applied sciences

MDPI

Article

Detail Guided Multilateral Segmentation Network for Real-Time Semantic Segmentation

Qunyan Jiang, Juying Dai *, Ting Rui, Faming Shao, Ruizhe Hu, Yinan Du and Heng Zhang

Department of Mechanical Engineering, College of Field Engineering, Army Engineering University of PLA, Nanjing 210007, China

* Correspondence: dinajy2001@aeu.edu.cn

Abstract: With the development of unmanned vehicles and other technologies, the technical demand for scene semantic segmentation is more and more intense. Semantic segmentation requires not only rich high-level semantic information, but also rich detail information to ensure the accuracy of the segmentation task. Using a multipath structure to process underlying and semantic information can improve efficiency while ensuring segmentation accuracy. In order to improve the segmentation accuracy and efficiency of some small and thin objects, a detail guided multilateral segmentation network is proposed. Firstly, in order to improve the segmentation accuracy and model efficiency, a trilateral parallel network structure is designed, including the context fusion path (CF-path), the detail information guidance path (DIG-path), and the semantic information supplement path (SIS-path). Secondly, in order to effectively fuse semantic information and detail information, a feature fusion module based on an attention mechanism is designed. Finally, experimental results on CamVid and Cityscapes datasets show that the proposed algorithm can effectively balance segmentation accuracy and inference speed.

Keywords: semantic segmentation; feature fusion; trilateral parallel; graph convolution

Citation: Jiang, Q.; Dai, J.; Rui, T.; Shao, F.; Hu, R.; Du, Y.; Zhang, H. Detail Guided Multilateral Segmentation Network for Real-Time Semantic Segmentation. *Appl. Sci.* **2022**, *12*, 11040. https://doi.org/10.3390/app122111040

Academic Editor: Byung-Gyu Kim

Received: 20 August 2022
Accepted: 26 October 2022
Published: 31 October 2022

1. Introduction

Semantic segmentation is an important research direction in the field of computer vision. The task of semantic segmentation is to classify the input image at the pixel level. This technology is widely used in the field of medical images and unmanned driving, which has important research significance for human production and life.

In recent years, deep learning has made great progress in the field of computer vision, but previous models often sacrifice the computing power of hardware devices to improve the accuracy of models, which is not conducive to the deployment of models on edge devices and will lead to excessive delays in the processing of tasks. To solve this problem, researchers have mainly proposed two solutions. One is to compress the depth model using methods such as model pruning and quantification [1–3]. Another solution is to design a lightweight semantic segmentation model [4–8], so that the model can achieve the goal of efficient semantic segmentation. SegNet [4] adopts a lightweight network and skip connection to realize rapid segmentation of objects. ICNet [5] uses image cascading to accelerate semantic segmentation. BiSeNet [6] designs a bilateral segmentation network, a spatial path and a context path to retain high-resolution features and sufficient receptive fields. BiSeNetv2 [7] designs a more efficient bilateral segmentation network using fewer channels and a fast down-sampling strategy. STDC [8] proposes a detail aggregation module, which improves the speed of semantic segmentation by integrating the learning of spatial information into the low level in a single stream way. Although their work has achieved fast segmentation speed, the accuracy of their network is far behind that of large networks, especially for small and thin objects. With the increase of the number of network layers, the edge and other details of the segmented object will be gradually lost,

especially for small, segmented objects, the loss of details will be more serious. To solve this problem, inspired by BiSeNet v2 and Gated-SCNN [9], this paper proposes a detail guided multilateral segmentation network (DGMNet) for real-time semantic segmentation. The network is composed of three paths to maximize the use of spatial detail information, high-level semantic information and context information. Figure 1 shows the superiority of the proposed algorithm. In general, the contributions of this paper are as follows:

- A context fusion strategy guided by spatial details is proposed, which effectively reconstructs the spatial information lost in the global environment, so that the detailed information can still be retained in the deeper features, and improves the segmentation performance of small and thin objects.
- A semantic information supplement path is designed to further supplement the semantic information of the lightweight network, increase the connection between local information and global information and further improve the accuracy of image segmentation.
- A three-path feature fusion module based on full attention is designed, so that the features with different information output from the three paths can be effectively fused, and the features are refined to guide the network to extract more valuable features, to improve the image segmentation ability.
- DGMNet achieved good segmentation results on Cityscapes [10] and CamVid [11] datasets. Specifically, for the Cityscapes dataset with a resolution of 512 × 1024 and the CamVid dataset with a resolution of 960 × 720 on the NVIDIA RTX 2080Ti, the mIoU value can reach 76.9% and 74.8% at 141.5FPS and 121.4FPS, respectively.

The rest of this paper is organized as follows. In Section 2, the related work of semantic segmentation is introduced. In Section 3, the specific structure of the proposed DGMNet is described in detail. In Section 4, the effectiveness of the DGMNet is verified via comparative experiments. In Section 5, the paper is summarized.

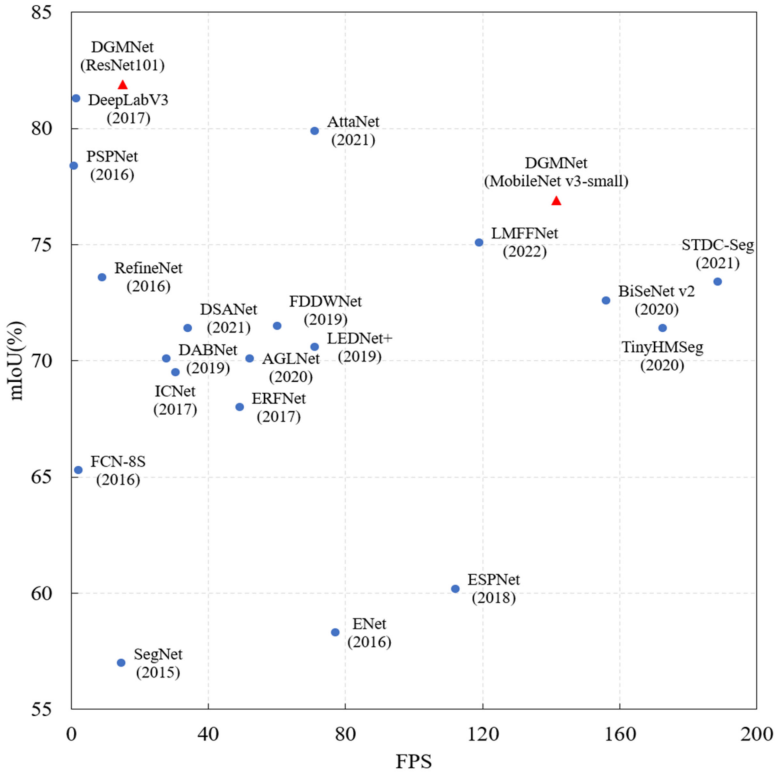


Figure 1. Comparison of the reasoning speed (FPS) and mIoU (%) of different networks on the test dataset of Cityscapes.

2. Related Works

The rapid development of deep neural networks has greatly promoted the research of semantic segmentation. More and more researchers have proposed different strategies to make the segmentation model achieve a balance between speed and accuracy. Next, the generic semantic segmentation method and real-time semantic segmentation method are introduced.

2.1. Generic Semantic Segmentation

Traditional semantic segmentation algorithms usually extract the color value, gray value and geometric shape features of the image to segment the image into several non-intersecting regions and then annotate these segmented regions to get the image segmentation. This kind of segmentation effect is often not ideal. Compared with traditional semantic segmentation algorithms, the semantic segmentation algorithm based on deep learning can not only extract the color, texture and shape information of the image, but also extract the high-level semantic information of the image. This kind of algorithm often has higher segmentation accuracy. Classical semantic segmentation algorithms based on a convolutional neural network include FCN [12], PSPNet [13], U-Net [14], DeepLab series [15–17], etc.

FCN [12] replaces the fully connected layer in the CNN network with a fully convolutional layer to construct an end-to-end, pixel-to-pixel semantic segmentation network. FCN opens a new idea for the field of semantic segmentation, but the network does not consider the relationship between pixels and lacks spatial consistency, which will cause the misjudgment of categories and the segmentation results to not be fine enough. PSPNet [13] improves the segmentation problem of the FCN network and proposes a spatial pyramid module to extract the context information and multi-scale information on the dilation backbone, reducing the probability of false segmentation of image categories in the FCN network. DeepLab series algorithm [15–17] is a semantic segmentation model developed by the Google team based on CNN. The latest algorithm is DeepLab v3+ [17]. DeepLabv3+ does not use full connection CRF and proposes deep separable convolution. The algorithm combines atrous spatial pyramid pooling (ASPP) with the Encoder-Decoder model and applies Xception and depthwise separable convolution to ASPP and a decoder, which can greatly reduce the amount of computation while maintaining the performance. Both the extended backbone network and the encoder-decoder structure can learn detailed information and high-level semantic information at the same time, but this method often has a high computational cost. In this paper, we propose an efficient architecture that achieves a good balance between speed and accuracy.

2.2. Real-Time Semantic Segmentation

Because many previous segmentation models cannot achieve real-time performance and cannot meet real-time application scenarios, efficient lightweight networks have been constantly proposed in recent years. ENet [18] is a lightweight real-time segmentation model based on SegNet [4], which is the first network to realize real-time semantic segmentation. Aiming at the problem of the low timeliness of previous deep neural networks, a new effective deep neural network is proposed. The bottleneck module used in this method has a jump connection structure, which can enhance the expression ability of features, so it can greatly improve the inference speed without much loss of accuracy. However, it gives up the final stage of the model, which is not large enough for the perception field of large objects, which will lead to poor recognition ability. DFANet [19] adopts a lightweight backbone to reduce the computational cost and design a cross-level feature aggregation module to transfer semantic and spatial information between different network layers, so as to solve the problem of spatial information loss caused by the increase of the number of network layers. DFNet [20] uses a “partial order pruning” algorithm to obtain a lightweight backbone and efficient decoder. Researchers also design a multi-path scheme to achieve the balance between segmentation speed and accuracy. ICNet [5] combines

medium-resolution and high-resolution features, considers segmentation accuracy and uses cascading strategies to accelerate real-time image semantic segmentation. BiSeNet v1 [6] and BiSeNet v2 [7] propose two flow paths for low-level details and high-level context information, respectively.

In this paper, we propose an efficient lightweight backbone to provide different receptive field information and high-level semantic information. In addition, we set up a low-cost detail guidance path and a semantic information supplement path to guide the backbone network to learn low-level details and supplement semantic information, so that the network can achieve a balance between speed and accuracy.

3. Detail Guided Multilateral Segmentation Network

3.1. Network Structure

The semantic segmentation network is composed of three paths: context fusion path, detail guidance path and semantic information supplement path. The context fusion path is used to extract the main features for semantic segmentation, the spatial detail guidance path is used to guide the context path to learn and retain the spatial detail information in the deeper network, and the semantic information supplement path provides the connection between the local information and the global information to supplement the semantic information. The overall framework of DGMNet is shown in Figure 2.

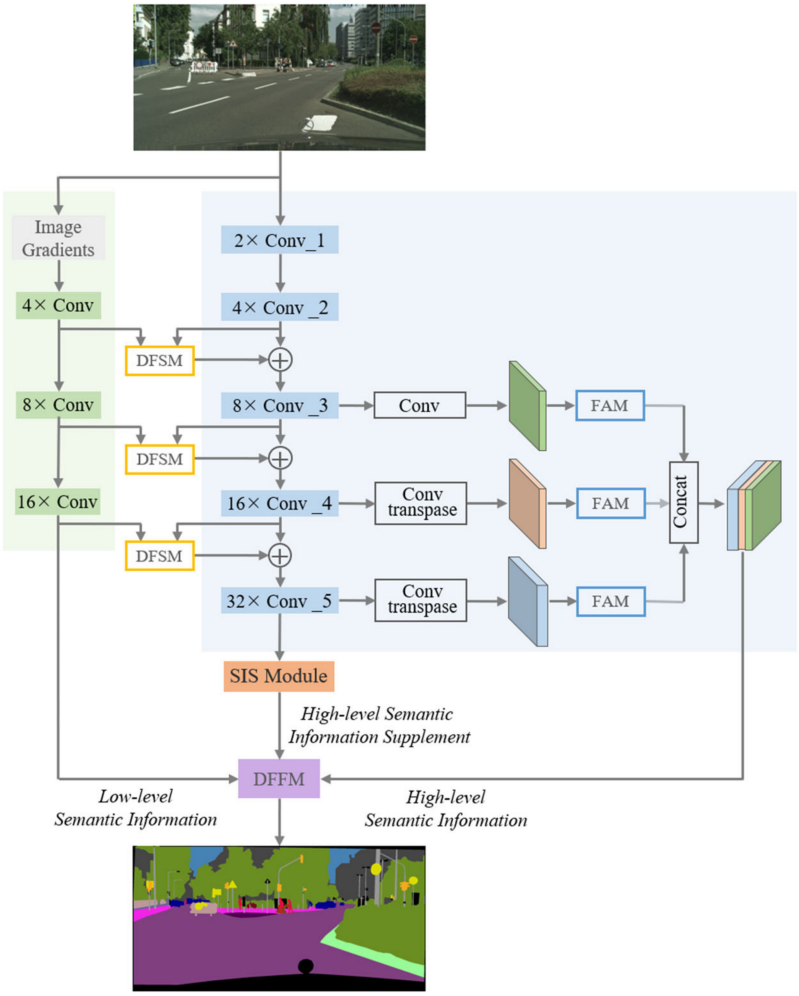


Figure 2. DGMNet network structure.

First, the input is passed to the context path and the detail guidance path. In the detail guidance path, the Gaussian Laplacian operator is used to extract the edge information of the image, and the feature maps of different sizes containing rich detail information are

obtained after a simple convolution operation. The features obtained by the lightweight backbone network and the detail guidance path are input into the feature interaction module, and the output features are fused with the features of the lightweight backbone network, so that the network can learn more detailed information. In the context path, the outputs of different layers of the lightweight backbone network will pass through the context feature fusion network, so as to extract features with rich semantic information and a large receptive field. At the same time, the convolutional features of the last layer of the backbone network are used to construct the graph structure in the semantic information supplement path. Then, the three features are effectively fused through the feature fusion module, and the transposed convolution is used for decoding. Finally, segmentation is carried out according to the features with rich semantic information and detailed information extracted from the network.

3.2. Network Structure

3.2.1. Context Fusion Path

In order to obtain various features with rich feature information and reduce the consumption of hardware resources by the model, a lightweight backbone network is used to encode visual features in the context path to obtain feature maps with rich semantic information and large receptive fields. The last three output features of different scales and levels extracted by the lightweight backbone network are fused, so that the model can obtain more context information when performing semantic segmentation on the original input image. In addition, in order to improve the segmentation performance, the proposed full attention module (FAM) is used in the multi-scale fusion to refine the output features.

3.2.2. Detail Information Guidance Path

There are multi-scale segmented objects in the image. With the deepening of the network layers, the detail information declines with the deepening of the model layers. Compared with the larger segmented objects, the loss of detail information of small objects is more severe with the increase of the model layers. In order to improve the segmentation accuracy of small and thin objects and produce fine segmentation results near the boundary, we introduce the detail information guidance path.

In the Figure 3, the image gradient represents the image edge detail information extracted via the Gaussian Laplace operator, and a feature map of size 1/2 is obtained. The edge information is used to further guide the lightweight backbone network to perform feature learning on the detailed information of the segmented object, to further improve the segmentation accuracy of fine or small objects. Image Gradient gets a feature map of size 1/4 after the operation of a convolution layer with step size 2. After the feature map and the feature map of the same size of the lightweight backbone network are input to the detail feature supplement module (DFSM) at the same time, a feature containing rich bottom-level detail information is obtained, and then, the information is fused with the input of the lightweight backbone network to supplement the detail information lost when the network layers are deepened. Similar operations are also performed after the Conv_3 module and Conv_4 module to obtain feature maps with richer edge details, so as to improve the segmentation accuracy of the network for thin and small objects in the input image. The FAM module is formulated as follows:

$$\begin{cases} X_c^h(h) = \frac{1}{W} \sum\limits_{0 \le i < W} x_c(h,i) + Max\big(x_c(h)\big), \\ X_c^w(h) = \frac{1}{W} \sum\limits_{0 \le j < H} x_c(j,w) + Max\big(x_c(w)\big) \end{cases} \tag{1}$$

$$f = \delta\Big(F_1\Big(\Big[X^h, X^w\Big]\Big)\Big) \tag{2}$$

$$\begin{cases} F^h = \sigma\left(F_h\left(f^h\right)\right), \\ F^w = \sigma\left(F_w\left(f^w\right)\right), \\ F^C = \sigma\left(X^{\mathbf{c}}_{\mathbf{max}} + X^{\mathbf{c}}_{\mathbf{avg}}\right) \end{cases} \tag{3}$$

where $\frac{1}{W}\sum_{0\le i<W} x_c(h,i)$ denotes the average pooling of the h-th row of the c-th channel of the input feature $X \in \mathbb{R}^{C\times W\times H}$; $\frac{1}{W}\sum_{0\le j<H} x_c(j,w)$ denotes the average pooling of the w-th column of the c-th channel. $Max\left(x_c(h)\right)$ represents the max-pooling of the h-th row of the c-th channel of the input feature; $Max\left(x_c(w)\right)$ refers to the max-pooling of the w-th column of the c-th channel of the input feature. Where $[\cdot,\cdot]$ represents the join operation of $X^h_c(h) \in \mathbb{R}^{C\times W\times 1}$ and $X^w_c(h) \in \mathbb{R}^{C\times 1\times H}$ that will be generated along the spatial dimension, δ is a non-linear activation function, and $F_1(\cdot)$, $F_h(\cdot)$ and $F_w(\cdot)$ are 1×1 convolution. $X^{\mathbf{c}}_{\mathbf{max}}$ and $X^{\mathbf{c}}_{\mathbf{avg}}$ represent maximum pooling and average pooling of input features, respectively. $f^h \in \mathbb{R}^{C/r\times H\times 1}$ and $f^w \in \mathbb{R}^{C/r\times 1\times W}$ divide f into two independent tensors along with the two-dimensional directions. The outputs F^h, F^w and R^C are attention feature maps. The output of the FAM module is as follows:

$$y_c(i,j) = F^h_c(i) \times F^w_c(j) \times x_c(i,j) \tag{4}$$

$$Y = y_c \times F^C \tag{5}$$

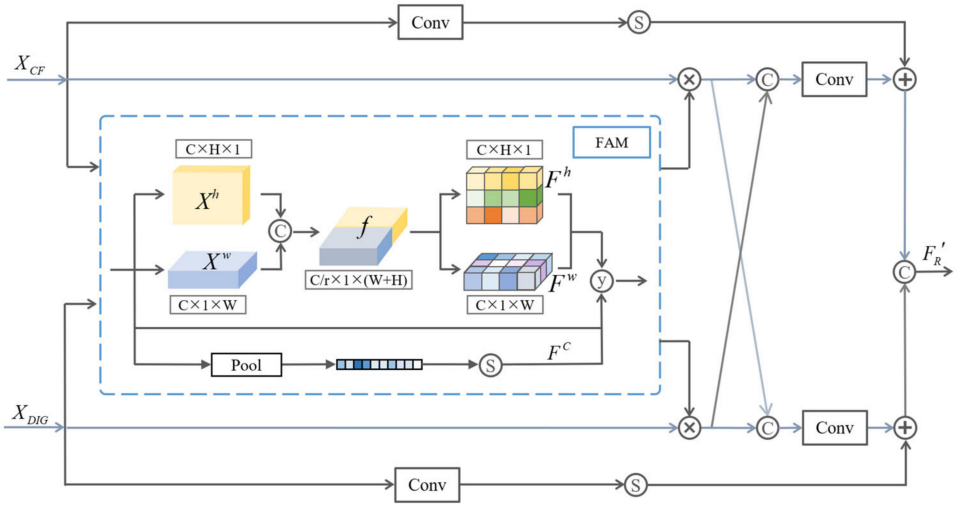


Figure 3. Detail feature supplement module.

3.2.3. Semantic Information Supplement Path

The stacking of convolutions has limited connection between local information on the graph, which limits the effective receptive field of the network. In order to strengthen the connection between local information and global information and effectively expand the range of the receptive field, we propose a scheme using a graph convolution neural network to further supplement the semantic relationship (capture long-distance dependency) between regions of a relatively long distance on the graph and further promote the network's learning in the current context. Inspired by GAS [21], to improve the global reasoning ability, we add a graph convolution path to obtain supplementary semantic information. Figure 4 shows the detailed structure of SIS-path with graph convolution.

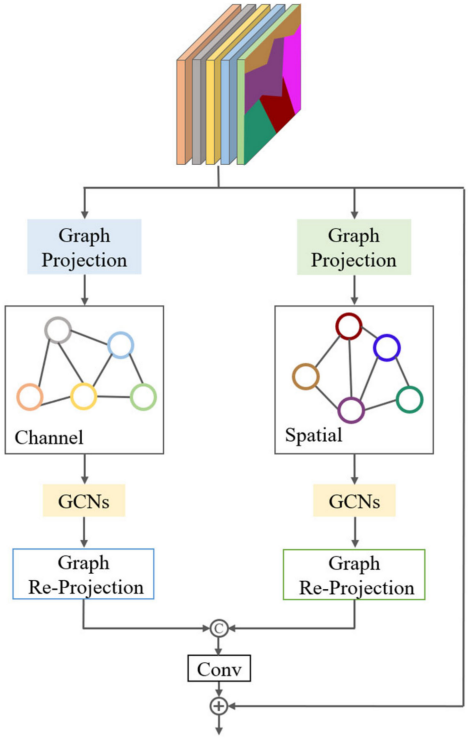


Figure 4. Semantic information supplement module (SIS Module).

We use the features of the output of the last layer of the backbone network of MoblieNet v3 [22] in the context path to build the graph structure of the SIS-path. First, the feature map is mapped into the graph, and pixels with similar features are assigned to the same vertex to determine the vertices of the graph structure. The edges of the graph structure are used to measure the similarity between the vertices. Secondly, the convolution on the graph structure is used to update the vertex features based on the feature information of the edges. Finally, the updated features are interpolated into the feature map via reprojection to recover the pixel-to-vertex assignment. Thus, the connection between local and global features is constructed and the dependencies between regions are encoded. On this basis, we construct associations between different layers of the feature map to associate channels with different features and further improve the network's ability to perform global learning. First, we assign channels with similar features to the same vertex to construct another graph structure. The edges of the graph structure measure the similarity between vertices. The encoding process is the same as GAS. The formula in the semantic information supplement path is expressed as follows:

$$\begin{cases} Y_{GCNS} = GCNs(graph1(X_C)), \\ Y_{GCNC} = GCNs(graph2(X_C)) \end{cases} \tag{6}$$

$$\begin{cases} Y_{ST} = Graph2Tensor(Y_{GCNS}), \\ Y_{CT} = Graph2Tensor(Y_{GCNC}) \end{cases} \tag{7}$$

$$Y_{SIS} = X_{CP} \oplus (conv(concat(Y_{ST}, Y_{CT}))) \tag{8}$$

where $X_C \in \mathbb{R}^{C \times H \times W}$ represents the output of the last layer of the CF-path, which is the input of the SIS-path, $graph1(\cdot)$ the transformation of the input tensor into the spatial graph structure feature $G_S \in \mathbb{R}^{N \times D}$, $graph2(\cdot)$ is the transformation of the input tensor into the channel graph structure feature $G_C \in \mathbb{R}^{N \times D}$, $GCNs$ represents the graph convolutional layer, $Graph2tensor(\cdot)$ represents the transformation of graph structure features into tensor, $conv$ represents 1×1 convolution, $\oplus$ represents the element-wise sum, and $concat$ represents the channel-wise concatenation.

3.2.4. Three-Path Feature Fusion Module

The detail guided path and context path are different in the level of feature representation. The output of the detail guided path contains more low-level detail information, while the output of the context path is high-level semantic information. The two kinds of information complement each other well, and the output of the semantic information supplement module provides semantic information from another perspective. Therefore, in order to effectively fuse the information extracted from the three paths, the TFFM module is proposed, whose specific structure is shown in Figure 5.

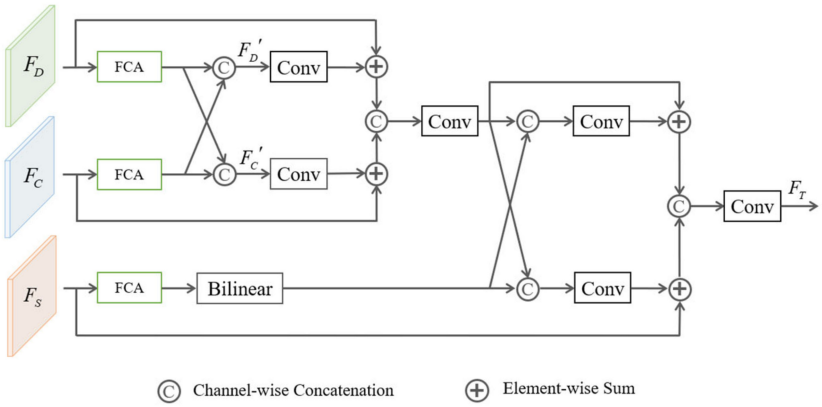


Figure 5. Three-Path Feature Fusion Module.

Firstly, the low-level semantic information output F_D via the detail guidance path and the high-level semantic information output F_C via the context path are fused. Referring to the idea of feature interaction and repeated use, the two features from different paths are refined by the FAM module. After fusion, the refined features pass through a convolution layer and are summed with the original input element by element to obtain $F_D{}'$ and $F_C{}'$. $F_D{}'$ and $F_C{}'$ are concatenated along the channel direction and convolved to obtain the feature fused by the two path features. Next, we fuse the features from the graph convolutional path in a similar scheme to obtain an output F_T of size 1/16. The decoder of the network is composed of four transposed convolution layers. After passing through one transposed convolution layer, the size of the feature map will be doubled. Finally, the output feature of the original size is obtained, and the image is segmented using this feature. The formula of the feature fusion module is expressed as follows:

$$\begin{cases} F_D{}' = conv(concat(FAM(F_D), FAM(F_C))), \\ F_C{}' = conv(concat(FAM(F_D), FAM(F_C))) \end{cases} \tag{9}$$

$$F_{DC} = conv(concat(F_D \oplus F_D{}', F_C \oplus F_C{}')) \tag{10}$$

$$\begin{cases} F_{DC}{}' = conv(concat(F_{DC}, UpSample(FAM(F_S)))), \\ F_S{}' = conv(concat(F_{DC}, UpSample(FAM(F_S)))) \end{cases} \tag{11}$$

$$F_T = conv(concat(F_{DC} \oplus F_{DC}{}', F_S \oplus F_S{}') \tag{12}$$

where F_D, F_C and F_S respectively represent the feature maps output via the CF-path, DIG-path and SIS-path, $FAM(\cdot)$ represents the FAM channel submodule, and $UpSample$ represents bilinear upsampling.

4. Experimental Results

4.1. Dataset

The datasets used in the experiments are Cityscapes and CamVid, both of which have images taken from the perspective of a driving car and are widely used benchmarks for semantic segmentation.

Cityscapes: Cityscapes contains a large number of real, complex urban street scene images, which is a challenging dataset. The dataset contains a total of 25,000 real road scene images, of which 5000 images are finely annotated and 20,000 images are roughly annotated. In our experiments, using only finely annotated data, 5000 high-quality annotated images are further divided into 2975, 500 and 1525 for training, validation and testing, respectively. The fine-annotated semantic subset contains a total of 30 categories. Referring to previous works [13,23], only 19 categories are used for evaluation.

CamVid: CamVid is a high-resolution road scene dataset collected from the first video collection with semantic labels of object categories, with a resolution of 960 × 720. The dataset contains 701 images which are divided into 367, 101 and 233 for training, validation and testing, respectively. The dataset provides ground truth semantic labels for 32 categories, we use 11 categories, and pixels that do not belong to these classes are ignored for a fair comparison with other state-of-the-art methods.

4.2. Implementation Details

The experiment is conducted under the PyTorch framework, and the network is trained and tested under a device with GeForce RTX 2080Ti GPU. Referring to [15,24], the "poly" learning rate is used in the experiments, $lr = lr_i \times \left(1 - \frac{epoch}{\max\_epoch}\right)^{power}$, where the power is set to 0.9 and the initial learning rate is set as 0.01. In the training phase, we choose the stochastic gradient descent (SGD) optimizer, where the momentum is set to 0.9 and the weight decay is set to 1×10^{-5}. Due to the differences between the two datasets, different training parameters are set when using different public datasets to train the model. Specifically, since the image quality of Cityscapes is high and the original image resolution is 1024 × 2048, we set the input resizing to 512 × 1024, and the training epoch is set to 350. For CamVid with relatively few samples and low image resolution, we do not resize the images and set the training epoch to 250.

4.3. Ablation Experiment

To verify the effectiveness of the proposed module, we design the following ablation experiments. Only the CF-path is used as the baseline; modules are added to the baseline model to verify the performance of the proposed algorithm, and the test results of each proposed module on the Cityscapes test dataset are shown in Table 1.

Table 1. Ablation study results of the DGMNet.

Model	Method				mIoU (%)
	FAM	DIG-Path	SIS-Path	TPFF	
model_1 (baseline)					56.3
model_2	✓				61.1
model_3	✓	✓			66.7
model_4	✓		✓		67.9
model_5	✓	✓	✓		73.2
model_6 (DGMNet)	✓	✓	✓	✓	76.9

Baseline: The context path uses MobileNet V3-Small to extract image features and fuse multi-scale features for image segmentation. The mIoU of the baseline is 56.3%.

Baseline + FAM: The experimental results in Table 1 show that the mIoU of the network is improved from 56.3% to 61.1% after using the FAM module. It indicates that our proposed FAM module pays attention to both spatial and channel information and effectively combines multi-dimensional attention to enrich the attention map, so as to refine the features and effectively improve its segmentation performance.

Baseline + FAM + DIG-Path: The mIoU of the network is improved from 61.1% to 66.7% after adding the detailed information supplementary path. The experimental results show that the DFSM module can introduce richer detail information into the deep feature

map, the ability of the neural network to understand and segment object details is greatly improved, thus verifying the effectiveness of this module.

Baseline + FAM + SIS-Path: With the introduction of the DIG-path, the mIoU of the network is improved from 61.1% to 67.9%. It can be seen that the semantic graph path can supplement the high-level semantic information and add a connection between distant features, thereby expanding the receptive field of features, increasing the global semantic information. From another perspective, the understanding of the network on the image is increased, so as to effectively improve the segmentation performance of the network.

Baseline + FAM + DIG-Path + SIS-Path: Experimental results show that using both the DIG-path and SIS-path can further improve the segmentation accuracy of the network, and the mIoU value is 73.2%. We believe that the addition of DIG-path + SIS-path can further enhance the network's learning of some small and thin objects and detail information and still retain enough detail information in the deep network.

Baseline + FAM + DIG-Path + SIS-Path+ TPFF: With the addition of the feature fusion module, the performance of the network is further improved, and the mIoU value reaches 76.9%. The fused feature maps contain semantic information under different angles and make full use of the features of different paths to further improve the segmentation accuracy of the model. In the following, we discuss the proposed attention mechanism.

The TPFF module fuses the feature information of different paths to obtain richer semantic information and supplement the lost detail information to a certain extent. To evaluate the effectiveness of the TPFF module, we perform the following experiments on different fusion schemes, and the experimental results are shown in Table 2. Directly using the traditional Add or Concat methods gives poor segmentation accuracy, with mIoU values of 71.5% and 73.2%, respectively. After adding an attention mechanism on the basis of Add and Concat methods, the segmentation accuracy is improved by 2.8% and 2.4%, respectively. Experimental results show that adding a full attention mechanism can effectively promote the semantic segmentation of images. Compared with the Add + Attention scheme, the proposed fusion module improves the segmentation accuracy by 2.6%. Compared with the Concat + Attention scheme, the segmentation accuracy is improved by 1.3%, which proves the effectiveness of our proposed feature fusion module.

Table 2. The discussion on feature fusion module.

Fusion Method	mIoU (%)
Add	71.5
Add + Attention	74.3
Concat	73.2
Concat + Attention	75.6
TPFF	76.9

In addition, we also visualize the segmentation results of the network after adding different structures. In order to make a clearer comparison of the segmentation effects of the network after adding different structures, we enlarged some details, as shown in Figure 6. It can be seen from the visualization results that the segmentation effect of the baseline is the worst, and the outline of the object cannot be clearly segmented, and the difference between the segmented outline and the real outline is large. By continuously optimizing the network, the segmentation results become more and more accurate, and the segmentation effect of the baseline + FAM + DG path + SIS path + TPFF scheme is the best. The experimental results verify the effectiveness and efficiency of DGMNet again.

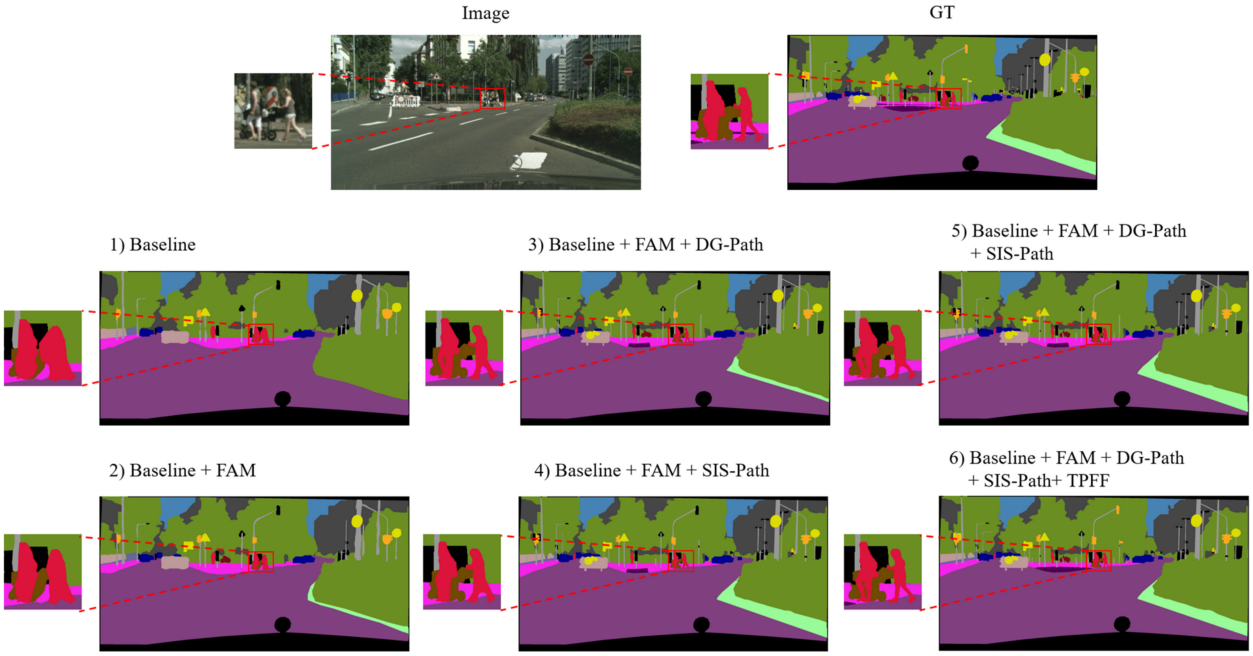


Figure 6. Ablation study results and the ground truth (GT).

4.4. Comparison to State-of-the-Art Methods

4.4.1. Comparative Experiments in Cityscapes

Quantitative Analysis: In order to further verify the effectiveness of our proposed method, we compare the proposed method with real-time and non-real-time semantic segmentation networks, in which the non-real-time segmentation networks are SegNet, FCN, DeepLabv2, SA-FFNet, RefineNet, PSPNet, AttaNet and FLANet. The real-time segmentation networks are ENet, ESPNet, ICNet, ERFNet, AGLNet, DABNet, LEDNet, DSANet, FDDWNet, BiSeNet V2, SwiftNet and LMFFNet.

Experimental results in Table 3 show that our proposed algorithm achieves the highest segmentation accuracy in the lightweight network, with a segmentation accuracy of 76.9%. The segmentation results for individual categories of different models on the Cityscape test set are shown in the table, achieving state-of-the-art results in 9 out of 19 categories. Compared with the second place in the segmentation accuracy of the single class, the segmentation accuracy of the class of some slender objects is significantly improved, in which the mIoU value of the pole is increased by 4.9%, which further proves the effectiveness of the proposed algorithm in improving the accuracy of thin object segmentation. For some small objects, the proposed method also has good improvement effects. Among them, the traffic light is increased by 3.1%, the traffic sign is increased by 3.8%, the person is increased by 1.7%, and the rider is increased by 3.2%. Experiments show that the algorithm is effective for small objects.

For a more comprehensive and fair comparison, we contrast the number of parameters (Params), GFLOP, inference speed (FPS), and mIoU (%) of different networks. The experimental results in Table 4 show that when using the lightweight backbone network (MobileNet V3-Small), the number of parameters of DGMNet is only 2.4 M, the mIoU reaches 76.9%, and the FPS is 141.5. Compared with other real-time segmentation networks, DGMNet has the highest accuracy, and the segmentation speed also reaches a relatively high level. Compared with LMFFNet, we achieved a better improvement in both speed and accuracy performance; compared with STDC-Seg, although the detection speed is slightly reduced, the segmentation accuracy is improved by 3.5%. Although the FPS is reduced, it can still meet the real-time requirements, and compared with the non-real-time network, the accuracy of DGMNet exceeds some networks. In order to make a fair comparison with the non-real-time network and verify the effectiveness of our proposed algorithm, ResNet101 is

used as the backbone network of the network. Compared with the non-real-time network, our proposed network can still achieve the highest segmentation accuracy, and the FPS is 14.9. In conclusion, our proposed network achieves a balance between speed and accuracy and is highly competitive with other segmentation networks.

Table 3. Comparison of IoU values and mIoU for 19 categories of different methods on the Cityscapes test dataset. The highest IoU for each class is highlighted in bold, and the next highest IoU is highlighted in blue.

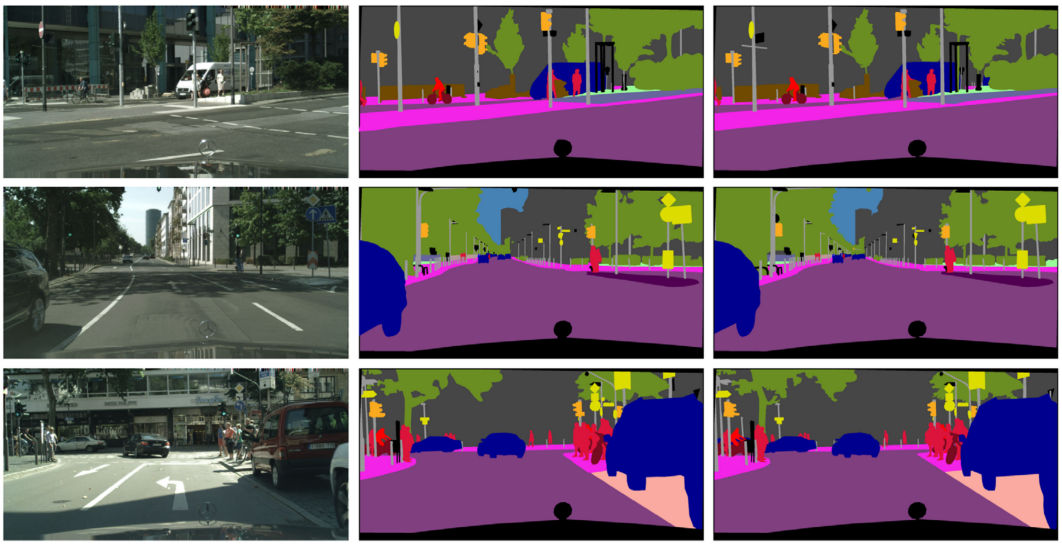
Method	Roa	Sid	Bui	Wal	Fen	Pol	Lig	Sig	Veg	Ter	Sky	Tra	Tru	Bus	Car	Mot	Bic	Per	Rid	mIoU(%)
SegNet [4]	96.4	73.2	84.0	28.4	29.0	35.7	39.8	45.1	87.0	63.8	91.8	44.1	38.1	43.1	89.3	35.8	51.9	62.8	42.8	57.0
FCN [12]	97.4	78.4	89.2	34.9	44.2	47.4	60.1	65.0	91.4	69.3	93.9	46.5	35.3	48.6	92.6	51.6	66.8	77.1	51.4	65.3
DeepLabv2 [15]	97.9	81.3	90.3	48.8	47.4	49.6	57.9	67.3	91.9	69.4	94.2	57.2	56.5	67.5	93.7	57.7	68.8	79.8	59.8	70.4
SA-FFNet [25]	98.6	78.1	91.5	48.9	45.4	54.4	65.0	71.1	92.9	56.0	95.1	72.6	77.0	86.0	92.9	56.8	69.7	78.8	58.8	73.1
RefineNet [26]	97.9	81.3	90.3	48.8	47.4	49.6	57.9	67.3	91.9	69.4	94.2	57.5	56.5	67.5	93.7	57.7	68.8	79.8	59.8	73.6
PSPNet [13]	98.6	86.2	92.9	50.8	58.8	64.0	75.6	79.0	93.4	72.3	95.4	73.8	68.2	79.5	95.9	69.5	77.2	86.5	71.3	78.4
AttaNet [27]	98.7	87.0	93.5	55.9	62.6	70.2	78.4	81.4	93.9	72.8	95.4	78.0	71.2	84.4	96.3	68.6	78.2	87.9	74.7	80.5
FLANet [28]	98.8	87.7	94.3	64.1	64.9	72.4	78.9	82.6	94.2	73.5	96.2	91.6	80.2	93.8	96.6	74.3	79.5	88.7	76.0	83.6
ENet [18]	96.3	74.2	75.0	32.2	33.2	43.4	34.1	44.0	88.6	61.4	90.6	48.1	36.9	50.5	90.6	38.8	55.4	65.5	38.4	58.3
ESPNet [29]	95.6	73.2	86.4	32.7	36.4	46.9	46.7	55.2	89.7	65.9	92	40.6	39.9	47.7	89.8	36.3	54.8	68.3	45.7	60.2
ICNet [5]	98.0	81.9	90.3	46.4	46.5	50.5	57.3	64.2	91.7	68.2	94.4	63.8	53.7	72.5	93.6	48.6	64.6	77.1	57.9	69.5
ERFNet [30]	97.9	82.1	90.7	45.2	50.4	59.0	62.6	68.4	91.9	69.4	94.2	53.7	52.3	60.8	93.4	49.9	64.2	78.5	59.8	69.7
AGLNet [31]	97.8	80.1	91.0	51.3	50.6	58.3	63.0	68.5	92.3	71.3	94.2	42.1	48.4	68.1	93.8	52.4	67.8	80.1	59.6	70.1
DABNet [32]	97.9	82.0	90.6	45.5	50.1	59.3	63.5	67.7	91.8	70.1	92.8	56.0	52.8	63.7	93.7	51.3	66.8	78.1	57.8	70.1
LEDNet [33]	98.1	79.5	91.6	47.7	49.9	62.8	61.3	72.8	92.6	61.2	94.9	52.7	**64.4**	64.0	90.9	44.4	71.6	76.2	53.7	70.6
DSANet [34]	96.8	78.5	91.2	50.5	50.8	59.4	64.0	71.7	92.6	70.0	94.5	50.6	56.1	75.6	92.9	50.6	66.8	81.8	61.9	71.4
FDDWNet [35]	98.0	82.4	91.1	52.5	51.2	59.9	64.4	68.9	92.5	70.3	94.4	48.6	56.5	68.9	94.0	55.7	67.7	80.8	59.8	71.5
BiSeNet v2 [7]	98.2	82.9	91.7	44.5	51.1	63.5	71.2	75.0	92.9	71.1	94.9	56.8	60.5	68.7	94.9	61.5	72.7	83.6	65.4	73.8
LMFFNet [36]	98.3	84.1	92.0	56.1	55.1	62.2	69.0	72.7	93.0	**71.3**	95.0	64.1	60.6	76.7	95.0	60.6	71.7	83.8	66.1	75.1
SwiftNet [37]	**98.3**	83.9	92.2	46.3	52.8	63.2	70.6	75.8	**93.1**	70.3	**95.4**	**71.9**	63.9	**78.0**	**95.3**	**61.6**	**73.6**	84.0	64.5	75.5
DGMNet	98.1	**87.1**	**93.1**	**59.7**	**58.7**	**68.4**	**74.3**	**79.6**	92.8	70.5	94.9	68.8	64.1	77.3	94.8	61.1	73.5	**85.7**	**69.3**	**76.9**

Table 4. Comparison of our method and other state-of-the-art methods on the Cityscapes test dataset.

Method	Pretrain	Backbone	Resolution	Device	Params (M)	GFLOPs	FPS	mIoU
SegNet [4]	ImageNet	VGG16	360 × 640	-	29.5	286.0	14.6	57.0
FCN-8S [12]	ImageNet	VGG16	512 × 1024		-	136.2	2.0	65.3
DeepLabv2 [15]	ImageNet	ResNet101	512 × 1024	Titan X	44	457.8	<1	70.4
SA-FFNet [25]	ImageNet	ResNet101	768 × 768	Titan X	57.5	-	-	73.1
RefineNet [26]	ImageNet	ResNet101	512 × 1024	Titan X	118.1	428.3	9	73.6
PSPNet [13]	ImageNet	ResNet101	713 × 713	Titan X	250.8	412.2	0.78	78.4
FLANet [28]	ImageNet	HRNetV2-W48	768 × 768	-	436	19.37	-	78.9
AttaNet [27]	ImageNet	DF2	512 × 1024	GTX 1080Ti	-	-	71	79.9
DeepLabV3 [17]	ImageNet	ResNet101	769 × 769	Tesla K80	-	-	1.3	81.3
Lawin [38]	ImageNet	Swin-L	1024 × 1024	Tesla V100	-	1797	-	84.4
ViT-Adapter-L [39]	ImageNet	ViT-Adapter-L	896 × 896	Tesla V100	571	-	-	85.2
ENet [18]	No	No	630 × 630	Titan X	0.4	4.4	76.9	58.3
ESPNet [29]	No	ESPNet	512 × 1024	Titan X	0.36	4.0	112	60.2
ERFNet [30]	ImageNet	VGG16	512 × 1024	Titan X	2.1	26.8	49	68.0
ICNet [5]	ImageNet	PSPNet50	1024 × 2048	TitanX	26.5	29.8	30.3	69.5
AGLNet [31]	No	No	512 × 1024	GTX 1080Ti	1.12	13.9	52	70.1
DABNet [32]	No	No	1024 × 2048	GTX 1080Ti	0.76	10.5	27.7	70.1
LEDNet [33]	No	SSNet	512 × 1024	GTX 1080Ti	0.94	11.5	71	70.6
TinyHMSeg [40]	No	No	768×1536	GTX 1080Ti	0.7	3.0	172.4	71.4
DSANet [34]	No	FDSSNet	512 × 1024	GTX 1080Ti	3.47	37.4	34	71.4
FDDWNet [35]	No	No	512 × 1024	GTX 2080Ti	0.8	8.5	60	71.5
BiSeNet v2 [7]	No	No	512 × 1024	GTX 1080Ti	-	11.5	156	72.6
STDC-Seg [8]	No	STDC	512 × 1024	GTX 1080Ti	-	-	188.6	73.4
LMFFNet [36]	No	LMFFNet	512 × 1024	GTX 3090	-	16.7	118.9	75.1
DGMNet	No	MobileNet v3-small	512 × 1024	GTX 2080Ti	2.4	13.4	141.5	76.9
DGMNet_2	No	ResNet 101	512 × 1024	GTX 2080Ti	97.2	264.6	14.9	81.8

Qualitative Analysis: For a qualitative analysis of the proposed network, the visualization results of DGMNet on the Cityscapes test dataset are shown in Figure 7. The experimental results in show that the proposed method has a good segmentation effect for the pole and other thinner objects and produces fine segmentation results near the bound-

ary, which further verifies the effectiveness of our proposed method for the segmentation of thin and small objects.



(a) Image (b) DGMNet (c) GT

Figure 7. The visualization results of DGMNet on the Cityscapes test dataset.

4.4.2. Comparative Experiments in Camvid

To further evaluate the advancement of the proposed method, we also evaluate DGMNet on the CamVid dataset. As shown in Table 5, DGMNet achieves excellent performance. It can process input images with a resolution of 720 × 960 at 121.4 FPS, and the mIoU value on the CamVid test dataset reaches 74.8%. The experimental results show that our proposed method achieves the highest segmentation accuracy under the premise of ensuring the real-time performance of segmentation, which further verifies the effectiveness of the proposed method.

Table 5. Comparisons with other state-of-the-art methods on CamVid.

Method	Resolution	Device	FPS	mIoU (%)
ENet [18]	720 × 960	Titan X	61.2	51.3
ESPNet [29]	720 × 960	Titan X	219.8	55.6
ICNet [5]	720 × 960	Maxwell TitanX	34.5	67.1
ERFNet [30]	720 × 960	Titan X	139.1	67.7
DABNet [32]	360 × 480	GTX 1080Ti	146	66.4
BiSeNetV1 [6]	720 × 960	Titan XP	116.3	68.7
AGLNet [31]	720 × 960	GTX 1080Ti	90.1	69.4
DSANet [34]	360 × 480	GTX 1080Ti	75.3	69.93
CAS [41]	720 × 960	Titan XP	169	71.2
TinyHMSeg [40]	720 × 960	GTX 1080Ti	278.5	71.8
GAS [21]	720 × 960	Titan XP	153.1	72.8
BiSeNet v2 [7]	360×480	GTX 1080Ti	124.5	72.4
STDC2-Seg [8]	720 × 960	GTX 1080Ti	152.2	73.9
DGMNet	720 × 960	GTX 2080Ti	121.4	74.8

5. Conclusions

The detail guided multilateral segmentation network proposed in this paper constructs trilateral parallel paths to fully retain the low-level detail information and high-level semantic information and designs a three-path feature fusion scheme to further optimize the network structure. The ablation experiments of Cityscapes verify the effectiveness of the proposed structure and analyze the segmentation accuracy of a single category.

The analysis results show that the proposed method has superior performance in the segmentation of thin and small objects. Through a comparative analysis of a number of parameters, GFLOPs and FPS, it is proved that DGMNet improves computational efficiency while ensuring segmentation accuracy. The experimental results show that our proposed algorithm achieves a good balance between segmentation speed and accuracy and is competitive with other advanced networks.

Author Contributions: Methodology, Q.J.; software, Q.J.; validation, J.D., T.R. and F.S.; investigation, R.H. and Y.D.; resources, T.R.; writing—original draft preparation, Q.J.; writing—review and editing, J.D. and F.S.; visualization, R.H. and H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (grant number: 61671470) and the China National Key Research and Development Program (grant number: 2016YFC0802904).

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to restrictions privacy.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Han, S.; Mao, H.; Dally, W.J. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding. *arXiv* **2015**, arXiv:1510.00149.
2. Bhattacharya, S.; Lane, N.D. Sparsification and separation of deep learning layers for constrained resource inference on wearables. In Proceedings of the 14th ACM Conf. Embedded Network Sensor System (CD-ROM), Stanford, CA, USA, 14–16 November 2016; pp. 176–189.
3. Liu, Z.; Li, J.; Shen, Z.; Huang, G.; Yan, S.; Zhang, C. Learning Efficient Convolutional Networks through Network Slimming. In Proceedings of the 16th IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2755–2763.
4. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]
5. Zhao, H.; Qi, X.; Shen, X.; Shi, J.; Jia, J. ICNet for Real-Time Semantic Segmentation on High-Resolution Images. In Proceedings of the 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 418–434.
6. Yu, C.; Wang, J.; Peng, C.; Gao, C.; Yu, G.; Sang, N. BiSeNet: Bilateral Segmentation Network for Real-Time Semantic Segmentation. In Proceedings of the 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 334–349.
7. Yu, C.; Gao, C.; Wang, J.; Yu, G.; Shen, C.; Sang, N. BiSeNet V2: Bilateral Network with Guided Aggregation for Real-Time Semantic Segmentation. *Int. J. Comput. Vis.* **2021**, *129*, 3051–3068. [CrossRef]
8. Fan, M.; Lai, S.; Huang, J.; Wei, X.; Chai, Z.; Luo, J.; Wei, X. Rethinking BiSeNet For Real-time Semantic Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Electric Network, Nashville, TN, USA, 19–25 June 2021; pp. 9711–9720.
9. Takikawa, T.; Acuna, D.; Jampani, V.; Fidler, S. Gated-SCNN: Gated Shape CNNs for Semantic Segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 5228–5237.
10. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The Cityscapes Dataset for Semantic Urban Scene Understanding. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 27–30 June 2016; pp. 3213–3223.
11. Brostow, G.J.; Shotton, J.; Fauqueur, J.; Cipolla, R. Segmentation and Recognition Using Structure from Motion Point Clouds. In Proceedings of the 10th European Conference on Computer Vision (ECCV 2008), Marseille, France, 12–18 October 2008; pp. 44–57.
12. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
13. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid Scene Parsing Network. In Proceedings of the 30th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6230–6239.
14. Ronneberger, O. *Invited Talk: U-Net Convolutional Networks for Biomedical Image Segmentation*; Springer: Berlin/Heidelberg, Germany, 2017; p. 3.
15. Liang-Chieh, C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 17600089.

16. Chen, L.-C.; Papandreou, G.; Schroff, F.; Adam, H.J. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv* **2017**, arXiv:1706.05587.
17. Chen, L.-C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In Proceedings of the 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 833–851.
18. Paszke, A.; Chaurasia, A.; Sangpil, K.; Culurciello, E. ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation. *arXiv* **2016**, arXiv:1606.02147.
19. Li, H.; Xiong, P.; Fan, H.; Sun, J.; Soc, I.C. DFANet: Deep Feature Aggregation for Real-Time Semantic Segmentation. In Proceedings of the 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 9514–9523.
20. Li, X.; Zhou, Y.; Pan, Z.; Feng, J.; Soc, I.C. Partial Order Pruning: For Best Speed/Accuracy Trade-off in Neural Architecture Search. In Proceedings of the 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 9137–9145.
21. Lin, P.; Sun, P.; Cheng, G.; Xie, S.; Li, X.; Shi, J. Graph-guided Architecture Search for Real-time Semantic Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Electric Network, Seattle, DC, USA, 14–19 June 2020; pp. 4202–4211.
22. Howard, A.; Sandler, M.; Chu, G.; Chen, L.-C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for MobileNetV3. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 1314–1324.
23. Zhao, H.; Zhang, Y.; Liu, S.; Shi, J.; Loy, C.C.; Lin, D.; Jia, J. PSANet: Point-wise Spatial Attention Network for Scene Parsing. In Proceedings of the 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 270–286.
24. He, J.; Deng, Z.; Zhou, L.; Wang, Y.; Qiao, Y.; Soc, I.C. Adaptive Pyramid Context Network for Semantic Segmentation. In Proceedings of the 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 7511–7520.
25. Zhou, Z.; Zhou, Y.; Wang, D.; Mu, J.; Zhou, H. Self-attention feature fusion network for semantic segmentation. *Neuro-Computing* **2021**, *453*, 50–59. [CrossRef]
26. Lin, G.; Milan, A.; Shen, C.; Reid, I. RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation. In Proceedings of the 30th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 5168–5177.
27. Song, Q.; Mei, K.; Huang, R. AttaNet: Attention-Augmented Network for Fast and Accurate Scene Parsing. In Proceedings of the 11th Symposium on Educational Advances in Artificial Intelligence, Electric Network, Virtual, 2–9 February 2021; pp. 2567–2575.
28. Song, Q.; Li, J.; Li, C.; Guo, H.; Huang, R.J. Fully Attentional Network for Semantic Segmentation. *arXiv* **2021**, arXiv:2112.04108. [CrossRef]
29. Mehta, S.; Rastegari, M.; Caspi, A.; Shapiro, L.; Hajishirzi, H. ESPNet: Efficient Spatial Pyramid of Dilated Convolutions for Semantic Segmentation. In Proceedings of the 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 561–580.
30. Romera, E.; Alvarez, J.M.; Bergasa, L.M.; Arroyo, R. ERFNet: Efficient Residual Factorized ConvNet for Real-Time Semantic Segmentation. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 263–272. [CrossRef]
31. Zhou, Q.; Wang, Y.; Fan, Y.W.; Wu, X.F.; Zhang, S.F.; Kang, B.; Latecki, L.J. AGLNet: Towards real-time semantic segmentation of self-driving images via attention-guided lightweight network. *Appl. Soft Comput.* **2020**, *96*, 106682. [CrossRef]
32. Li, G.; Yun, I.; Kim, J.; Kim, J.J. DABNet: Depth-wise Asymmetric Bottleneck for Real-time Semantic Segmentation. *arXiv* **2019**, arXiv:1907.11357.
33. Wang, Y.; Zhou, Q.; Liu, J.; Xiong, J.; Gao, G.; Wu, X.; Latecki, L.J. LEDNET: A Lightweight Encoder-Decoder Network for Real-Time Semantic Segmentation. In Proceedings of the 26th IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 1860–1864.
34. Elhassan, M.A.M.; Huang, C.; Yang, C.; Munea, T.L. DSANet: Dilated spatial attention for real-time semantic segmentation in urban street scenes. *Expert Syst. Appl.* **2021**, *183*, 115090. [CrossRef]
35. Liu, J.; Zhou, Q.; Qiang, Y.; Kang, B.; Wu, X.; Zheng, B. FDDWNET: A Lightweight Convolutional Neural Network for Real-Time Semantic Segmentation. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Barcelona, Spain, 4–8 May 2020; pp. 2373–2377.
36. Shi, M.; Shen, J.; Yi, Q.; Weng, J.; Huang, Z.; Luo, A.; Zhou, Y. LMFFNet: A Well-Balanced Lightweight Network for Fast and Accurate Semantic Segmentation. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**. [CrossRef] [PubMed]
37. Orsic, M.; Kreso, I.; Bevandic, P.; Segvic, S.; Soc, I.C. In Defense of Pre-trained ImageNet Architectures for Real-time Semantic Segmentation of Road-driving Images. In Proceedings of the 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 12599–12608.
38. Chen, Z.; Duan, Y.; Wang, W.; He, J.; Lu, T.; Dai, J.; Qiao, Y.J. Vision Transformer Adapter for Dense Predictions. *arXiv* **2022**, arXiv:2205.08534.
39. Yan, H.; Zhang, C.; Wu, M.J. Lawin Transformer: Improving Semantic Segmentation Transformer with Multi-Scale Representations via Large Window Attention. *arXiv* **2022**, arXiv:2201.01615.

40. Li, P.; Dong, X.; Yu, X.; Yang, Y. When Humans Meet Machines: Towards Efficient Segmentation Networks. In Proceedings of the 31st British Machine Vision Virtual Conference BMVC, Cardiff, UK, 7–10 September 2020.
41. Zhang, Y.; Qiu, Z.; Liu, J.; Yao, T.; Liu, D.; Mei, T.; Soc, I.C. Customizable Architecture Search for Semantic Segmentation. In Proceedings of the 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 11633–11642.