided aggregation for real-time semantic segmentation," *International Journal of Computer Vision*, vol. 129, no. 11, pp. 3051–3068, 2021.

[24] Q. Xu, Y. Ma, J. Wu and C. Long, "Faster BiSeNet: A faster bilateral segmentation network for real-time semantic segmentation," in *2021 Int. Joint Conf. on Neural Networks (IJCNN)*, Shenzhen, China, pp. 1–8, 2021.

[25] F. Wang, X. Luo, Q. Wang and L. Li, "Aerial-BiSeNet: A real-time semantic segmentation network for high resolution aerial imagery," *Chinese Journal of Aeronautics*, vol. 34, no. 9, pp. 47–59, 2021.

[26] R. P. K. Poudel, S. Liwicki and R. Cipolla, "Fast-SCNN: Fast semantic segmentation network," arXiv preprint, arXiv:1902.04502, 2019. [Online]. Available: https://arxiv.org/abs/1902.04502

[27] H. Zhao, X. Qi, X. Shen, J. Shi and J. Jia, "ICNet for real-time semantic segmentation on high-resolution images," in *Computer Vision-ECCV 2018*, Munich, Germany, pp. 418–434, 2018.

[28] H. Li, P. Xiong, H. Fan and J. Sun, "DFANet: Deep feature aggregation for real-time semantic segmentation," in *2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, pp. 9514–9523, 2019.

[29] J. Liu, Q. Zhou, Y. Qiang, B. Kang, X. Wu *et al.,* "FDDWNet: A lightweight convolutional neural network for real-time semantic segmentation," in *2020 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Barcelona, Spain, pp. 2373–2377, 2020.

[30] M. Pei, "MSFNet: Multi-scale features network for monocular depth estimation," arXiv preprint, arXiv:2107.06445, 2021. [Online]. Available: https://arxiv.org/abs/2107.06445

[31] M. Fan, S. Lai, J. Huang, X. Wei, Z. Chai *et al.,* "Rethinking BiSeNet for real-time semantic segmentation," in *2021 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, pp. 9711–9720, 2021.

[32] Z. Yang, H. Yu, Q. Fu, W. Sun, W. Jia *et al.,* "NDNet: Narrow while deep network for real-time semantic segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, pp. 5508–5519, 2020.

[33] X. Tang, W. Tu, K. Li and J. Chen, "DFFNet: An IoT-perceptive dual feature fusion network for general real-time semantic segmentation," *Information Sciences*, vol. 565, pp. 326–343, 2021.

[34] L. Shi, Y. Zhang, J. Cheng and H. Lu, "Skeleton-based action recognition with multi-stream adaptive graph convolutional networks," *IEEE Transactions on Image Processing*, vol. 29, pp. 9532–9545, 2020.

[35] Q. Wan, Z. Huang, J. Lu, G. Yu and L. Zhang, "Seaformer: Squeeze-enhanced axial transformer for mobile semantic segmentation," arXiv preprint, arXiv:2301.13156, 2023.