Improve SegNet with feature pyramid for road scene parsing

Xinbo Ai[1,2,*], *Yunhao* Xie[1,2] , *Yinan* He[1,2], and *Yi* Zhou[3]

[1]School of Artificial Intelligence, Beijing University of Posts and Telecommunications, 100876, Beijing, China
[2]Beijing Key Laboratory of Work Safety Intelligent Monitoring, 100876 Beijing, China
[3]Beijing Academy of Safety Science and Technology, Beijing, China

Abstract. Road scene parsing is a common task in semantic segmentation. Its images have characteristics of containing complex scene context and differing greatly among targets of the same category from different scales. To address these problems, we propose a semantic segmentation model combined with edge detection. We extend the segmentation network with an encoder-decoder structure by adding an edge feature pyramid module, namely Edge Feature Pyramid Network (EFPNet, for short). This module uses edge detection operators to get boundary information and then combines the multiscale features to improve the ability to recognize small targets. EFPNet can make up the shortcomings of convolutional neural network features, and it helps to produce smooth segmentation. After extracting features of the encoder and decoder, EFPNet uses Euclidean distance to compare the similarity between the presentation of the encoder and the decoder, which can increase the decoder's ability to restore from the encoder. We evaluated the proposed method on Cityscapes datasets. The experiment on Cityscapes datasets demonstrates that the accuracies are improved by 7.5% and 6.2% over the popular SegNet and ENet. And the ablation experiment validates the effectiveness of our method.

1 Introduction

Road scene parsing is a fundamental but challenging task in computer vision. There are mainly two main kinds of approaches: the traditional segmentation method[1, 2] and the segmentation method based on deep learning. Traditional methods tend to have good results in simple scenes. However, the context of images with many vehicles and pedestrians is usually complex and changeful. The same category objects have a large difference in colour and appearance[3]. Therefore, these methods are rarely used alone in the road scene parsing. With the continuous development of deep learning, convolutional neural networks have achieved great success in image feature extraction[4-5]. Therefore, we design an edge feature pyramid module to improve the performance of the semantic segmentation model. The main contributions of our work are as follows: (1) We design a novel edge feature pyramid module for edge feature extraction. This module will help

* Corresponding author: axb@bupt.edu.cn

merge the manual features from different stages. Based on this way, the ability to extract edge features is enhancing, so that the model can get semantic boundary feature easily. Eventually, the model can obtain features that are more conducive to road segmentation, thereby improving segmentation accuracy.(2) For our edge feature pyramid module, a new hybrid loss function is defined. By using it, our model will improve the ability of the decoder to restore high-level semantic features and help itself to further improve the accuracy.

2 Related work

In 2015, Shelhamer[6] proposed a model called Fully Convolutional Network (FCN), which removes the common fully connected layer in the CNN and laid the framework for solving semantic segmentation tasks. Then Badrinarayanan et al[7] proposed an encoder-decoder model called SegNet for image segmentation in 2016. Since then, many other methods have been proposed to enhance the performance of segmentation, like ENet[8], UNet[9], DeepLab family of algorithms[10-14], PSPNet[15] and so on.

The above methods are all based on the deep CNN. However, only relying on deep learning methods is not enough to complete changeful and complex tasks, nor can it deal with specific problems in a targeted manner[16]. So, there are some examples of fusing various operator with convolution features[17-18]. In fact, common edge[19-20] can help neural network to extract semantic boundaries contained in the edges of objects. Besides, there are also edge detection methods based on deep learning[21].

There are already some semantic segmentation models that combine edge detection and deep learning. Marmanis et al[22] combine edge detection with semantic segmentation which uses the image after edge detection as input. On the contrary, Huang et al[23] combine the two process to improve the overall segmentation effect. Song et al[24] build a semantic boundary detection subnetwork by the method of multitask learning. However, this method will increase the difficulty of convergence of the overall network to a certain extent easily[25]. Considering about the above methods, we improve the SegNet model with a new designed pyramid module which can obtain edge features from multi-scale stages.

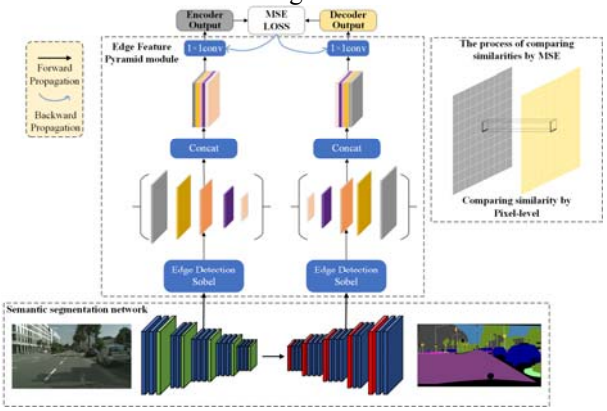

Fig. 1. Our proposed model EFPNet. Dashed box on the upper right manifests the way to compare similarities.

3 Proposed methods

We propose a semantic segmentation model combined with edge detection Sobel operator, the whole model is shown Figure 1. This model is mainly composed of two parts:

the semantic segmentation network and edge feature pyramid module. We use SegNet as the basic semantic segmentation model. The structure of the SegNet model is shown at the bottom of Figure 1. SegNet is a network model of autoencoder type, which is composed of an encoder and a decoder. The encoder network in SegNet is topologically identical to the first 13 convolutional layers in VGG16[26]. And the decoder network and the encoder network are completely symmetrical.

3.1 Edge feature pyramid

The feature maps obtained by each stage in SegNet are subjected to edge detection. The edge feature extraction process is same in each stage. In the module, after each convolutional layer of SegNet, a 1×1 convolutional layer is used for dimension reduction. The number of output channels is 19, which represents the number of the road categories.

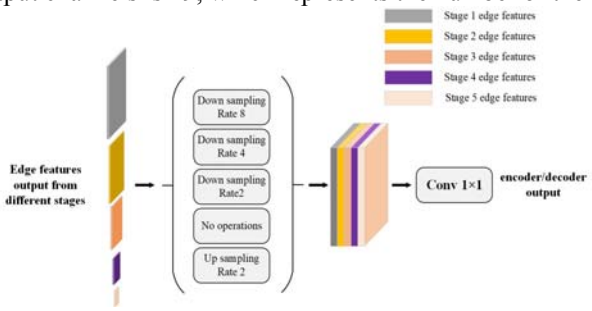

Fig. 2. Edge Feature Pyramid. Handling different stages edge features by sampling operations to harvest features of same size. Then we use a 1×1 convolution layer to help features fuse.

As the feature map size after different convolution layers under the same stage remains constant. Therefore, these features can be enhanced by simple linear addition. By this way, we get the edge information feature map of the feature map under this stage.

To reflect the edge information representation ability of the entire encoder, we need to perform a multiscale fusion of the feature maps obtained at each stage. A feature pyramid module is introduced to help feature fusion of the feature maps of each stage here. The architecture of the module is shown in Figure 2. Considering that up-sampling will lose some information and the complementarity between features will be impaired, it is not suitable for this fusion method in which feature maps are directly added. We use the 1×1 convolution to help the fusion of features of multiple scales. This operation can guarantee the preservation of the original information and obtain appropriate subsequent features.

3.2 Loss function

To be able to measure the gap between the edge feature representation capabilities of the encoder and the decoder, the model adopts a hybrid supervision method on the loss function, which is defined as Equation 1:

$$loss_{total} = loss_{FL} + \alpha loss_{mse} \tag{1}$$

The first term is the Focal loss[27], and the second one is the Mean Squared Error (MSE) loss. Focal loss is a loss function proposed for the imbalance of categories. It improves the cross entropy(CE) loss function by adding a modulation coefficient before CE. Focal Loss is defined as Equation 3. Through this modulation coefficient, the learning process of the model can be more focused on the learning of hard negative examples.

$$p_t = \begin{cases} p & if \quad y=1 \\ 1-p & otherwise \end{cases} \tag{2}$$

$$FL(p_t) = -(1-p_t)^{\gamma} \log(p_t) \tag{3}$$

For the model of the encoder-decoder structure, we believe that the final output of the decoder should be like the input as possible [28]. So, we measure the feature extraction capability of the encoder and decoder by calculating the similarity of the edge feature maps, which can make the presentation of the decoder approach the encoder. Comparing the feature expression ability of the two is a regression task, so we use the MSE loss function. The calculation formula of the MSE is shown in Equation 4, where Y represents output, and subscripts D and E represent decoder and encoder respectively.

$$MSE = \frac{1}{n}\sum (Y_D - Y_E)^2 \tag{4}$$

$$\alpha = \frac{cur\_epoch}{total\_epoch} \tag{5}$$

And we add a dynamic constant coefficient alpha before MSE. This coefficient is the ratio of the current training epoch to the total epoch. The calculation formula is as Equation 5. The existence of alpha is to balance the proportion of the two loss functions in the overall loss in the backpropagation process. Since the decoder has not learned the expression of the dataset during the initial training, the expression ability is limited. With the continuous training of the model, the decoder gradually learns the expression of the segmentation feature. At the same time, the coefficient will increase the proportion of MSE loss function which stands for the edge difference in the overall loss function, to correctly supervise the training of the model.

In summary, when combining the above two loss functions, Focal Loss can be used to provide the gradient of semantic segmentation. MSE can help the decoder better acquire the edge representation ability and provide the gradient for the entire model as well, thereby improving image semantic segmentation accuracy.

4 Results and discussion

4.1 Implementation details

The algorithm in this paper is implemented based on the deep learning framework Pytorch. The GPU used is two pieces of 16G NVIDIA Tesla P100. The evaluation method used in this paper is mean Intersection over Union (mIoU), which is used on the Cityscapes dataset[29] in this paper. In the training of the model, we use stochastic gradient descent (SGD) as the optimization algorithm to train all the variants with an initial learning rate of 0.001 and a momentum of 0.9. The sample batch size is set to 24, the weight decay coefficient is set to 5×10^{-4}, and the maximum epoch is set to 700. For faster training, we load all the first 13 convolutional layers weights of VGG16 into the model and the remaining parameters are initialized using the kaiming initialization[30].

4.2 Experiments results

We compare the segmentation results obtained by the method proposed in this paper with those obtained by SegNet. The comparison between the segmentation results of the two methods and the label of the dataset is shown in Figure 3.

It can be seen from scene 1 that it is difficult to distinguish the sidewalk from the road due to the presence of shadows. Our method basically recognizes the road turning right,

which SegNet cannot. In scene 2, SegNet caused a misclassification due to the small truck target, while EFPNet is still sensitive to small objects. In segmentation diagram of scene 3, the light pole can be identified by our method, which is discontinuous in SegNet result picture. The detailed information about road in scene 4 is misclassifying, but our model can recognize accurately.

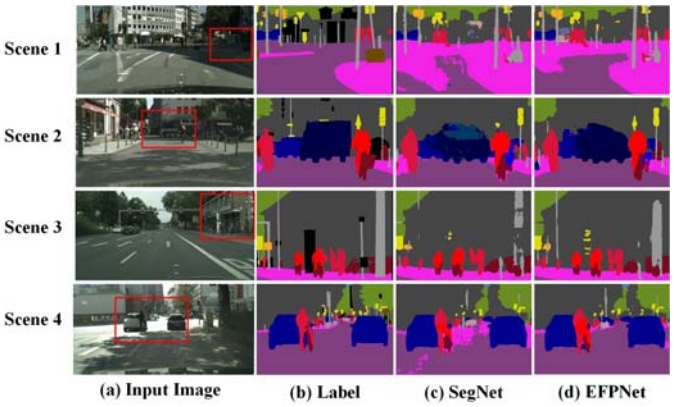

Fig. 3. These figures are enlarged version of these picture above. For each input image, we show its label, SegNet result and EFPNet result respectively.

Table 1. Comparison of accuracy with other algorithms.

Methods	Road	Sidewalk	Building	Wall	Fence	Pole	Traffic light	Traffic sign	Tree	Ground
Segnet	96.4	73.2	84.0	28.4	29.0	25.7	29.8	45.1	87.0	63.8
ENet	96.3	74.2	75.0	32.2	33.2	43.4	34.1	44.0	88.6	61.4
EFPNet	95.9	**75.8**	**90.0**	**42.5**	**50.5**	**59.7**	**64.5**	**73.4**	**91.6**	60.0

Methods	Sky	Person	Rider	Car	Truck	Bus	Train	Motor cycle	bicycle	mIoU
Segnet	91.8	62.8	52.8	89.3	38.1	43.1	44.1	35.8	51.9	57.0
ENet	90.6	65.5	38.4	90.6	36.9	50.5	48.1	38.8	55.4	58.3
EFPNet	**93.3**	**75.9**	**48.4**	**91.6**	**39.3**	**61.6**	27.5	**45.6**	**69.8**	**66.1**

Besides, we compare the model EFPNet proposed in this paper with some other existing methods for specific categories. Table 1 shows the comparison of the accuracy of several methods in 19 categories on the Cityscapes dataset. It shows from the Table 1 that our method has a great improvement in most categories. Also, the categories that are sensitive to edge features have been greatly improved, which also proves the effectiveness of our proposed method.

4.3 Ablation experiment

To verify the necessity and effectiveness of the edge detection operator in our method, we conducted some ablation experiments on the Cityscapes dataset.

We remove the Sobel operator in the edge detection part of the encoder and decoder and check the performance of this model. The accuracy of model with Sobel is 65.3% and that

of model without Sobel is 63.2%, which prove that the existence of Sobel operator in the entire model is meaningful. The edge detection operator is a manual features extractor, which can only extract the edge features of the entire feature map. That will be a kind of degradation of the complexity of the features. However, the undetected feature map is directly used for similarity comparison, which results that the model is difficult to converge due to the high complexity of the feature map.

We select four different edge detection operators and put them into the edge detection pyramid module and compare their effects. The results are shown in Table 2.

Table 2. The impact of Sobel operators on performance.

	sobel	prewitt	laplacian	LoG
Pixel accuracy(%)	82.130	82.127	81.936	81.907
mIoU(%)	65.36	65.36	63.82	73.79

From the above results, we can find that Sobel and Prewitt operators have similar performance, while the second-order operators perform worse. This is mainly because these two second-order operators are mainly used to filter out noise. Their edge detection capabilities are not as good as two first-order operators.

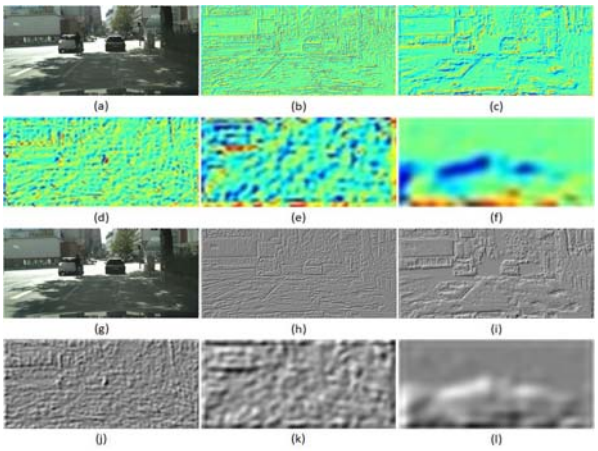

Fig. 4. Visualized feature maps of our model from each stage.

Table 3. comparison of different basic stage in our algorithm.

	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5
Pixel accuracy(%)	82.130	82.366	82.305	82.366	82.248
mIoU(%)	65.36	66.12	66.07	66.13	66.05

In addition, we also compare the model segmentation performance of different stages, which is treated as the basic stage in pyramid module. From Table 3, we can see that there is a certain gap in the experimental results obtained by selecting different stages as the basic scale. Among them, stage 2 and stage 4 have better effects and can be used as basic scales. Figure 4 shows the edge feature maps and corresponding heat maps obtained after different stages. We can see that because stage 1 has a larger scale and less semantic information. The low-level information of the feature map obtained is more complete, but the boundary information is not obvious enough. For stage 2 and stage 4, it can be seen from the heat map that the boundary is more obvious.

5 Conclusions

Modern traffic road scenes are so complex that the scales of objects in the same category

vary greatly. Common segmentation methods often have problems like discontinuous edges and the difficulty of identifying small objects. To solve the above problems, we design an edge pyramid module to help model get semantic boundary features derived from feature pyramid and other methods. By using the module, our model is more sensitive to boundary features and effectively improves the ability to extract semantic edge. Finally, we validate the effectiveness of our method on the Cityscapes. The results manifest that this method effectively improves the problem of edge blur in semantic segmentation and greatly improves the accuracy of segmentation.

This study was funded by Beijing Young Backbone Individual Program (No.2018742603767G301), National Natural Science Foundation of China (No.61503034), and Safety and Emergency Key Technology Development Plan of Beijing Emergency Management Bureau (No.202002009).

References

1. R. Adams, and L. Bischof, "Seeded region growing," IEEE Transactions on Pattern Analysis and Machine Intelligence, **16**, 6 (1994).
2. L. Vincent, and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," IEEE Transactions on Pattern Analysis and Machine Intelligence, **13**, 6 (1991)
3. F. F. Kong, and B. B. SONG, "Improved YOLOv3 panoramic traffic monitoring target detection". Computer Engineering and Applications, **56**, 8 (2020)
4. Z. J. Sun, L. Xue, Y. M. Xu, and Z. WANG, "Overview of deep learning," Application Research of Computers, **29**, 8 (2012)
5. T. Y. Lin, Dollár, Piotr, G. Ross, K. M. He, H. Bharath, and B. Serge, "Feature Pyramid Networks for Object Detection," in 2017 *IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Hawaii, 2017)
6. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in the Proceedings of the IEEE Conference on *Computer Vision and Pattern Recognition*, (IEEE, 2015).
7. V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation". IEEE Transactions on Pattern Analysis and Machine Intelligence, **39**, 12 (2017)
8. A. Paszke, A. Chaurasia, S. Kim and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," arXiv preprint arXiv:1606.02147,2016.
9. R. Olaf, F. Philipp, and B. Thomas "U-Net: Convolutional Networks for Biomedical Image Segmentation" in the Proceedings of International Conference on *Medical Image Computing and Computer-Assisted Intervention*. pp. 234-241. 2015.
10. S. Ghosh, N. C. Das, I. Das, and U. Maulik, "Understanding Deep Learning Techniques for Image Segmentation," ACM Computing Surveys(CSUR), 2019.
11. L. C. Chen, P. George, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs," in the International Conference on *Learning Presentations* (2015)
12. L. C. Chen, P. George, S. Florian, and A. Hartwig, "Rethinking Atrous Convolution for Semantic Image Segmentation." in 2017 IEEE Conference on *Computer Vision and Pattern Recognition*, (IEEE, Hawaii, 2017)

13. L. C. Chen, Y. Zhu, P. George, S. Florian, A. Hartwig, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation." In the Proceedings of *European Conference of Computer Vision*, pp. 801-818 (Munich, Germany, 2018)
14. Y. Fisher, K. Vladlen "Multi-Scale Context Aggregation by Dilated Convolutions," in the *International Conference on Learning Presentations* (2015)
15. H. S. Zhao, J. P. Shi, X. J. Qi, X. G. Wang, and J. Y. Jia. "Pyramid Scene Parsing Network," in 2017 IEEE Conference on *Computer Vision and Pattern Recognition*, (IEEE, Hawaii, 2017).
16. H. Y. Hou, T. Gao, and T. Li, "Overview of Image Segmentation Methods," Computer Knowledge and Technology, **15**, 5 (2019)
17. H. J. Jun, B. S. Ko, Y. J. Kim, I. Kim, and J. Kim, "Combination of Multiple Global Descriptors for Image Retrieval," arXiv:1903.10663 (2019)
18. F. Z. Liu, Y. D. Xia, D. Yang, A. Yuille, and D. G. Xu, "An Alarm System for Segmentation Algorithm Based on Shape Model," arXiv:1903.10645 (2019)
19. J. M. PREWITT, "Object Enhancement and Extraction," Picture Processing and Psychopictoric Press, 75-149 (1970)
20. J. CANNY "A computational approach to edge detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, **8**, 6 (1986)
21. Y. Liu, M. M. Cheng, X. W. Hu, J. W. Bian, L. Zhang, X. Bai, and J. H. TANG, "Richer Convolutional Features for Edge Detection," in 2017 IEEE Conference on *Computer Vision and Pattern Recognition*, (IEEE, Hawaii, 2017)
22. D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Dtcu, and U. Stiila, "Classification with an edge: improving semantic image segmentation with boundary detection," ISPRS Journal of Photogrammetry and Remote Sensing, **135** 158-172 (2018)
23. Q. Huang, C. Y. Xia, W. C. Zheng, Y. H. Song, H. Xu, and K. C. C. Jay, "Object boundary guided semantic segmentation," in Proceedings of the 13th *Asian Conference on Computer Vision*, (Springer, Berlin, 2016)
24. X. N. Song, T. Rui, and X. Q. Wang, "Semantic segmentation method of road environment combined semantic boundary information," Journal of Computer Applications, **39**, 9 (2019)
25. R. Sebastian "An Overview of multi-Task Learning in Deep Neural Networks," arXiv: 1706.05098v1 (2017)
26. K. Simonyan, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition" in the *International Conference on Learning Presentations* (2015)
27. Y. T. Lin, P. Goyal, R. Girshick, K. M. He, and P. Dollar, "Focal Loss for Dense Object Detection." in 2019 IEEE Conference on *Computer Vision and Pattern Recognition* (IEEE, 2019)
28. M.A. Kramer, "Autoassociative neural networks," Computers and Chemical Engineering, **16**, 4 (1992)
29. M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding". arXiv:1604.01695v2 (2016)
30. K. M. He, X. Y. Zhang, S. Q. Ren, and J. Sun, "Delving Deep into rectifiers: Surpassing Human-Level Performance on ImageNet Classification," Presented at *International Conference of Computer Vision* (2016)