ISSN: 2710-0588

Multiscale Fusion Convolutional Network in Real-time Semantic Segmentation

Jiali Xing, Yongsheng Dong

School of Information Engineering, Henan University of Science and Technology, Luoyang 471023, China

Abstract

To achieve semantic segmentation tasks in practical applications such as autonomous driving, networks need to efficiently process high-resolution images while maintaining high accuracy. This requires methods to effectively fuse spatial information in high-resolution images with semantic information in low-resolution images. To address this, this paper proposes a Multi-scale Fusion Convolutional Network (MFCNet) based on a single-branch network structure. In order to simultaneously handle information at different scales and assist the network in capturing a wide range of contextual information, separable Multi-Scale Convolution Modules (MSCM) are introduced to enable the network to obtain richer and more comprehensive feature representations. Additionally, considering that shallow-level information is difficult to directly restore resolution, a Dual-Attention Fusion Module (DAFM) is designed, introducing two attention mechanisms to respectively weight feature maps at different resolutions. Experimental results demonstrate that MFCNet achieves outstanding performance in real-time semantic segmentation tasks.

Keywords

Real-time Semantic Segmentation; Multiscale; Dual Attention; Single-branch.

1. Introduction

Semantic segmentation, as a core topic in the field of computer vision, aims to achieve pixel-level classification of images. With the rapid development of deep learning technology, the accuracy and efficiency of semantic segmentation have been significantly improved through the introduction of innovative methods. This progress has been applied in various practical scenarios such as autonomous driving[1], video surveillance[2,3], and robotics[4], inspiring researchers to explore efficient and effective segmentation networks.

To address the challenges of real-time semantic segmentation, the research community has introduced convolutional neural network models that are both efficient and low-latency, aiming to accelerate inference speed while ensuring segmentation accuracy. Models such as DFANet[5] and BiSeNetV1[6] adopt lightweight network architectures and explore feature fusion and aggregation techniques to compensate for potential accuracy loss due to lightweight design. However, most of these lightweight architectures originate from the image classification domain and may not fully meet the specific requirements of image segmentation tasks.

Researchers have adopted different strategies, with a common approach being to reduce the size of input images to improve processing speed. While this method is effective to some extent, it may sacrifice details of image edge regions and small objects, impacting the final segmentation quality. To overcome this challenge, approaches like ICNet[7], HRNet[8], ERFNet[9], BiSeNet[6][10,11] among others, employ multi-path strategies. This structure can integrate low-level detail information with high-level semantic information, aiming for better segmentation results. These methods extract features at different scales in parallel and fuse

ISSN: 2710-0588

them together to obtain more comprehensive and accurate segmentation results. However, adding extra paths to acquire low-level features increases computational complexity, resulting in a decrease in the model's inference speed.

Therefore, in designing real-time semantic segmentation networks, it is necessary to find a balance point that can maintain high segmentation accuracy while achieving low latency and efficient inference. For this purpose, this paper proposes a new network structure called the Multi-scale Fusion Convolutional Network (MFCNet), aimed at reducing the overall computational complexity of the network.

Our main contributions can be outlined as follows:

(1) We construct Multiscale Fusion Convolutional Network architecture called MFCNet, which enables the model to maintain a lightweight design and handle high-resolution images more efficiently.

(2) We introduce a Multiscale Separable Convolution Module (MSCM), which allows for simultaneous processing of information at different scales and helps the network capture a wide range of contextual information, thus enabling the network to acquire richer and more comprehensive feature representations.

(3) We construct a Dual Attention Fusion Module (DAFM), incorporating two attention mechanisms that weight the feature maps of different resolutions. This facilitates better integration of semantic and detailed features.

(4) We validate the effectiveness of our method on the Cityscapes and CamVid datasets. Specifically, our MFCNet method achieves a balanced performance with an mean Intersection over Union (mIoU) of 75.1% at 156.4 Frames Per Second (FPS) on the Cityscapes dataset.

2. Related Work

In recent years, with the advancements in deep learning and computer hardware, semantic segmentation has also made significant progress. In this section, we primarily discuss representative works on real-time performance utilizing multiscale fusion modules and bottleneck blocks.

2.1. Multiscale Fusion Module

The importance and widespread application of multiscale fusion modules in various semantic segmentation models are evident. For instance, the DeepLab series networks[12-14], represent classical semantic segmentation models, employing the Atrous Spatial Pyramid Pooling (ASPP) module, effectively capturing multiscale contextual information and thereby enhancing segmentation performance. Pyramid Scene Parsing Network (PSPNet)[15] utilizes pyramid pooling modules to acquire context information at different scales. This pyramid pooling mechanism enables feature pooling at different scales, capturing semantic information at various scales and achieving good results in segmentation tasks. Dual Attention Network (DANet)[16] employs multiscale attention mechanisms, introducing two attention modules to capture context information at different scales. These attention mechanisms adaptively adjust the weights of each position in the feature map to better integrate multiscale features. These research works all apply multiscale fusion modules in the field of semantic segmentation, effectively improving segmentation accuracy and robustness through the introduction of different mechanisms and architectural designs.

2.2. Bottleneck Block

Bottleneck blocks serve to reduce the computational complexity of networks, decrease parameter count and computational load, facilitating lightweight and efficient model designs. Consequently, many research works incorporate bottleneck blocks. In the classic deep residual

ISSN: 2710-0588

network structure, ResNet (Residual Network)[17], bottleneck blocks are employed to lessen computational complexity by reducing channel numbers, thereby reducing parameter count and computational load while preserving network expressive capability. DenseNet[18], a densely connected network structure, also utilizes bottleneck blocks, which reduce channel numbers through the introduction of 1x1 convolutional layers. Subsequently, in dense connection layers, information propagation and fusion effectively reduce network parameter count and computational load. EfficientNet[19] balances network depth, width, and resolution using compound coefficients, employing 1x1 convolutional layers in each bottleneck block to reduce channel numbers and enhance network efficiency. ShuffleNet[20] introduces bottleneck blocks and channel shuffle operations. In ShuffleNet, bottleneck blocks reduce computational load and parameter count, while channel shuffle operations increase information exchange between feature maps, enhancing network expressive capability. These models hold significant value in scenarios with constrained computational resources, such as computer vision tasks on mobile and embedded devices.

3. Our Proposed Method

In this section, the architecture of the proposed single-branch Multiscale Fusion Convolutional Network (MFCNet) is first introduced. Then, the Multiscale Separable Convolution Modules (MSCM) designed for the network are described, enabling each layer of the network to capture multiscale information. Furthermore, the Dual Attention Fusion Module (DAFM) is introduced, which is used for semantic information fusion.

3.1. Multiscale Fusion Convolutional Network (MFCNet)

In this subsection, the proposed Multiscale Fusion Convolutional Network (MFCNet) is introduced, with the network model depicted in Figure 1. Among many real-time semantic segmentation models, the representative FCN network[21] possesses relatively minimal redundancy. Therefore, the network model in this chapter is also designed based on the single-branch structure of the FCN network. We choose the lightweight MobileNetV2[22] classification network as the backbone network to pursue faster network operation. To further enhance the network's performance in semantic segmentation tasks and reduce computational memory consumption, we have made improvements and optimizations to MobileNetV2. Additionally, we have introduced the Multiscale Separable Convolution Module (MSCM) and the Dual Attention Fusion Module (DAFM).

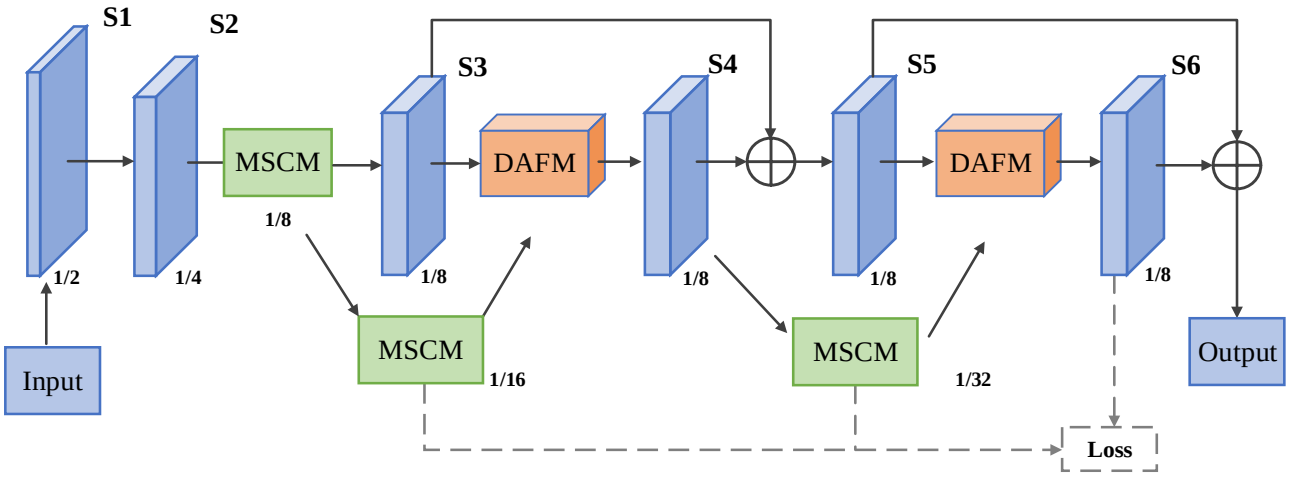

Figure 1. The structural diagram of Multiscale Fusion Convolution Network (MFCNet).

ISSN: 2710-0588

In addition to including standard input and prediction layers, this network architecture comprises six stages to optimize processing flow and enhance segmentation accuracy. These six stages adopt different strategies, ranging from basic spatial resolution downsampling to complex multiscale feature fusion, with each step aimed at improving network efficiency and performance. In the first three stages of the network (S1-S3), we employ a stride of 2 for spatial resolution downsampling. This operation helps reduce the computational load required for subsequent processing stages while laying the foundation for deep-level feature extraction. Specifically, in stages S2 and S3, we replace the depthwise separable bottleneck blocks originally used in MobileNetV2 with 3x3 convolutions. This not only simplifies the network structure but also enlarges the receptive field, enhancing the network's feature extraction capability.

In semantic segmentation tasks, the size of the receptive field is crucial for accurate pixel-level classification. Traditional lightweight networks often fail to provide a sufficient receptive field and require coordination with other modules such as the Pyramid Pooling Module (PSP)[15], Dilated Convolution Module (ASPP)[12], and Deep Aggregation Pyramid Pooling Module (DAPPM)[23]. To address the limited receptive field issue in traditional lightweight networks, we have designed the Multiscale Separable Convolution Module (MSCM). By combining multiscale convolution and separable convolution, MSCM effectively enlarges the receptive field, improving the accuracy of semantic segmentation. Starting from the third Stage (S3), all bottleneck blocks are replaced by MSCM, marking a new level in the network structure that emphasizes multiscale feature fusion and extraction.

In Stage 4 (S4), we introduce the Dual Attention Fusion Module (DAFM) to finely fuse images at 1/8 resolution and 1/16 resolution. DAFM utilizes attention mechanisms to precisely control the fusion process of feature maps, ensuring that key information is retained and highlighted. Subsequently, in Stage 6 (S6), by further fusing images at 1/8 resolution and 1/32 resolution, the network enhances its ability to recognize image details, which is crucial for improving segmentation accuracy.

In the final stage of the network, the prediction layer is composed by combining convolutional layers, global average pooling layers, and fully connected layers. The network architecture proposed in this chapter, through the design of each stage, from basic spatial downsampling to complex feature fusion, each step fully considers the need to improve performance and reduce computational resource consumption. By introducing MSCM and DAFM, the network's receptive field is greatly expanded, enhancing its ability to capture details. Thus, while maintaining lightweight design, significant improvements in semantic segmentation accuracy are achieved.

3.2. Multiscale Separable Convolution Module (MSCM)

In this subsection, we introduce the Multiscale Separable Convolution Module (MSCM). MSCM adopts the multiscale pyramid module commonly used in heavyweight models, complementing the backbone network. The depthwise separable bottleneck blocks in MobileNetV2 perform feature extraction by employing depthwise and pointwise convolutions, followed by downsampling and information preservation through residual connections. Finally, linear bottleneck operations are applied to enhance feature representation and the network's nonlinearity.

The detailed structure of the Multiscale Separable Convolution Module (MSCM) is shown in Figure 2. We hierarchically divide the feature map into four parts through point convolution, with the first part comprising five-eighths of the total channels, and the remaining three parts each comprising one-eighth, undergoing different degrees of pooling operations and utilizing pooling with strides of 2, 4, and 8 to obtain feature representations at different scales. Pooling operations reduce the dimensionality of the feature map to lower computational costs while

ISSN: 2710-0588

retaining core feature information. This approach enables the model to comprehend images at different scales, enhancing its generalization ability. The feature maps after pooling are concatenated and restored through point convolution. By employing pooling at different scales, the module can capture diverse features, enabling the model to better handle image content of various sizes and detail levels. It is noteworthy that using pooling operations to obtain multiscale information incurs lower computational costs compared to changing the size of convolution kernels. Methods such as Dilated Convolution ASPP[12], Mixed Convolution[24], and Crosformer[25] capture multiscale information by directly adjusting the size of convolution kernels or using kernels of different sizes, which provide richer feature representations but significantly increase computational overhead. Whether through expanding convolution kernels or increasing their size, the essence is to extract multiscale features from images. However, the former strategy utilizing pooling operations is more practical due to its lower computational costs. The original information of the feature map is preserved through skip connections.

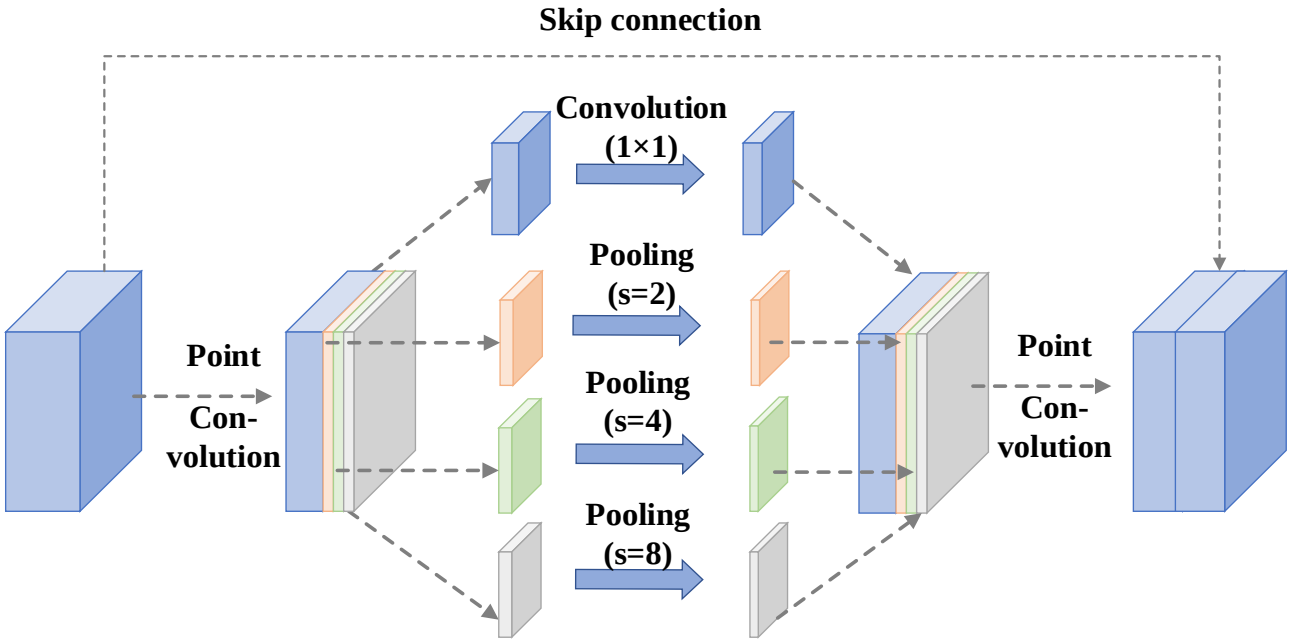

Figure 2. The structural diagram of Multiscale Separable Convolution Module (MSCM).

In addition to compensating for the deficiencies of the backbone network, MSCM also has the effect of increasing the receptive field. In the past, the method for increasing the receptive field was to add receptive field modules after the backbone network. However, the advantage of multiscale convolution lies in integrating modularization into the network, ensuring sufficient receptive field is obtained from the backbone network. Obtaining the receptive field in advance also benefits the backbone network in better feature extraction without adding new contextual modules. In summary, the network introduces the multiscale pyramid module in MobileNetV2, compensating for the deficiencies of the backbone network through depthwise separable bottleneck blocks and multiscale operations, and enhancing the receptive field effect. This integrated modular design approach not only improves model performance in lightweight networks but also controls computational costs.

ISSN: 2710-0588

3.3. Dual Attention Fusion Module (DAFM)

In semantic segmentation network architectures, effectively integrating spatial information and semantic information from high-resolution images is crucial. The detailed structure of the Dual Attention Fusion Module (DAFM) Figure 3.

The DAFM first concatenates input information from two branches, namely the high-resolution and low-resolution branches. The feature maps from the low-resolution detail branch are adjusted to the same size as the semantic branch from the high-resolution using rapid downsampling techniques. Subsequently, the feature maps undergo a "ConvBNReLU" operation and are split into two streams. One stream performs channel-wise fusion by adding the feature maps pixel-wise , followed by processing the fused feature map with a Sigmoid activation function to generate a feature mask. This feature mask contains essential information from the fused feature map and can guide further fusion processes. By applying attention weighting to the feature maps, the model can more accurately focus on important feature regions. These weighted feature maps are then fused with the feature maps from the branches through matrix multiplication.

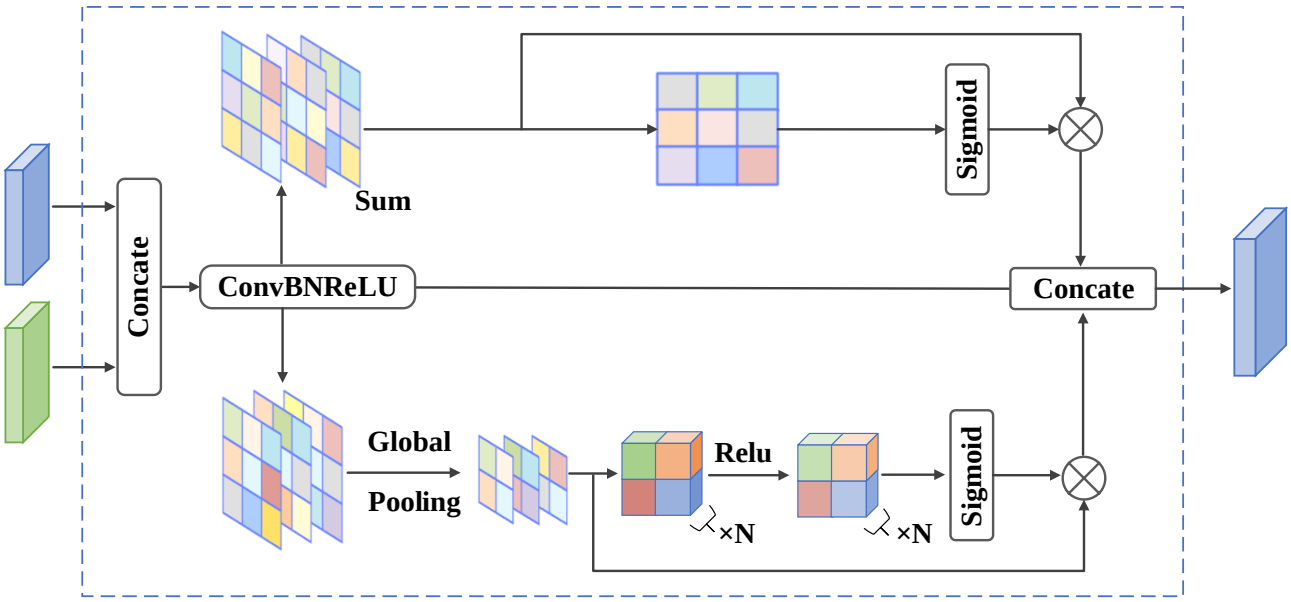

Figure 3. The structural diagram of Dual Attention Fusion Module (DAFM).

The other branch first obtains a feature vector through global pooling, allowing the calculation of attention weights for each position on the feature map, thereby guiding the model to focus more on important regions. Subsequently, features from each branch are extracted through convolution operations, enhancing the model's perception of image details and semantics. Following this, the features undergo activation functions (ReLU and Sigmoid) to generate feature masks rich in information. These masks contain information from both branches, enabling the model to more accurately focus on important feature regions while suppressing irrelevant or interfering information, thus achieving more effective information fusion. Finally, the two branches, processed through attention mechanisms, are concatenated with the original input information to achieve deep-level information integration.

4. Experiments

In this section, we evaluate our proposed MFCNet on two datasets, Cityscapes and CamVid, and compare it against representative methods. Firstly, we introduce the datasets and implementation details. Then, we assess the effectiveness of our proposed MFCNet through

ISSN: 2710-0588

ablation experiments and analyze the performance of each module on the Cityscapes validation set. Next, we report our final accuracy and speed (FPS) results on various benchmark tests by comparing with other algorithms.

4.1. Datasets and Evaluation Metrics

Cityscapes: The Cityscapes dataset[26] focuses on semantic understanding of urban street scenes, typically captured from the perspective of vehicles, and is commonly employed in segmentation tasks. This dataset comprises 5,000 images with high-quality dense pixel annotations, divided into training, validation, and test sets. For our experiments, we utilize 2,975, 500, and 1,525 images respectively to evaluate the effectiveness of the methods. The annotated images in Cityscapes encompass 30 classes, of which 19 are utilized for semantic segmentation tasks. These images boast a high resolution of up to 2048×1024 pixels. The high-resolution nature of the images in this dataset poses a significant challenge for real-time semantic segmentation methods.

Camvid: The Cambridge-driving Labeled Video Database (CamVid)[27] is a dataset dedicated to road scene segmentation, similar to urban landscapes. CamVid stands out for its smaller scale and resolution. It consists of 701 frames with detailed annotations, of which 367 are allocated for training, 101 for validation, and 233 for testing. All images maintain a uniform resolution of 960×720 pixels. We have selected 11 classes from these images to enhance our training dataset. We combine the training and validation sets for training purposes and evaluate our proposed method on the test set.

Evaluation metrics: On all the datasets we utilize, we adopt the mean Intersection over Union (mIoU) and Frames Per Second (FPS) as the standard metrics for evaluation.

4.2. Implementation Details

Training Strategy: The MFCNet method proposed is implemented based on the PaddlePaddle framework[28]. In the experiments, we continue to use the Adam optimizer combined with polynomial decay for training the model. Regarding data augmentation, adopted strategies include random scaling, random padding crops, random horizontal flipping, random brightness, and normalization. Specifically, the scale variation range for random scaling operations is [0.5, 2]. For the Cityscapes dataset, we employ a two-stage training strategy. In the first stage, training is conducted once on the original images with a resolution of 1024×2048. The initial learning rate is set to 0.002, and the batch size is 6. Considering the potential negative impact of low batch size in a single-card environment, we perform two rounds of training to improve the results. In the second stage, the results from the first round are used as pre-training parameters, and training is conducted on half-size images with a resolution of 512×1024. The batch size is set to 24, and the learning rate is adjusted to 0.003.For the CamVid dataset, the image input resolution is set to 720×960. The initial learning rate is 0.003, and the batch size is set to 6. Training is conducted for 80,000 iterations (approximately 1026 rounds). These are the specific details of the training strategy adopted.

Inference settings: The inference speed is measured using Tesla V100, and the testing code is provided by PaddleSeg. The environment includes CUDA 10.1 and Paddle 2.2.2. During the inference process of the model, batch normalization functionality is integrated into the convolutional layers, thereby excluding the computation of batch normalization layers to simplify calculations. For a network with an input resolution of 2048×1024, 1000 iterations are run on the Cityscapes dataset; likewise, for a network with an input resolution of 960×720, 1000 iterations are run on the CamVid dataset. Random factors are eliminated, and the average time is reported. This setup enables accurate measurement of inference speed and provides a reliable assessment of model performance.

ISSN: 2710-0588

4.3. Experiments on Cityscapes

4.3.1. Ablation Study

We use the MFCNet model as the baseline model and conduct ablation experiments on its different modules, including the Multiscale Separable Convolution Module (MSCM) and the Dual Attention Fusion Module (DAFM).

Table 1. Comparison results of different ablation conditions

Model	MSCM	DAFM	FPS	FLOPS	Params	mIoU	MPA
Baseline			180.2	12.4	1.1	70.5	95.0
MFCNet	√		176.7	12.8	1.1	71.7	95.2
MFCNet		√	170.7	13.1	1.2	73.8	95.3
MFCNet	√	√	156.4	14.7	1.5	74.0	95.6
MFCNet *	√	√	156.4	14.7	1.5	75.1	95.7

In the experiments, the model adopts a single-round training strategy and undergoes multiple sets of experiments on the Cityscapes dataset. To ensure comparability, the experimental settings adhere to fixed conditions to obtain more accurate conclusions. The experimental results are shown in Table 1 ("*" denotes methods with pre-trained weights). We compares and analyzes the effects of different modules and their computational resource consumption. Based on the experimental results in Table 1, we can draw the following conclusions:

MSCM: In our experiments, we compared the MSCM with the baseline model, intuitively demonstrating the segmentation performance advantages of the MSCM The experimental results show that after adding the MSCM, the model's computational and parameter quantities are approximately equal to those of the baseline model. Although there is a slight decrease in processing speed, there is a significant improvement in segmentation performance, demonstrating that this module can enhance network performance without sacrificing computational efficiency. The improvement in mean Intersection over Union (mIoU) indicates that the module can more accurately perform image segmentation, capturing more detailed information.

Table 2. MFCNet Network Compared to Other Methods on the Cityscapes Validation Set

Model	GPU	Resolution	GFLOPs	Params	FPS	mIou
ESPNet V2 [29]	-	512×1024	-	-	-	66.4
Fast-SCNN [30]	TitanXp	1024×2048	-	1.1M	123.5	68.6
LiteSeg [31]	GTX1080Ti	360×640	4.9	4.38M	161	67.8
ERFNet [9]	TitanX M	512×1024	27.7	20M	41.7	70.0
GAS [32]	TitanXp	769×1537	-	-	108.4	72.4
RFNet [33]	RTX2080Ti	1024×2048	-	23.69M	22.2	72.5
FasterSeg [34]	GTX1080Ti	1024×2048	28.2	4.4M	163.9	73.1
BiSeNetV1 [6]	GTX1080Ti	768×1536	14.8	5.8M	105.8	69.0
BiSeNetV1 [6]	GTX1080Ti	768×1536	55.3	49M	65.8	74.8
BiSeNetV2Ł [10]	GTX1080Ti	512×1024	21.1	-	156	73.4
BiSeNetV2-LŁ [10]	GTX1080Ti	512×1024	118.5	-	47.3	75.8
MFCNet	RTX3090	1024×2048	14.7	1.5M	156.4	74.0
MFCNet*	RTX3090	1024×2048	14.7	1.5M	156.4	75.1

ISSN: 2710-0588

DAFM: We conducted comparative experiments between the Dual Attention Fusion Module and the baseline model. As shown in the experimental results in Table 1, when using the DAFM, the computational and parameter quantities of MFCNet increase compared to the baseline model, leading to a slower inference speed. However, the increase in parameters is primarily utilized for deeper feature fusion in the network. Overall, considering both the inference speed and performance, the overall segmentation performance did not suffer significantly, with a greater improvement in performance. DAFM has shown improvement in performance, providing the model with enhanced learning capabilities for feature fusion.

Two-Stage Training Strategy: To address the issue of training with images of different resolutions, we propose a two-stage training strategy. Firstly, training with lower resolution images of 512×1024 facilitates quick convergence and acquisition of preliminary model parameters. This approach effectively reduces training time and makes it easier to achieve optimal performance in a single GPU environment. Secondly, in the second stage of training, original resolution images of size 1024×2048 are used for fine-tuning based on the results obtained from the first stage of training. As training with lower resolution images leads to rapid convergence and training with higher resolution images may be affected by batch normalization layer performance, this two-stage training strategy makes full use of the rapid convergence of lower resolution images while avoiding issues with batch normalization, thereby improving network performance and effectiveness. Through the two-stage training strategy, the network effectively reduces training time while ensuring training effectiveness. The first stage of training uses lower resolution images for rapid convergence and obtaining initial model parameters. The second stage of training involves fine-tuning with original resolution images, further enhancing model performance. The benefit of this training strategy lies in its ability to leverage the rapid convergence of lower resolution images while avoiding model generalization issues caused by batch normalization, thereby improving network performance and effectiveness.

4.3.2. Comparing with State-of-the-art Techniques

Table 3. MFCNet network compared to other methods on the Camvid testing set

Model	Backbone	GPU	FPS	mIou
Enet [35]	no	TitanX	61.2	51.3
ICNet [7]	PSPNet50	TitanX M	27.8	67.1
DFANet A [5]	XceptionA	TitanX M	120	64.7
DFANet B [5]	XceptionB	TitanX M	160	59.3
AGLNet [36]	no	GTX1080Ti	90.1	69.4
BiSeNetV1 [6]	Xception39	GTX1080Ti	175	65.6
BiSeNetV1 [6]	ResNet18	GTX1080Ti	116.3	68.7
BiSeNetV2Ł [10]	XceptionA	TitanX M	124.5	72.4
BiSeNetV2-LŁ [10]	no	GTX1080Ti	32.7	73.2
SwiftNet [37]	ResNet18	GTX1080Ti	-	72.6
STDC1-SegŁ [11]	STDC1	GTX1080Ti	197.6	73.0
STDC2-SegŁ [11]	STDC2	GTX1080Ti	152.2	73.9
S^2-FPN18 [38]	ResNet18	GTX1080Ti	124.2	69.5
S^2-FPN34 [38]	ResNet34	GTX1080Ti	107.2	71.0
S^2-FPN34M [38]	ResNet34	GTX1080Ti	55.5	74.2
MFCNet	MobileNetV2	RTX3090	151.9	74.0
MFCNet *	MobileNetV2	RTX3090	151.9	75.2

ISSN: 2710-0588

In this section, we compare our proposed model with other state-of-the-art methods on the Cityscapes. For the Cityscapes dataset, we present the segmentation accuracy of the model on the validation set, as well as the inference speed on the test set. The optimal segmentation accuracy is obtained through training on the validation set, and the evaluation of segmentation accuracy is conducted on an NVIDIA 3090 GPU. The measurement of inference speed is carried out on a Tesla V100 GPU.

In Table 2, we present the comparison results between our proposed method and representative approaches, providing detailed metrics such as model names, GPU, resolution, GFLOPs, parameters, speed, and segmentation accuracy. Our method achieves a balanced performance with an mIoU of 75.1% at 156.4 FPS, demonstrating good trade-offs. Additionally, our network model exhibits superiority in terms of GFLOPs and parameters. ("*" indicates that the experiment incorporates the model results trained on the Cityscapes dataset as pretraining parameters.) It is worth noting that while some advanced models may employ specific techniques such as multiscale testing and sliding window methods to enhance segmentation accuracy (It is denoted by "Ⱡ"), we did not adopt such methods due to considerations of inference speed, as they often result in higher computational time.

4.3.3. Experiments on CamVid

For the CamVid dataset, we display the segmentation accuracy and inference speed of our model on the test set. The experiment follows conventional methods, training on a combination of the training and validation sets, and then evaluating on the test set. Inference time measurements were conducted on a Tesla V100 GPU, with input image resolution of 720×960. Table 3 compares our method with representative approaches. Our proposed method achieves an average IoU of 75.2 at 151.9 FPS, with "*" indicating the experiment incorporates training results on the Cityscapes dataset as pretraining weights and "Ⱡ" indicating the method utilizes TensorRT acceleration. Compared to most methods, our model demonstrates higher segmentation accuracy and faster speed.

4.3.4. Visualization Experiments on Cityscapes

In this section, visual segmentation results on the Cityscapes dataset are presented, comparing the outputs of BiseNetV1[6], BiseNetV2[10], STDCNet[11], and the ground truth labels arranged from top to bottom.

As shown in Figure 4, the visual comparison results demonstrate the effectiveness of the proposed multiscale fusion convolution method in semantic segmentation tasks. By observing the images, it is evident that this method has achieved significant improvements in segmenting small-scale objects, resulting in more complete and clear object contours. This improvement facilitates accurate segmentation of detailed objects such as small traffic signs and pedestrian crossings in the images. However, there are still some segmentation issues, particularly in the junctions between foreground and background, where the misclassification rate remains relatively high. Some objects or their local regions are incorrectly classified as other categories; For example, a luggage carried by a pedestrian might be misclassified as the same category as the pedestrian. These misclassifications could have practical implications, such as leading to misjudgments in automated driving systems. Nonetheless, overall, the method demonstrates relative effectiveness in semantic segmentation tasks and achieves a good balance. It exhibits noticeable improvements in segmenting small-scale objects, but further refinement and optimization are needed to address the misclassification issues in the junctions between foreground and background. These results provide guidance for further refinement and optimization of methods to enhance the accuracy and completeness of semantic segmentation.

ISSN: 2710-0588

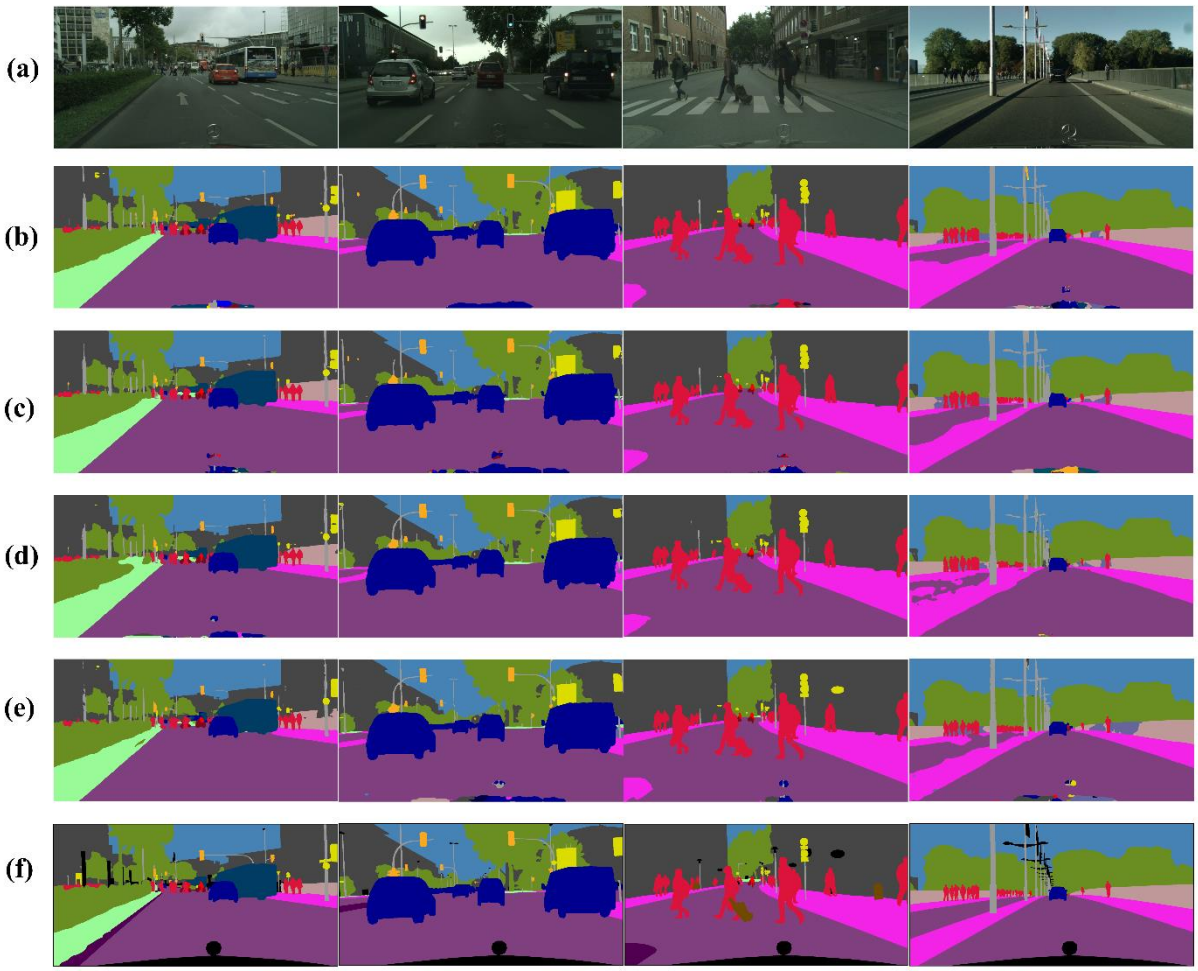

Figure 4. Visualization of segmentation results on the Cityscapes validation set. (a) Image; (b) MFCNet(ours); (c) STDCNet; (d) BiseNetV2; (e) BiseNetV1;(f) GroundTruth

5. Conlusion

In this paper, we propose a method called the Multiscale Fusion Convolution Network (MFCNet). This method significantly improves the problem of multiscale information loss and insufficient receptive fields in the network by introducing Multiscale Separable Convolution Module (MSCM). This improvement enhances the model's fitting ability by incorporating multiscale information without adding excessive computational overhead. Additionally, the network is designed with a Dual Attention Fusion Module (DAFM), which integrates two attention mechanisms to weight the feature maps of different resolutions. This aids the network in focusing better on task-relevant features while suppressing task-irrelevant information, thus fusing the spatial and semantic information required for semantic segmentation. This fusion enables the model to better understand the semantic structure of the image and achieve more accurate segmentation. Through extensive experimental evaluation, it is observed that under the same computational conditions and inference time, our design achieves satisfactory segmentation performance.

References

[1] Tsai J, Chang C C, Li T. Autonomous driving control based on the technique of semantic segmentation [J]. Sensors, 2023, 23(2): 895.

ISSN: 2710-0588

[2] Zhuang J, Wang Z, Wang B. Video semantic segmentation with distortion-aware feature correction [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2020, 31(8): 3128-3139.

[3] Tan Z, Liu B, Chu Q, et al. Real time video object segmentation in compressed domain [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2020, 31(1): 175-188.

[4] Li X, Su J, Yue Z, et al. Adaptive multi-ROI agricultural robot navigation line extraction based on image semantic segmentation [J]. Sensors, 2022, 22(20): 7707.

[5] Li H, Xiong P, Fan H, et al. DFANet: Deep feature aggregation for real-time semantic segmentation [C]. IEEE Conference on Computer Vision and Pattern Recognition, 2019: 9522-9531.

[6] Yu C, Wang J, Peng C, et al. BiseNet: Bilateral segmentation network for real-time semantic segmentation [C]. European Conference on Computer Vision, 2018: 325-341.

[7] Zhao H, Qi X, Shen X, et al. ICNet for real-time semantic segmentation on high-resolution images [C]. European Conference on Computer Vision, 2018: 405-420.

[8] Sun K, Xiao B, Liu D, et al. Deep high-resolution representation learning for human pose estimation [C] IEEE Conference on Computer Vision and Pattern Recognition. 2019: 5693-5703.

[9] Romera E, Alvarez J M, Bergasa L M, et al. ERFNet: Efficient residual factorized convnet for real-time semantic segmentation [J]. IEEE Transactions on Intelligent Transportation Systems, 2017, 19(1): 263-272.

[10] Yu C, Gao C, Wang J, et al. BiseNet V2: Bilateral network with guided aggregation for real-time semantic segmentation [J]. International Journal of Computer Vision, 2021, 129(11): 3051-3068.

[11] Fan M, Lai S, Huang J, et al. Rethinking BiSeNet for real-time semantic segmentation [C]. IEEE Conference on Computer Vision and Pattern Recognition, 2021: 9716-9725.

[12] Chen L C, Papandreou G, Kokkinos I, et al. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs [J]. IEEE Transactions on Transactions on Pattern Analysis and Machine Intelligence, 2017, 40(4): 834-848.

[13] Chen L C, Papandreou G, Schroff F, et al. Rethinking atrous convolution for semantic image segmentation [J]. arXiv preprint arXiv: 1706.05587, 2017.

[14] Chen L C, Zhu Y, Papandreou G, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation [C]. European Conference on Computer Vision, 2018: 801-818.

[15] Zhao H, Shi J, Qi X, et al. Pyramid scene parsing network [C]. IEEE Conference on Computer Vision and Pattern Recognition, 2017: 2881-2890.

[16] Fu J, Liu J, Tian H, et al. Dual attention network for scene segmentation [C]. IEEE Conference on Computer Vision and Pattern Recognition, 2019: 3146-3154.

[17] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition [C]. IEEE Conference on Computer Vision and Pattern Recognition, 2016: 770-778.

[18] Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks [C] IEEE Conference on Computer Vision and Pattern Recognition. 2017: 4700-4708.

[19] Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks [C] International Conference on Machine Learning. 2019: 6105-6114.

[20] Zhang X, Zhou X, Lin M, et al. Shufflenet: An extremely efficient convolutional neural network for mobile devices [C] IEEE Conference on Computer Vision and Pattern Recognition. 2018: 6848-6856.

[21] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation [C]. IEEE Conference on Computer Vision and Pattern Recognition, 2015: 3431-3440.

[22] Sandler M, Howard A, Zhu M, et al. MobileNetV2: Inverted residuals and linear bottlenecks [C]. IEEE Conference on Computer Vision and Pattern Recognition, 2018: 4510-4520.

[23] Hong Y, Pan H, Sun W, et al. Deep dual-resolution networks for real-time and accurate semantic segmentation of road scenes [J]. arXiv preprint arXiv: 2101.06085, 2021.

[24] Tan M, Le Q V. MixConv: Mixed depthwise convolutional kernels [J]. arXiv preprint arXiv: 1907.09595, 2019.

[25] Wang W, Chen W, Qiu Q, et al. Crossformer++: A versatile vision transformer hinging on cross-scale attention [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023.

ISSN: 2710-0588

[26] Cordts M, Omran M, Ramos S, et al. The cityscapes dataset for semantic urban scene understanding [C]. IEEE Conference on Computer Vision and Pattern Recognition, 2016: 3213-3223.

[27] Brostow G J, Shotton J, Fauqueur J, et al. Segmentation and recognition using structure from motion point clouds [C]. European Conference on Computer Vision, 2008: 44-57.

[28] Ma Y, Yu D, Wu T, et al. PaddlePaddle: An open-source deep learning platform from industrial practice [J]. Frontiers of Data and Domputing, 2019, 1(1): 105-115.

[29] Mehta S, Rastegari M, Shapiro L, et al. ESPNetv2: A light-weight, power efficient, and general purpose convolutional neural network [C]. IEEE Conference on Computer Vision and Pattern Recognition. 2019: 9190-9200.

[30] Poudel R P K, Liwicki S, Cipolla R. Fast-SCNN: Fast semantic segmentation network [J]. arXiv preprint arXiv: 1902.04502, 2019.

[31] Emara T, Abd El Munim H E, Abbas H M. Liteseg: A novel lightweight convnet for semantic segmentation [C]. Digital Image Computing Techniques and Applications, 2019: 1-7.

[32] Lin P, Sun P, Cheng G, et al. Graph-guided architecture search for real-time semantic segmentation [C]. IEEE Conference on Computer Vision and Pattern Recognition, 2020: 4203-4212.

[33] Sun L, Yang K, Hu X, et al. Real-time fusion network for RGB-D semantic segmentation incorporating unexpected obstacle detection for road-driving images [J]. IEEE Robotics and Automation Letters, 2020, 5(4): 5558-5565.

[34] Chen W, Gong X, Liu X, et al. Fasterseg: Searching for faster real-time semantic segmentation [J]. arXiv preprint arXiv: 1912.10917, 2019.

[35] Paszke A, Chaurasia A, Kim S, et al. ENet: A deep neural network architecture for real-time semantic segmentation [J]. arXiv preprint arXiv: 1606.02147, 2016.

[36] Kumaar S, Lyu Y, Nex F, et al. CABiNet: Efficient context aggregation network for low-latency semantic segmentation [C]. IEEE International Conference on Robotics and Automation, 2021: 13517-13524.

[37] Orsic M, Kreso I, Bevandic P, et al. In defense of pre-trained imagenet architectures for real-time semantic segmentation of road-driving images [C]. IEEE Conference on Computer Vision and Pattern Recognition, 2019: 12607-12616.

[38] Elhassan M A M, Yang C, Huang C, et al. S^2-FPN: Scale-ware strip attention guided feature pyramid network for real-time semantic segmentation [J]. arXiv preprint arXiv: 2206.07298, 2022.