Lightweight spatial pyramid pooling network for real-time semantic segmentation

Peilong Jiang\textsuperscript{1,∗} and Xiao Ke\textsuperscript{1}

\textsuperscript{1}Fujian Provincial Key Laboratory of Networking Computing and Intelligent Information Processing, College of Computer and Data Science, Fuzhou University, Fuzhou 350116, China

N190320025@fzu.edu.cn

Abstract. In recent years, the state-of-the-art semantic segmentation models have made extremely successful in various challenging scenes. However, the high computation costs of these models make it difficult to deploy to mobile devices. To better serve in computation constraint scenes, the semantic segmentation model should not only have high segmentation performance, but also fast inference speed. In this paper, we proposed an efficient multi-scale context module named LSPPM, which can gather abundant context information at a low computation cost. Base on this, we present a real-time semantic segmentation model called LSPPNet which is specially designed for real-time application. We have done an exhaustive experiment to evaluate LSPPNet in the challenge urban street scenes datasets Cityscapes. Extensive experiment shows that LSPPNet gets a better trade-off between segmentation performance and inference speed. We test LSPPNet on an NVIDIA 2080 super graphics card and it can achieve 75.8% MIoU in Cityscapes test set in real-time speed.

1. Introduction

Semantic segmentation is one of the three fundamental tasks in computer vision. The goal is to label each pixel of the input image with the class corresponding to the pixel. Semantic segmentation has many application scenarios in areas including autonomous driving systems and robotics-related fields, which require not only good segmentation performance, but also fast inference speed. Existing semantic segmentation models, such as ENet\textsuperscript{[1]} and ESPNet\textsuperscript{[2]}, have very fast inference speed, but this fast inference speed is obtained by reducing the segmentation performance. Some other networks such as ContextNet\textsuperscript{[3]} and ICNet\textsuperscript{[4]} have achieved good performance on segmentation, but they sacrifice inference speed and model size, respectively. These semantic segmentation networks cannot meet the requirements of real-time semantic segmentation (typically 30 frames/second). Therefore, in this paper, we aim to find a better trade-off between segmentation performance and inference speed to better serve in real-time scenarios.

In this paper, a novel lightweight multi-scale context module is proposed and a real-time semantic segmentation network LSPPNet is built by applying this module. The main contributions of this paper are as follows.

1. We propose an efficient multi-scale context module, called LSPPM (Lightweight Spatial Pyramid Pooling Module), which combines adaptive averaging pooling and 3 × 3 convolution. This
module extracts feature maps and context information of different sizes jointly by pooling at different scales and has the advantage of small computational cost with a small number of parameters.

2. LSPPNet (Lightweight Spatial Pyramid Pooling Network) is designed on top of LSPPM. LSPPNet achieves a good balance between inference speed and segmentation performance, making it well suited for deployment into performance-constrained devices.

3. LSPPNet can process high-resolution images (2048×1024) at 30FPS on a single NVIDIA 2080 super graphics card and achieves 75.8% MIoU on the Cityscapes test set.

2. Analysis of the pyramid pooling module

In image recognition tasks, the accuracy of recognition largely depends on the perceptual field size of the network. Although the theoretical perceptual field of ResNet[5] can already cover or even exceed the size of the input image, Bolei Zhou et al.[6] show that the perceptual field of convolutional neural networks in practice is much smaller than the theoretically calculated perceptual field, so a global context module is needed to further expand the perceptual field in convolutional neural networks.

![Pyramid pooling module](image)

Figure 1. Pyramid pooling module.

The global average pooling can be a good baseline for the global context module, and it has shown good results in image classification tasks, and the performance in semantic segmentation tasks is also being fully verified to be effective. Given this, Heng-Shang Zhao et al.[7] proposed the PPM (Pyramid Pooling Module). In Fig. 1, the network structure of PPM is shown. To further reduce the loss of background information between different sub-regions, a hierarchical pooling structure is designed by Heng-Shang Zhao et al.

The hierarchical pooling scales of PPM are set to 1×1, 2×2, 3×3, and 6×6, which include the global average pooling mentioned in the previous paragraph. In their experiments, Hengshang Zhao et al. concluded that the average pooling is significantly better than the maximum pooling in expanding the perceptual field, and both the scale and the number of hierarchical pooling can be adjusted so that sub-regions of different sizes can be abstracted by using different numbers of pooling kernels of different sizes to obtain contextual information at different scales.

3. The proposed method

3.1. Lightweight spatial pyramid pooling module

The context module in the semantic segmentation task benefits from the excellent performance of pyramid pooling by expanding the network perceptual field, thus improving the segmentation results of the model against multi-scale object edges. In this paper, the LSPPM (Lightweight Spatial Pyramid Pooling Module) is proposed, and its network structure is shown in Fig. 2.

The LSPPM adopts adaptive pooling as the basic operator of the module like PPM. First, LSPPM passes the input representation through four different levels of adaptive averaging pooling layers with pooling kernel sizes of 1×1, 2×2, 3×3, and 6×6, to fully obtain the feature maps at different scales. Then the input representation is downscaled through a 3×3 convolutional layer, followed by a residual connection. Connect between the output of the previous stage and the output of the next adaptive averaging pooling layer after downsampled.
The biggest difference between LSPPM and PPM is that the 1×1 convolution used for dimensionality reduction is changed to 3×3 invariant convolution. This change can improve the efficiency of dimensionality reduction and avoids the problem of insufficient mapping capability of 1×1 convolution due to too few parameters. Since the representations after dimensionality reduction and upsampling all have the same number of channels and resolution, the way of feature fusion is replaced by the residual connection from the dense connection to avoid speed drop due to the expansion of the channel number caused by representation stitching.

![Figure 2. Lightweight spatial pyramid pooling module.](image)

### 3.2. Network architecture

The network structure of LSPPNet (Lightweight Spatial Pooling Network) is shown in Fig. 3. We use ResNet18 as LSPPNet’s encoder which is the combination of stems and layers1-4 in Fig. 3. ResNet has a large number of learnable parameters and powerful feature representation capabilities. Considering that the model needs to be applied to real-time scenes, LSPPNet uses ResNet with depth 18, while removing its final average pooling layer and fully connected layer. The context module, on the other hand, uses the LSPPM proposed in Section 3.1, which expands the perceptual field of the network and improves the segmentation results of the model against multi-scale objects. It is worth mentioning that this chapter introduces DSB (Deep Supervision Block) in layer4 of the encoder, which improves the efficiency of feature fusion in the upsampling phase.

![Figure 3. LSPPNet network architecture.](image)

### 4. Experimental results

To fully explore the effectiveness of LSPPNet in real-time segmentation scenarios, Cityscapes is used in the experiment. The Cityscapes dataset focuses on semantic understanding in urban road scenes. The dataset is collected from 50 different cities in different seasons and weather with richer scenes. The resolution of all pictures is 1024×2048.

In this paper, MIoU (Mean Intersection over union) is used to measure the segmentation performance of a model, and the formula of MIoU is shown as follows.

\[
\text{MIoU} = \frac{\sum_{i=1}^{N} |S_i \cap G_i|}{\sum_{i=1}^{N} |S_i| + \sum_{i=1}^{N} |G_i| - \sum_{i=1}^{N} |S_i \cap G_i|}
\]
\[ M_{\text{IoU}} = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{n} p_{ij} + \sum_{j=1}^{n} p_{ji} - p_{ii} \]  

The performance comparison in the Cityscapes test set is shown in Table 1. The main goal of the design principle is to achieve the best possible segmentation performance while maintaining real-time speed (30fps). Since the use of different GPUs for different methods has a significant impact on speed, the GPUs used in the experiments for each method are also reported in the table. To avoid losses in segmentation performance, in the experiments LSPPNet does not do any preprocess (downsample or scale) on the input image. In Table 1 it can be seen that LSPPNet has a significant advantage in segmentation performance compared to other methods.

Table 1. Speed and accuracy comparison on Cityscapes test set.

| Method      | GPU    | Speed(FPS) | Param(M) | MIoU(%) |
|-------------|--------|------------|----------|---------|
| SQ[8]       | TitanX | 16.7       | -        | 59.8    |
| SegNet[9]   | TitanX | 16.7       | 29.5     | 57      |
| ERFNet[10]  | TitanX | 11.2       | 20       | 69.7    |
| CAS[11]     | TitanXp| 108        | -        | 70.5    |
| GAS[12]     | TitanXp| 108.4      | -        | 71.8    |
| DABNet[13]  | 1080Ti | 27.7       | 0.76     | 70.1    |
| Fast-SCNN[14]| TitanXp| 123.5     | 1.1      | 68      |
| CGNet[15]   | TitanXp| 31         | 0.5      | 68.3    |
| ADSCNet[16] | 1080Ti | 20.8       | -        | 67.5    |
| LEANet[17]  | 1080Ti | 77.3       | 0.74     | 71.9    |
| JPANet[18]  | 1080Ti | 109.9      | 3.49     | 71.62   |
| LSPPNet     | 2080 s | 31         | 15.37    | 75.8    |

The results of LSPPNet visualized on the Cityscapes validation set are shown in Fig. 4. The images from left to right are the original input image, the labeled image, and the LSPPNet predicted result.

![Figure 4](image_url)  
(a) image  (b) label  (c) predicted  

**Figure 4.** Qualitative results on Cityscapes validation set when employing our best model.

5. Conclusion
In this paper, we propose a novel lightweight pyramidal pooling module for expanding the perceptual field of the network and enhancing the segmentation results of the model against multi-scale objects, and design a lightweight pyramidal pooling network based on it. Exhaustive experiments are
conducted on the urban streetscape dataset Cityscapes, and the results show that the proposed method achieves a better trade-off in terms of speed and performance.

References

[1] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, “Enet: A deep neural network architecture for real-time semantic segmentation,” ArXiv Prepr. ArXiv160602147, 2016.

[2] S. Mehta, M. Rastegari, A. Caspi, L. Shapiro, and H. Hajishirzi, “Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation,” in Proceedings of the european conference on computer vision (ECCV), 2018, pp. 552–568.

[3] R. P. Poudel, U. Bonde, S. Liwicky, and C. Zach, “Contextnet: Exploring context and detail for semantic segmentation in real-time,” ArXiv Prepr. ArXiv180504554, 2018.

[4] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, “Icnets for real-time semantic segmentation on high-resolution images,” in Proceedings of the European conference on computer vision (ECCV), 2018, pp. 405–420.

[5] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[6] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, “Object detectors emerge in deep scene cnns,” ArXiv Prepr. ArXiv14126856, 2014.

[7] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2881–2890.

[8] M. Treml et al., “Speeding up semantic segmentation for autonomous driving,” 2016.

[9] V. Badrinarayanan, A. Kendall, and R. Cipolla, “Segnet: A deep convolutional encoder-decoder architecture for image segmentation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 12, pp. 2481–2495, 2017.

[10] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, “Erfnet: Efficient residual factorized convnet for real-time semantic segmentation,” IEEE Trans. Intell. Transp. Syst., vol. 19, no. 1, pp. 263–272, 2017.

[11] Y. Zhang, Z. Qiu, J. Liu, T. Yao, D. Liu, and T. Mei, “Customizable architecture search for semantic segmentation,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 11641–11650.

[12] P. Lin, P. Sun, G. Cheng, S. Xie, X. Li, and J. Shi, “Graph-guided architecture search for real-time semantic segmentation,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 4203–4212.

[13] G. Li, I. Yun, J. Kim, and J. Kim, “Dbnnet: Depth-wise asymmetric bottleneck for real-time semantic segmentation,” ArXiv Prepr. ArXiv190711357, 2019.

[14] R. P. Poudel, S. Liwicky, and R. Cipolla, “Fast-scnn: Fast semantic segmentation network,” ArXiv Prepr. ArXiv190204502, 2019.

[15] T. Wu, S. Tang, R. Zhang, J. Cao, and Y. Zhang, “Cgnet: A light-weight context guided network for semantic segmentation,” IEEE Trans. Image Process., vol. 30, pp. 1169–1179, 2020.

[16] J. Wang, H. Xiong, H. Wang, and X. Nian, “ADSCNet: asymmetric depthwise separable convolution for semantic segmentation in real-time,” Appl. Intell., vol. 50, no. 4, pp. 1045–1056, 2020.

[17] X.-L. Zhang, B.-C. Du, Z.-C. Luo, and K. Ma, “Lightweight and efficient asymmetric network design for real-time semantic segmentation,” Appl. Intell., pp. 1–16, 2021.

[18] X. Hu, L. Jing, and U. Sehar, “Joint pyramid attention network for real-time semantic segmentation of urban scenes,” Appl. Intell., pp. 1–15, 2021.