Dual ASPP for Lightweight Semantic Segmentation on High-Resolution Image

Dongpeng Xiao [*1], Meiling Wang [*2], Lin Zhao [*3] and Siyuan Chen [*3]

[*1] Beijing Institute of Technology, Beijing 100081, China
E-mail: xdp0425@163.com
[*2] Beijing Institute of Technology, Beijing 100081, China
E-mail: wangml@bit.edu.cn
[*3] Beijing Institute of Technology, Beijing 100081, China

Abstract. In recent years, the efficient and lightweight convolutional neural networks (CNNs) such as ShuffleNet and MobileNet, have been widely applied in the field of image classification. But in image semantic segmentation, challenges remain a lot. Although many network models perform well in semantic segmentation tasks, most of them contain large parameters and suffer from high computational complexity. In this paper, we explored the application of lightweight CNNs and atrous spatial pyramid pooling (ASPP) module in semantic segmentation. In the model, MobileNetV2 / MobileNetV3 were chosen as encoders and a new segmentation head named Dual ASPP was proposed as decoder, which was an improved version of DeepLabV3+. By this method, the amount of parameters can be compressed from 47.73M to 1.03M, and the computation amounts can be reduced from 458.5G to 26.1G accordingly. Besides, while testing on the high-resolution (1024×2048) Cityscapes datasets, the accuracy of the proposed lightweight model is significantly improved up to 2%.

Keywords: Semantic Segmentation, Lightweight Neural Network, Dual ASPP Segmentation Head

1. INTRODUCTION

Semantic segmentation is one of the main tasks of computer vision. It is widely used in the fields such as autonomous driving, medical image processing, robot autonomous navigation, and video surveillance security. Semantic segmentation can achieve pixel-level classification for images so that it can be also considered as a dense prediction problem [1]. Conventional approaches such as N-Cut [2] and G-Cut [3] can only accomplish simple segmentation tasks and require manual intervention.

Since Fully Convolutional Network (FCN) [4] was proposed in 2016, the semantic segmentation task has been able to achieve end-to-end input and output by using convolutional neural networks (CNNs) as encoders. Subsequently, many excellent segmentation algorithms appeared in this field, such as U-Net [5], SegNet [6] and DeepLabV3+ [7]. In order to improve the accuracy, most of CNN-based models choose heavy CNN such as VGG [8] and ResNet [9] as encoders. Although performing well in semantic segmentation tasks, most of them contained large parameters and suffered from high computational complexity and slow inference speed. In some applications, however, such as autonomous driving, the speed of model inference is crucial.

In this work, two types of neural networks were taken into consideration that use lightweight CNNs for image classification and atrous spatial pyramid pooling module for semantic segmentation, where the former significantly reducing the number of model operations and memory needed while maintaining the similar accuracy, and the latter capturing rich contextual information at different resolutions by pooling features and more sensitive to the position and edge information of the objects in the image. Hence, MobileNetV2 [10] / MobileNetV3 [11] were chosen as the backbone to construct our model's encoder. In addition, a Dual Atrous Spatial Pyramid Pooling segmentation head (Dual ASPP) was proposed to construct our model's decoder, which can capture and fuse semantic information of different scales to a greater extent. Finally, we demonstrate the effectiveness of the proposed model on Cityscapes data sets and attain a good performance without any post-processing.

Our main contributions are summarized as following:

- We apply lightweight CNNs (MobileNetV2 / V3) as the encoder for semantic segmentation task to compress the amounts of parameters and to reduce the calculation.
- We propose a Dual ASPP segmentation head for lightweight CNNs, which can extract and fuse the semantic information of image effectively.
- We carry out some visualization experiments to prove that the prediction accuracy of our improved lightweight semantic segmentation model has a good performance on the high-resolution Cityscapes datasets.

2. RELATED WORK

Our research relates to two aspects: lightweight CNNs and semantic segmentation algorithms.

In lightweight CNNs, SqueezeNet [12] and MobileNet [10, 11,13] are representative works, which are widely

applied in image classification and object detection. In SqueezeNet [12], the core module is Fire module, which consists of squeeze layer and expand layer. 1×1 convolution was used to convolute the input feature maps, and the model's dimension was reduced by controlling the number of output's channels. In MobileNetV1 [13], depth-wise separable convolution took the place of traditional standard convolution. It factorized a standard convolution into a depth-wise convolution followed by a point-wise convolution, and drastically reduced computation complexity. The ratio of calculation cost of depth-wise separable convolution and standard convolution can be obtained by Equation (1).

$$\frac{D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F}{D_F \cdot D_F \cdot M \cdot N \cdot D_F \cdot D_F} = \frac{1}{N} + \frac{1}{D_K^2} \quad (1)$$

where M is the number of input channels, N is the number of output channels, $D_K \cdot D_K$ is the kernel size and $D_F \cdot D_F$ is the feature map size. When we use 3×3 depth-wise separable convolution, the calculation amount could be reduced to 1/8 to 1/9 of the standard convolution. In addition, batch normalization (BN) was also used in this work. It could speed up the model training and improve the accuracy of the model. In MobileNetV2 [10], researchers used ResNet [9] for reference to propose linear bottleneck structures and inverted residual structures. These two structures could reduce the number of parameters, reduce the amount of convolution calculation, and alleviate the problem of gradient disappearance with the increase of network depth. In MobileNetV3 [11], researchers designed two models, V3-Small and V3-Large, which were suitable for low and high computing power devices respectively. They used 5×5 convolution kernels instead of 3×3 in some layers. Besides, the squeeze-and-excite modules (SE) and h-swish activation function were used to optimize the model.

In semantic segmentation algorithms, FCN [4] is the fundamental work. It replaced CNN's fully connected layer with convolution layer to achieve the original resolution output. SegNet [6] employed pooling index for nonlinear up sampling, and retained the spatial location information of features to a large extent. PSPNet [14] input the feature map into a pyramid pooling module composed of four different scales. The output results of each layer were connected with the initial global feature map through up sampling, which made full use of the global context information of features. DeepLabV3+ [7] is representative of the DeepLab series proposed by Google. This algorithm used dilated convolution to control the resolution of extracted features, so as to balance the accuracy and running time. In addition, it also proved that ASPP played a significant role in semantic segmentation.

3. PROPOSED METHOD

In this section, our proposed Dual ASPP segmentation head for lightweight CNNs will be discussed. The content is divided into two parts: the first part is to discuss the lightweight network as the model's encoder; the second part is to discuss our proposed model's structure and some details.

3.1. Lightweight CNNs as Encoders

In the previous research of image semantic segmentation, DeepLabV3+ is well known for its high efficiency. We select the lightweight CNNs (MobileNetV2 / MobileNetV3) with good performances as the backbone to construct the DeepLabV3+ model's encoder. In the original model, the encoder of DeepLabV3+ is ResNet / Xception [15], as shown in Fig. 1.

The parameter quantity and floating point operations (FLOPs) of this model are investigated when choosing ResNet101 / Xception65 / MobileNetV2 / MobileNetV3 as encoder respectively, as shown in Table 1. When MobileNetV2 / V3 is employed as the encoder, the amount of parameters and calculation of the model decreases obviously. The parameter amount of ResNet101 is about 46 times of MobileNetV3-Small, and the calculation amount is about 17.6 times.

Besides, mean intersection over union (mIOU) is used to evaluate the accuracy of the model on the Cityscapes validation set [16]. Referring to paper [7], the mIOU result for Xception65 on the Cityscapes validation set is 78.79% (without ResNet101). In the case of a significant decrease in the amount of parameters and calculations, the accuracy of MobileNetV2 / V3 as encoder is considerable, but there is still much space for improvement.

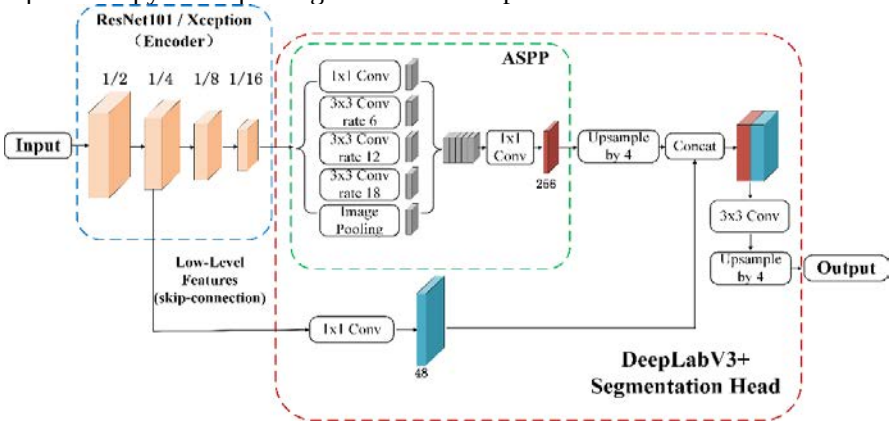


Fig. 1. The network structure of DeepLabV3+

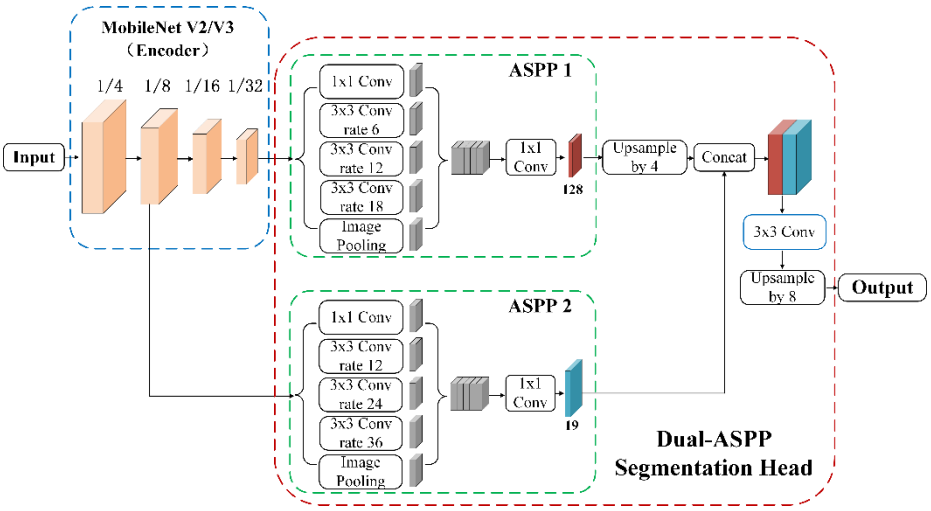


Fig. 2. The structure of Dual ASPP segmentation head.

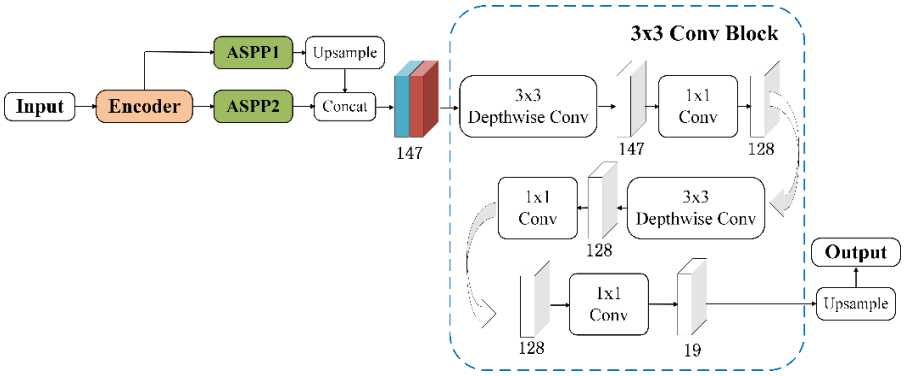


Fig. 3. The structure of 3×3 Convolution Block.

The advantages of lightweight CNNs as encoder don't only rely on its low computational complexity but also its considerable accuracy. Its disadvantages, however, are also obvious. It's a challenge for the model with few parameters to converge to a well state in the process of training. And lightweight CNNs' ability to capture and fuse semantic information is relative weak. Hence, the next step will be to optimize this ability.

Table 1. Different encoders' performances on DeepLabV3+. FLOPs are estimated for an input of 3×1024×2048.

Encoder	Params(M)	FLOPs(G)	mIOU(%)
ResNet101	47.73	458.5	-
Xception65 [7]	41.05	413.2	**78.79**
MobileNetV2	2.72	45.7	72.23
MobileNetV3-Large	2.14	35.5	66.74
MobileNetV3-Small	**1.03**	**26.1**	62.03

3.2. Dual ASPP Segmentation Head

According to the results of the previous subsection analysis, some improvements were achieved to the lightweight model for semantic segmentation. We proposed Dual ASPP structure based on the DeepLabV3+ segmentation head, as shown in Fig. 2.

In Dual ASPP, the original skip connection structure at the encoder's output of 1/4 resolution feature map is removed, since it is found that the function of this structure was not obvious for lightweight network, and it took up some computing resources in experiment (section 4). In addition, the encoder's output stride is set to 32, so that the model could subsample the input image to 1/32 of the original resolution. Then the ASPP1 module is set after 1/32 resolution output layer and the ASPP2 module is set after 1/8 resolution layer. ASPP can effectively avoid the loss of object position information in the image due to continuous pooling operations [7]. Multiple sets of ASPPs enhance the multi-scale perception of lightweight models to a greater extent.

ASPP1 and ASPP2 are different in parameter settings. In ASPP1, the atrous rate r was set to [6, 12, 18], and ASPP2 was [12, 24, 36]. Via Equation (2), we can calculate the influence of atrous rate on the receptive field of convolution kernel (where y is the receptive field size of kernel and k is convolution kernel size). When k is 3x3 and r set from 1 to 2, the receptive field of this kernel rises from 3x3 to 7x7.

$$\mathbf{y} = 2 * (\mathbf{r} - 1) * (\mathbf{k} - 1) + \mathbf{k} \qquad (2)$$

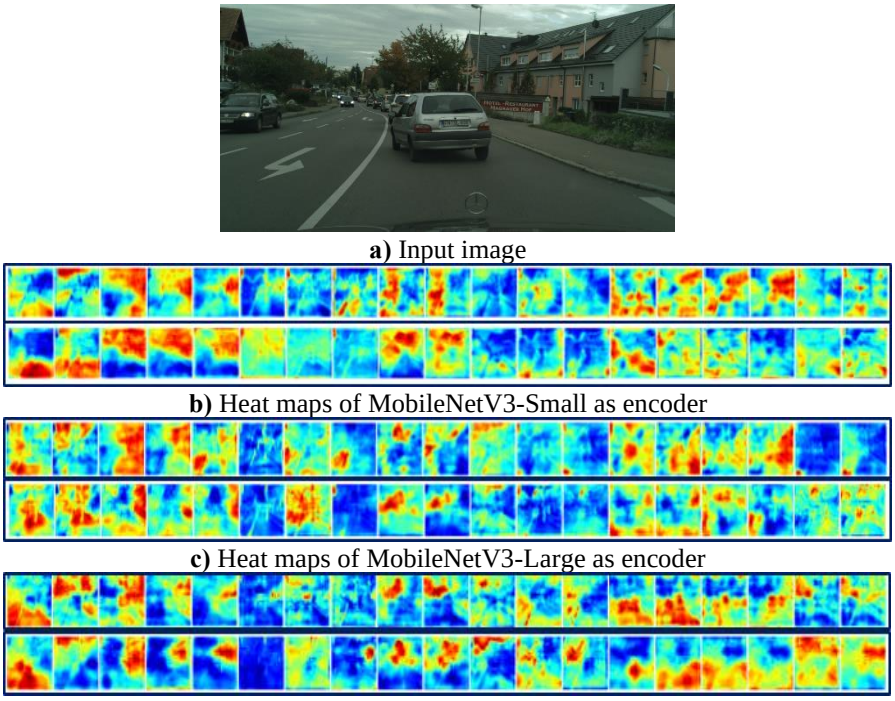
Details of the Network. In order to adapt to the output channels of MobileNetV2 / V3 model, the channel of

ASPP1's output feature map is set to 128, and ASPP2 is 19. These settings can properly enhance the learning ability of our model and effectively avoid the impact of the accuracy caused by the drastic change of channels.

The 3x3 convolution block at the end of our model is shown in Fig. 3. It is composed of 3x3 depth-wise separable convolution and 1x1 point convolution, which fully fuses the output information of ASPP1 and ASPP2, and effectively controls the parameters of the model. Since the number of output channels of ASPP1 is 128 and that of ASPP2 is 19, the number of filters of the first 3x3 depth-wise separable convolution is set to 147. After two times of 3x3 depth convolution and three times of 1x1 convolution, the final output channel number of this block is 19, which is in line with the number of categories of the Cityscapes dataset we selected.

Table 2. Comparison of DL and DA on MobileNetV2/V3.

Encoder	DL's Head	DA's Head	FLOPs(G)	mIOU(%)
MobileNetV2	–	–	20.9	68.24
MobileNetV2	√	–	45.6	72.23
MobileNetV2	–	√	**14.2**	**73.38**
MobileNetV3-Large	–	–	11.8	61.98
MobileNetV3-Large	√	–	35.5	66.74
MobileNetV3-Large	–	√	**10.8**	**67.69**
MobileNetV3-Small	–	–	**3.2**	58.06
MobileNetV3-Small	√	–	26.1	62.03
MobileNetV3-Small	–	√	4.2	**64.23**



a) Input image

b) Heat maps of MobileNetV3-Small as encoder

c) Heat maps of MobileNetV3-Large as encoder

d) Heat maps of MobileNetV2 as encoder

Fig. 4. Comparison of output heat maps of different encoders and different segmentation heads, DL (upside) and DA (downside).

4. EXPERIMENTS

In this section, some comparative experiments are conducted to evaluate the Dual ASPP segmentation head proposed in section 3. The proposed model is built, trained and validated on PyTorch, which is one of the most common network frameworks at present.

4.1. Datasets

In this work, the Cityscapes dataset [16] is used as the experimental basis. It is a high-resolution (1024×2048) dataset containing high quality pixel-level annotations of 5000 images and about 20000 coarsely annotated images. Each image in this dataset takes a street view from 18 European cities. It contains 30 individual classes, which are generally classified into 19 categories (road, sidewalk,

building, wall, fence, pole, traffic light, traffic sign, vegetation, terrain, sky, person, rider, car, truck, bus, train, motorcycle, bicycle.).

Training set (1525 images) is used to train our model and validation set (500 images) to test the trained model. In the process of training, we do not adopt data augmentation measures such as random image flipping to ensure that the performance of the model is only related to our improvements. In addition, no post-processing, such as conditional random field (CRF), was used for the output.

4.2. Evaluation Criterion

In the evaluation criteria of the model, we mainly consider the accuracy, inference speed and computational complexity. So the following three indicators were chosen to evaluate the model: mIOU, FPS and FLOPs.

Intersection over union (IOU) is the ratio of the overlapping area size of labels and prediction results to their total area size that is $IOU = \frac{TP}{TP+TN+FN}$, where TP is true positive, TN is true negative, FN is false negative. Mean IOU (mIOU) is the average value of IOU of all categories in the image that is:

$$mIOU = \frac{1}{N}\sum_{i=1}^{N} IOU_i \quad (3)$$

where N is the number of all categories.

Frames per second (FPS) represents the number of images (frames) processed by the model per second. It is a direct reflection of the speed of model inference. In the experiment, the hardware is a single GPU (GeForce RTX 2080 Ti).

Floating point operations (FLOPs) represents the number of addition and multiplication calculations of the model. It can be used to yard the computational complexity of the algorithm / model.

Table 3. Comparison with the existing approaches in terms of segmentation accuracy and inference speed.

Method	mIOU(%)	Time(ms)	FPS	Params(M)
ENet [17]	58.3	**13.0**	**77**	**0.36**
ESPNet [18]	60.3	18.5	54	0.40
CGNet [19]	64.8	20.0	50	0.50
ICNet [20]	69.5	33.3	30	7.80
ERFNet [21]	69.7	20.8	48	2.10
ESNet [1]	70.7	15.9	63	1.66
MobileNetV2+DA	**73.4**	16.4	61	2.16

Table 4. Impacts of skip connection structure.

Encoder	DA's Head	Skip connection	FLOPs(G)	mIOU(%)
MobileNetV3-Small	√	√	8.6	63.94
MobileNetV3-Small	√	–	**4.2**	**64.23**

4.3. Comparative Experiment and Discussion

DeepLabV3+ segmentation head and Dual ASPP segmentation head were taken as the decoder of MobileNetV2/V3 respectively and we train these models under the same training conditions. The results are shown in Table 2, where DL's Head is DeepLabV3+ segmentation head and DA's Head is Dual-ASPP segmentation head. If no segmentation head is used, the output of the encoder (output stride=16) is directly followed by a 3 x 3 convolution block and an up-sample block.

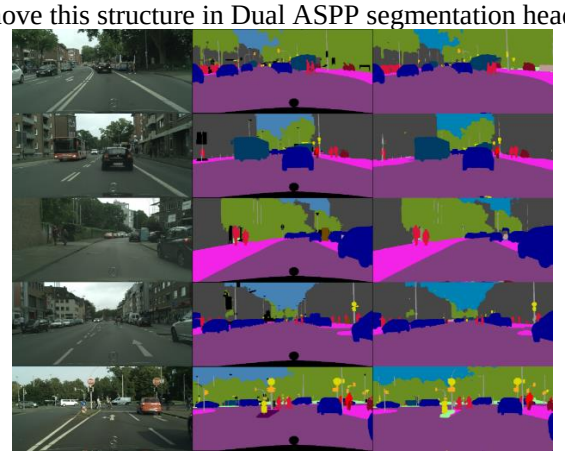
The results in Table 2 show that Dual ASPP segmentation head performs well. The mIOU is improved from 72.23% to 73.38% / 66.74% to 67.69% / 62.03% to 64.23% when using V2 / V3-Large / V3-Small as encoder respectively. The highest accuracy improvement is achieved by Dual ASPP on V3-Small, reaching 2.2%. In addition, the results in Table 2 also show that Dual ASPP can greatly reduce the computational complexity of the model. It means that Dual ASPP is suitable for lightweight models. We visualized the three models' last layer (19 channels) output heat maps under DL's head and DA's head respectively, as shown in Fig. 4. The comparison heat maps show that the model using DA has a more delicate perception of objects in the image than DL, which means that our proposed structure can extract semantic information more effectively.

We extract the results of V2 from the validation datasets and compare them with the original images and ground truth. As shown in Fig. 5, the prediction results of V2 with Dual ASPP are quite close to ground truth.

Furthermore, we compare the lightweight semantic segmentation networks (parameters' amount is less than 10M) proposed in recent years, and the results are shown in Table 3. Our model is tested on a single GeForce RTX 2080 Ti GPU, the test conditions of other models are similar to ours (such as Nvidia TitanX GPU). Our

model's inference speed (61FPS) is slower than ENet (77FPS) [17] and ESNet (63FPS) [1], but the accuracy performance is the best (73.4%), which is 2.7% higher than the second ESNet (70.7%).

In Section 3, we mentioned removing the original skip connection structure of DL's head. We choose MobileNetV3-Small as the encoder to study the impact of the original skip connection structure, and the results are shown in Table 4. With using Dual ASPP, skip connection does not improve the accuracy of the model, and consumes more computing resources. Hence, we remove this structure in Dual ASPP segmentation head.



a) Input images **b)** Ground truth **c)** Prediction results

Fig. 5. The prediction results of MobileNetV2 with Dual ASPP on Cityscapes validation set.

5. CONCLUSION

In this paper, we explored the application of lightweight CNNs for semantic segmentation. MobileNetV2 / MobileNetV3 were chosen as the backbone to construct our model's encoder. By this method, the parameters and computational complexity of the model were effectively controlled. In addition, a Dual Atrous Spatial Pyramid Pooling segmentation head (Dual ASPP) was proposed to construct our model's decoder, which can capture and fuse semantic information of different scales to a greater extent. The computation amounts are greatly reduced by using lightweight CNNs. In the process of research, however, we found that it was a challenge for the lightweight model to converge to a better state in training. We would like to explore and solve this problem to improve the performance of the model in the future work.

Acknowledgements

This work was partly supported by Program for Changjiang Scholars and Innovative Research Team in University (IRT-16R06, T2014224), National Natural Science Foundation of China (Grant No. 61903034, 61973034, U1913203 and 91120003).

REFERENCES:

[1] Wang, Y., Zhou, Q., Xiong, J., Wu, X. F., Jin, X.: ESNet: An Efficient Symmetric Network for Real-Time Semantic Segmentation. In: Conference on Pattern Recognition and Computer Vision (2019). Springer, Cham, 41-52.

[2] Shi, J., Malik, J.: Normalized cuts and image segmentation. In: IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8), 888-905(2000).

[3] Rother, C., Kolmogorov, V., Blake, A.: GrabCut: Interactive foreground extraction using iterated graph cuts. In: ACM Transactions on Graphics (2004). Association for Computing Machinery, 309-314.

[4] Long, J., Shelhamer, E., Darrell, T.: Fully Convolutional Networks for Semantic Segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (2015). IEEE Computer Society, 3431-3440.

[5] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional Networks for Biomedical Image Segmentation. In: International Conference on Medical image computing and computer-assisted intervention (2015). Springer, Cham, 234-241.

[6] Badrinarayanan, V., Kendall, A., Cipolla, R.: SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. In: IEEE Transactions on Pattern Analysis and Machine Intelligence (2017). 2481-2495.

[7] Chen, L. C., Zhu, Y., Papandreou, G., et al.: Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In: European Conference on Computer Vision (2018). Springer, Cham, 833-851.

[8] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference on Learning Representations (2015).

[9] He K, Zhang X, Ren S, et al.: Deep Residual Learning for Image Recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (2016). IEEE Computer Society, 770-778.

[10] Sandler, M., Howard, A., Zhu, M., et al.: Mobilenetv2: Inverted Residuals and Linear Bottlenecks. In: IEEE Conference on Computer Vision and Pattern Recognition (2018). IEEE Computer Society, 4510-4520.

[11] Howard, A., Ruoming Pang, Adam, H., et al.: Searching for MobileNetV3. In: International Conference on Computer Vision (2019). IEEE Computer Society, 1314-1324.

[12] Iandola, F. N., Han, S., Moskewicz, M. W., et al.: SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. In: arXiv preprint arXiv: 1602.07360. (2016).

[13] Howard, A. G., Zhu, M., Chen, B., et al.: Mobilenets: Efficient Convolutional Neural Networks for Mobile Vision Applications. In: arXiv preprint arXiv: 1704.04861.

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

[15] Chollet, F.: Xception: Deep Learning with Depthwise Separable Convolutions. In: IEEE Conference on Computer Vision and Pattern Recognition (2017). IEEE Computer Society, 1800-1807

[16] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The Cityscapes Dataset for Semantic Urban Scene Understanding. In: IEEE Conference on Computer Vision and Pattern Recognition (2016). IEEE Computer Society, 3213-3223.

[17] Paszke, A., Chaurasia, A., Kim, S., Culurciello, E.: Enet: A deep neural network architecture for real-time semantic segmentation. In: arXiv preprint arXiv: 1606.02147. (2016).

[18] Mehta, S., Rastegari, M., Caspi, A., Shapiro, L., Hajishirzi, H.: ESPNet: Efficient spatial pyramid of dilated convolutions for semantic segmentation. In: 15th European Conference on Computer Vision (2018). Springer Verlag, 561-580.

[19] Wu, T.Y., Tang, S., Zhang, R., Zhang, Y.D.: Cgnet: A light-weight context guided network for semantic segmentation. In: arXiv preprint arXiv: 1811.08201v1. (2018).

[20] Zhao, H.S., Qi, X.J., Shen, X.Y., Shi, J.P., Jia, J.Y.: ICNet for real-time semantic segmentation on high-resolution images. In: 15th European Conference on Computer Vision (2018). Springer Verlag, 418-434.

[21] Romera, E., Alvarez, J.M., Bergasa, L.M., Arroyo, R.: ERFNet: Efficient residual factorized convnet for real-time semantic segmentation. In: IEEE Transactions on Intelligent Transportation Systems (2018). 263–272.