Multilevel Feature Guided Real-time Semantic Segmentation

Luhang Shen

Abstract— **Aiming at the problems of high complexity, low training accuracy and large number of parameters, a real-time semantic segmentation algorithm based on convolutional neural network and multiple attention mechanisms is proposed. In this model, a two-branch real-time semantic segmentation algorithm BiseNetV2 is used as the benchmark model, in which the semantic branch is responsible for feature extraction and the detail branch is responsible for preserving spatial information. The model integrates large convolution kernel attention mechanism and parallel attention mechanism. Semantic branch fusion large convolution kernel attention mechanism can effectively extract features. The two-branch fusion adopts cross-attention mechanism, which effectively improves segmentation accuracy and reduces computation cost. Experiments on CityScapes, a public dataset, show that the algorithm achieves an average crossover ratio of 73.6%, which is a significant improvement over the benchmark model and reduces the number of references by 15%. Compared with the latest algorithms in the field, the proposed algorithm shows remarkable advantages in inference precision, parameter number and model complexity**

Index Terms— **Real-time semantic segmentation, Attention mechanism, Two-branch network**

I. INTRODUCTION

With the rapid development of autonomous[1] driving technology, it is required to be able to effectively identify key elements such as road surfaces, vehicles, pedestrians, traffic signs, obstacles, etc., to provide vehicles with a comprehensive understanding of the environment. This deep perception enables autonomous driving systems to navigate safely and reliably in complex urban traffic environments. Therefore, real-time semantic segmentation has become the key technology in automatic driving system. In order to achieve safe and efficient autonomous driving scenarios, accurate environment perception is essential, but the existing real-time semantic segmentation tasks have the problems of too many parameters, low segmentation accuracy, and uneven speed and accuracy.

In recent years, with the continuous progress of deep learning technology, many advanced semantic segmentation networks have been proposed, such as full Convolutional network (FCN), U-Net, DeepLab, etc. These networks have achieved significant improvements in accuracy and speed, driving advances in autonomous driving technology. However, despite many research results, how to maintain efficient semantic segmentation in dynamic and complex environments remains a challenge to be solved. There are many interfering factors in the automatic driving scene, such as the change of illumination, weather conditions, the dynamic behavior of pedestrians and other vehicles, etc., which may affect the accuracy of semantic segmentation.

In addition, most of the current semantic segmentation models rely on a large number of labeled data for training, and in practical applications, the cost of data acquisition and labeling is high, which limits the popularization and application of the model. Therefore, how to improve the generalization ability of the model and reduce the dependence on labeled data is also an important direction of research.

This paper will explore the application[2] of semantic segmentation networks in the field of autonomous driving, analyze the advantages and disadvantages of the current technology, and propose the direction of future research. We hope to contribute to the development of intelligent transportation system by studying the importance of semantic segmentation in autonomous[3] driving. Through continuous technological innovation and practical exploration, autonomous driving will play a more important role in the future transportation field and change the way we travel.

Based on BiseNetV2, a real-time semantic segmentation[4] algorithm based on guided aggregation proposed by Yu et al., this paper proposes a semantic segmentation algorithm that integrates convolutional neural networks and attention mechanisms. The parallel attention mechanism and the cross-self-attention mechanism are integrated respectively to retain more semantic information in the semantic branches through the parallel attention mechanism, reduce the information loss caused by fast downsampling and narrow channels, and improve the segmentation accuracy. By using depth separable convolutional design detail branches, it can effectively reduce the number of model parameters, improve computational efficiency, while maintaining good segmentation performance. deep separable convolution By decommissioning a standard convolution operation into two steps, this decomposition not only reduces the computational effort, also allows the model to achieve better segmentation results with limited resources. overcomes the limitation of traditional convolution receptive field and has the advantages of fewer parameters, lower calculation cost, higher inference speed and accuracy.

The main contributions of this paper can be summarized as follows:

(1) A semantic segmentation model named CANet is proposed, which mainly focuses on the fusion process of feature extraction and fusion branches in semantic branches, so as to effectively improve the accuracy of semantic segmentation.

(2) For the problem that the number of parameters is large and the features cannot be extracted effectively, deep separable convolution is used instead of ordinary convolution in the detail branch to enrich the feature information and effectively improve the segmentation effect.

Manuscript received April 16, 2025

Luhang Shen, School of computer science and technology, Tiangong University, Tianjin, China

(3) Lsk module is added after each stage in the semantic branch to make full use of features of different convolution nuclei and different scales for fusion, so that the features obtained are more global.
(4) After passing through the detail branch and the semantic branch, we use the cross-attention mechanism as the fusion layer. More important channels will be assigned important weight values, so in network training, important feature channels will get more attention, so as to obtain better training results.

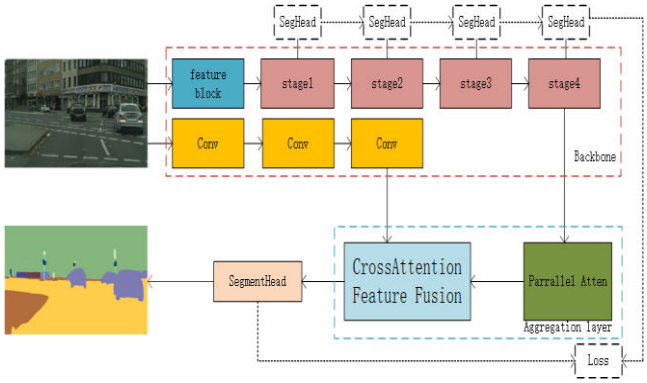


Fig. 1. System Structure Diagram

II. Proposed Method

A. Overall architecture

The structure of the model proposed in this paper is a two-branch network[5], and a segmentation head is placed at the end of the network to output the segmentation results. The overall structure of the model is shown in the figure above. The network has three main components: the two-branch backbone network in the red dashed line frame, the two-branch fusion layer in the blue dashed line frame, and the accelerated training section at the top. The two-channel trunk contains a detail branch (yellow rectangular block) and a semantic branch (pink rectangular block). Meanwhile, the numbers in the yellow rectangle show the ratio of the feature mapping to the input resolution. In addition, in the accelerated training component, we use some auxiliary segmentation heads to improve segmentation performance without additional inference costs.
In the backbone phase of the network, two branches are carried out simultaneously. In the upper part of the figure is the semantic branch, which has a narrow channel size, and narrow channels can also facilitate the model to capture more semantic information. What high-level semantics need to acquire are context dependencies and semantic features. Therefore, CANet adopts a fast downsampling strategy to rapidly improve the feature representation level and expand the receptive field. CANet uses an attention mechanism to capture global information. Semantic branching is lightweight due to fast downsampling strategies and narrow channel dimensions.

The feature block in the semantic branch is composed of two 1×1 convolution operations to preserve the features of the input image.
The stem block module is composed of convolution, batch normalization, activation function and pooling operation. A convolution operation with convolution kernel of 3 and step size of 2 is performed on the input images first, and then the convolution results are normalized and activated in batches. There are two branches, the left branch is to do the result two more convolution combination operations, convolution combination operations refers to convolution, normalization, and activation. The right branch is to do a maximum pooling operation, and then concatenate the results of the two branches according to the channel, and then do a convolution combination operation after concatenation. Get the final result of the module. Pooling can reduce the size of the input feature map and reduce the underfit of the model while preserving the semantic information.
The stem block[6] is followed by three stages. The structure of stage1 and stage2 is the same. stage3 has two more aggregation layers than the previous two stages. stage1 and stage3 differ in the number of times aggregation layer 1 is used, stage1 consists of one aggregation layer 1 and one aggregation layer 2 and one LSK module, while stage3 consists of three aggregation layers 1 and one aggregation layer 2 and one LSK module.
Parallel[11] to backbone is the detail branch, which is composed of three convolution, each of which has a different number of channels. The detail branch is responsible for retaining spatial details, and spatial information is low-level feature information, so the detail branch uses a large number of channels. Therefore, the branch requires a rich channel capacity to encode rich spatial details. At the same time, because the detail branch only pays attention to the underlying details, the convolution in the detail branch adopts the method with smaller steps. Overall, the key concept of the detail branch is to use wide channels and shallow layers to deal with spatial details. In addition, the convolution of detail branches adopts deep separable convolution, which can effectively reduce the parameters of the model and improve the segmentation speed while ensuring the extraction of spatial information.

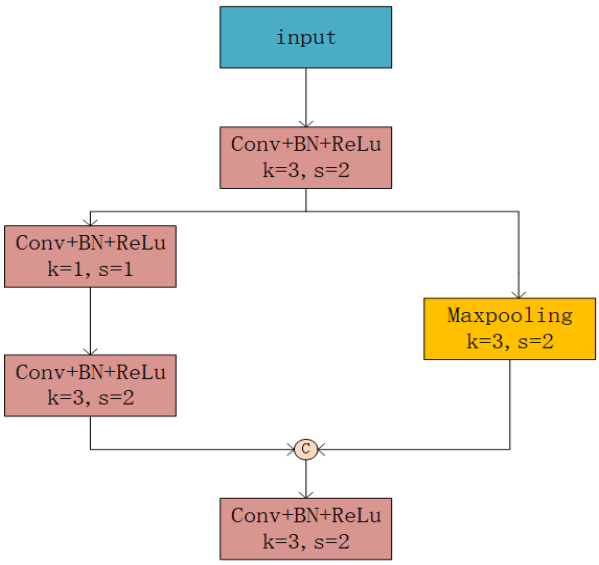


Fig. 2. stem block Diagram

B. Semantic feature

In order to ensure that a large number of context dependencies and semantic features can be extracted from the semantic branch, three stages are followed after the stem block, where stage1 and stage2 have the same structure, but differ in internal parameters, such as the number of input and output channels of convolution operations. As can be seen from Figure 3, stage1 is composed of two aggregation layers and one LSK module. The purpose of these two aggregation layers is to extract the feature information of smaller objects in the image so that they can be highlighted in the features.

The LSK module can efficiently capture long-distance dependencies, that is, provide key context information in semantic branches. Details of stage1 and stage3 are as follows:

In stage1, the input image first passes through aggregation layer 2, which is composed of six convolution and corresponding batch normalization and ReLU activation functions. Among them, INPUT needs to be computed by two branches, the left branch has four convolution combinations, the right branch has two convolution combinations, and there are two convolution types in stage1, namely 1×1 and 3×3. After the convolution operation of the left and right branches respectively, the results of the two branches are added to get the results of the aggregation layer 2.

The result of aggregation layer 2 will be sent to aggregation layer 1 as input. Unlike aggregation layer 2, aggregation layer 1 has a branch consisting of three convolution combinations, also 1×1 and 3×3 convolution, which will be added to the input after completing the three convolution operations.

After the characteristics of the two aggregation layers[8] are obtained, as an input to the attention of the large separation convolution kernel, it is able to efficiently capture long-distance dependencies and reduce the compute and memory footprint.

After the image is processed by stage1 and stage2, it is handed over to stage3 to obtain the final context information and semantic features. As this is the last step of feature extraction, three aggregation layers 1 are placed in stage3. After the features of aggregation layer 1 are obtained, LSK module is still used to obtain the context information. Capture dependency information over long distances.

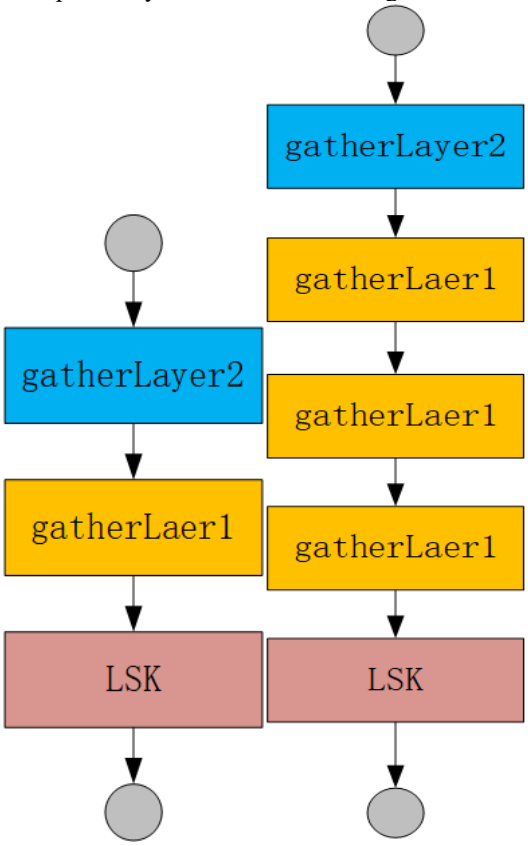


Fig. 3. stage1 and stage3 structure diagrams

C. LSK block

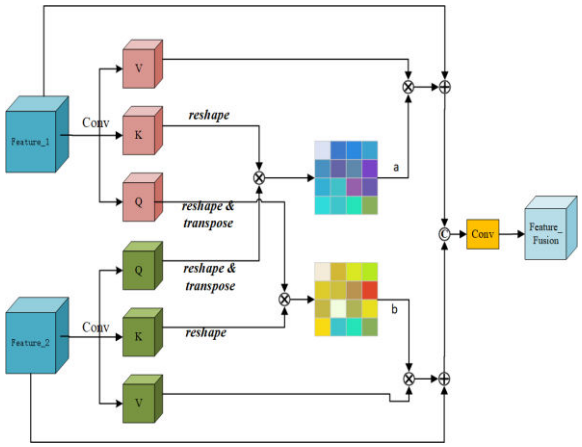
LSK module[9] can model the long-distance dependence of the input image through large convolution kernel, and effectively capture the correlation of the far-away regions in the image, helping the model better understand the overall structure and context information of the image. At the same time, it can reduce the com utation and memory consumption. The two-dimensional[10] convolutional kernel of the deep convolutional layer is decomposed into cascade horizontal and vertical one-dimensional convolutional kernel, which significantly reduces the quadratic increase of computational complexity and memory consumption caused by the increase of convolutional kernel size, and improves the operation efficiency of the model. In addition, it can also improve the context awareness ability, and can integrate multi-scale context information to better understand the background information of the target in the image, so as to make a more accurate distinction between the similar appearance of the target.

D. Feature Fusion block

The feature fusion module generates new feature maps by aggregating horizontal and vertical context information, and each position in these feature maps actually collects the information of all pixels, thus achieving the extraction of dense context features. This module not only enhances local features and increases pixel-level representation, but also provides a broad context view and selectively aggregates context information according to spatial attention map.

In addition, the fusion module is very low compute and only a few parameters are added, so the video memory usage is very low and rich context information can be obtained. This module not only improves the accuracy of semantic segmentation, but also helps to solve the inherent locality of convolutional neural networks, because the receptive field of ordinary convolutions is fixed, which can accurately capture the up and down information while ensuring a large receptive field.

Fig. 4. Feature Fusion diagrams



III. EXPERIMENTAL

A. Train

We used a 0.9 momentum stochastic gradient descent (SGD) algorithm to train our model. For all data sets, we used batch 4. For the cityscape dataset, the weight decays to 0.0003. We note that weight decay regularization is only used for parameters of the convolution layer. The initial rate is set to 5e-2, using a multi-learning rate strategy, where the initial rate is multiplied by (1−iter iter smax) power 0.9 per iteration. In addition, we conducted 150K and 10K iterative training

models on cityscape and Camvid data sets respectively. For data enhancement, we randomly flip horizontally, randomly scale, and randomly crop the input image to a fixed size for training. The random scale contains {0.75,1,1.25,1.5,1.75,2.0}. The cropping resolutions are 2048 × 1024 for cityscape and 960 × 720 for Camvid. In addition, the enhanced input from the cityscapes dataset will be adjusted to 1024 × 512 resolution to train our model.

B. Ablation Experiment

CANet contains two branches, consisting of three deeply separable convolution, each of which has a different number of channels, and adopts large channel convolution, which is responsible for preserving spatial details and complements the role of semantic branches. In order to prove the ability of the detail branch to retain spatial details, other parts of the network, including semantic branches and fusion modules, are kept unchanged, and only three convolution of the detail branch are removed. When the three depth separable convolution of the detail branch is removed, the loss of MioU is very large, which is 13.1% lower than that of the original network, which is unacceptable. Even if the trade-off is an increase in speed and a decrease in the number of participants. So branches of detail are essential in networks.

Convolution in the detail branch uses depth-separable[12] convolution, which enables cross-channel information interaction by breaking standard convolution operations into two steps: first independent convolution operations for each input channel, followed by point convolution. Depth-separable convolution can significantly reduce the number of model parameters without any significant drop in accuracy. In order to verify the necessity of the existence of depth-separable convolution in the detail branch, the depth-separable convolution is replaced by an ordinary convolution, the network structure is left unchanged, and the changes in the number of parameters, accuracy and speed of the model after this change are evaluated. When using ordinary convolution instead of deep separable convolution, the number of parameters is 0.17M more than the original network, and the network running speed is reduced by 11FPS, but the segmentation accuracy is only 0.1% higher than the original network. It is not worthwhile to exchange the relatively small segmentation accuracy with large running speed and 100,000-degree increase in the number of parameters. Therefore, it is necessary to use deep separable convolution in the detail branch.

After each stage, a large separation convolution kernel attention module is added, which can effectively capture the correlation of far-apart regions in the image and help the model better understand the overall structure and contextual information of the image. In order to verify the effectiveness of the LSK module, the LSK module in each stage is removed, and the other structure of the model is consistent with the original network. The changes in the number of parameters, accuracy and speed of the model after this change are evaluated. When LSK module is used in the stage, the number of parameters is only 0.23M higher than that without LSK, but the segmentation accuracy is greatly improved, which is 8.9% higher than that without LSK, and the running speed is only 8FPS slower than that without LSK. It is worthwhile to exchange a small number of parameters and running speed for a very high segmentation accuracy. Therefore, it is feasible to use LSK module in the stage.

Each stage consists of aggregation layer 1 and Aggregation layer 2 and the LSK module, but stage3 uses aggregation layer 1 more than the other two structures, which can help the model get better semantic information. In order to verify the impact of the number of aggregation layer 1 in stage3 on the performance of the model, the number of aggregation layer 1 in stage3 was set to 1, 2 and 4 respectively to test the performance of the model. When three aggregation layer 1 modules were used in stage3, the performance of the model was optimal. Specifically, when stage3 contains one aggregation layer 1, the number of parameters is correspondingly reduced by 0.06M, at the cost of a significant reduction in segmentation accuracy, and the same is the inclusion of two aggregation layers 1, although the FPS is improved a little, but the loss of such a high segmentation accuracy is obviously unreasonable. When stage3 contains four aggregation layers 1, the number of parameters is correspondingly increased by 0.03M, at the cost of lower segmentation accuracy and lower running speed.

The parallel attention module aims to learn the importance weights of each channel, and focus on the features of the key areas, while the features of the unrelated background areas are weakened. To verify its effect on model performance, the module is removed from the network and the other structures are left unchanged. When the parallel attention module is not used, the reference count is 0.18M less than CANet, and it runs 6FPS faster, but the segmentation accuracy is reduced by 5.1%, which is not worth it. Therefore, it is necessary to use parallel attention module in CANet.

IV. Conclusion

Aiming at the problems of large number of parameters, low segmentation accuracy and high model complexity in current real-time semantic segmentation methods, this paper proposes a model that can achieve balance between speed and accuracy. Through the design and optimization of the model, the experimental results on a large number of data sets show that the model has a high segmentation accuracy, a very low number of parameters, and a high reasoning speed.

References

[1] Yu C, Wang J, Peng C, et al. Bisenet: Bilateral segmentation network for real-time semantic segmentation[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 325-341.

[2] 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: 3051-3068.

[3] Nasreen, Ghazala, et al. "A comparative study of state-of-the-art skin image segmentation techniques with CNN." Multimedia Tools and Applications 82.7 (2023): 10921-10942.

[4] Cham, Switzerland, RISC-V Specification, vol. 1, Unprivileged Spec V, 2019.

[5] A. Sanchez-Flores, L. Alvarez and B. Alorda-Ladaria, "Accelerators in Embedded Systems for Machine Learning: A RISCV View," 2023 38th Conference on Design of Circuits and Integrated Systems (DCIS), Málaga, Spain, 2023.

[6] Zhang F, Li Y, Ye Z. Apply yolov4-tiny on an fpga-based accelerator of convolutional neural network for object detection[C]//Journal of Physics: Conference Series. IOP Publishing, 2022, 2303(1): 012032.

[7] Guo Z, Liu K, Liu W, et al. An Overlay Accelerator of DeepLab CNN for Spacecraft Image Segmentation on FPGA[J]. Remote Sensing, 2024, 16(5): 894.
[8] Mani V.R.S, Saravanaselvan A, Arumugam N. Performance comparison of CNN, QNN and BNN deep neural networks for real-time object detection using ZYNQ FPGA node[J]. Microelectronics Journal, 2022, 119: 105319.
[9] Wu N, Jiang T, et al. A reconfigurable convolutional neural network-accelerated coprocessor based on RISC-V instruction set[J]. Electronics, 2020, 9(6): 1005.
[10] Pestana D, Miranda P R, Lopes J D, et al. A full featured configurable accelerator for object detection with YOLO[J]. IEEE Access, 2021, 9: 75864-75877.
[11] Liu S, Fan H, Ferianc M, et al. Toward full-stack acceleration of deep convolutional neural networks on FPGAs[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 33(8): 3974-3987.
[12] Nguyen D T, Je H, Nguyen T N, et al. ShortcutFusion: From tensorflow to FPGA-based accelerator with a reuse-aware memory allocation for shortcut data[J]. IEEE Transactions on Circuits and Systems I: Regular Papers, 2022, 69(6): 2477-2489.