Learning More Accurate Features for Semantic Segmentation in CycleNet

Linzi Qu, Lihuo He(✉), Junji Ke, Xinbo Gao, and Wen Lu

School of Electronic Engineering, Xidian University, Xi'an, China
lhhe@mail.xidian.edu.cn

Abstract. Contextual information is essential for computer vision tasks, especially semantic segmentation. Previous works generally focus on how to collect contextual information by enlarging the size of receptive field, such as PSPNet, DenseASPP. In contrast to previous works, this paper proposes a new network – CycleNet, which considers assigning a more accurate representative for every pixel. It consists of two modules, Cycle Atrous Spatial Pyramid Pooling (CycleASPP) and Alignment with Deformable Convolution (ADC). The former realizes dense connections between a series of atrous convolution layers with different dilation rates. Not only the forward connections can aggregate more contextual information, but also the backward connections can pay more attention to important information by transferring high-level features to low-level layers. Besides, ADC generates accurate information during the decoding process. It draws support from deformable convolution to select and recombine features from different blocks, thus improving the misalignment issues caused by simple interpolation. A set of experiments have been conducted on Cityscapes and ADE20K to demonstrate the effectiveness of CycleNet. In particular, our model achieved 46.14% mIoU on ADE20K validation set.

1 Introduction

Semantic segmentation is a great challenge in dense image classification where the resolution of output labels is the same as that of the input images. Each pixel in the image needs a semantic label. This task has been widely used in video surveillance, automotive driving, medical image processing and other fields. Traditional segmentation methods aim to extract handicraft features of image regions which is not only complicated, but also lead to inaccurate results.

With the development of deep learning, especially Convolution Neural Networks (CNN), a landmark framework – Fully Convolutional Networks (FCN) has emerged in the field of semantic segmentation. Based on it, most of the subsequent works train model end to end to obtain representative image features automatically. FCNs use pooling layers to expand receptive fields and further achieve high-level information. However, these methods ignore the negative impact of down-sampling on the resolution, which is crucial for semantic segmentation.

H. Ishikawa et al. (Eds.): ACCV 2020, LNCS 12622, pp. 571–584, 2021.
https://doi.org/10.1007/978-3-030-69525-5_34

In order to obtain larger receptive fields and richer contextual information, recent works mainly rely on atrous convolution [1] or attention mechanism. Deeplab [2] and DenseASPP [3] concatenated features from a cascade of atrous convolution layers with different dilation rates. PSPNet [4] proposed pyramid pooling module to aggregate information from multi-scale features after pooling layers. However, a neglected issue in these works is whether a large receptive field is equally important for every pixel in the image. For example, a pixel in a semantic object requires a larger receptive field to see the entire object, but when a pixel approaches the boundary, a larger receptive field may bring more information about other categories, leading to incorrect segmentation. At the same time, the attention-based methods are designed to capture long-range context without being limited by the fixed size of convolution kernel. But it's time-consuming because more useful information mainly locates around the pixels, meaning that numbers of computation is unnecessary. In addition, in the process of obtaining high-level information, the size of models' output like [3,5] is 1/8 of the input size, and then interpolated to the same size of input. Simple methods of restoring resolution can lead to misalignment issue.

In this paper, an elaborate CycleNet is proposed to provide precise features for each pixel, on the premise of adequate receptive fields. CycleNet is mainly composed of two sub-modules CycleASPP and ADC. CycleNet is a DenseASPP-like method. They all consist of a backbone to encode features followed by a series of atrous convolution layers. The difference is that there are both forward and backward connections between any atrous convolution layers in CycleASPP, but DenseASPP only has forward connections. To be specific, the first time of an atrous convolution begins with the concatenation of all the previous layers' output, just like DenseASPP, to successively produce multi-scale features. Inspired by CliqueNet [6], the feedback mechanism is able to enhance the representation of models. Thus, CycleASPP applies backward connections to refine features. After the first time, outputs of update layers then are concatenated to be inputs of the previous layers, as illustrated in Fig. 1. By backward connections, the high-level information is fed back to previous layers. Benefits from this, CycleASPP not only refines the filters, but also produces more accurate features. Moreover, an ADC module is proposed to prevent the loss of accurate information caused by down-sampling. Deformable convolution layers are used to learn the positional correspondence between different resolution features.

Our main contributions are summarized as follows:

1. We introduce CycleASPP, which continuously refines the representativeness of asrous convolution layers through feedback mechanism.
2. ADC module is proposed to compensate for the misalignment issue caused by down-sampling.
3. The visualization between different parts of CycleASPP shows the backward connections can refine filters.
4. We verify CycleNet on two semantic segmentation benchmark datasets, Cityscapes [7] and ADE20K [8]. The experiments show that our model

achieves the state-of-the-art results including 82.0% mIoU on Cityscapes test set and 46.14% mIoU on ADE20K validation set.

2 Related Work

2.1 Context Model in Semantic Segmentation

Recent studies have shown that semantic segmentation benefits from rich contextual information. Although the emergence of FCN has made some progress in semantic segmentation, it can not produce enough contextual information by a single receptive field. PSPNet [4] designed a spatial pyramid pooling model to collect contextual information from different pooling layers. ASPP [2] utilized atrous convolutions to enlarge receptive fields thus further fusing different contextual information. Inspired by DenseNet [9], DenseASPP [3] added dense connections between a cascade of atrous convolution layers to capture multi-scale context. Some other works focused on attention-based methods. Contextual information in DANet [10] is collected by calculating the similarity between each pixel in image. To improve efficiency, CCNet [5] adopted criss-cross attention module which only computing pixels on the criss-cross path.

2.2 Recurrent Neural Network

Recurrent neural networks, such as LSTM [11] and GRU [12], which benefited from feature-usage and iterative learning, are mainly used for sequential tasks, especially natural language processing (NLP). In image classification tasks, to simulate feedback loops in human brain, I. Caswell [13] proposed loopy neural networks that allow the information flow from deeper layers to lower layers, CliqueNet [6] incorporated forwards and backwards connections between every layers in a block to maximize the information flow and realize spatial attention. RNN-like model also improved the ability of long-dependencies between pixels in semantic segmentation. Like, ReSeg [14] proposed a recurrent layer containing four RNN, which first horizontally computed the image patches, and then vertically computed the output of the hidden states, so as to efficiently collect contextual information.

2.3 Multi-level Features Fusion

Encoder-decoder structures are presented to balance the high-level semantic features with high resolution. Common methods are to add or concatenate low-level features with high-level features after interpolated. GFF [15] is inspired by the gate mechanism of LSTM to assign different weights to different features according to their validity, because multi-level features are not equally important to the results. Considering the misalignment of different layers, enlightened by the optical flow, Alignseg [16] proposed a learnable interpolation method to precisely align high and low level features. Different from other works, [17] firstly

down-sampling the low-level features to the same size as the high-level features, and then aggregated all the features at the low resolution. Finally, a data-based DUpsampling method is designed to reproduce the original size.

3 Method

3.1 Cycle Atrous Spatial Pyramid Pooling (CycleASPP)

DenseASPP. The purpose of atrous convolution layers is to balance the problem of large receptive fields and high resolution in semantic segmentation. It can be represented as follows:

$$Y[k] = \sum_{i=1}^{I} X[k + r \cdot i] \cdot w[i] \tag{1}$$

where $Y[k]$ is the output features, $X[k]$ is the input features, $w[i]$ is a parameter of convolution filter, and r is the dilation rate, and I denotes the filter size. We adopt $f_r(X)$ to represent atrous convolution to simplify symbolization.

Since the features generated by the simple atrous convolutional layer are difficult to cover a scale range, DenseASPP adopted atrous convolution layers with different rates, which not only realizes lager receptive fields, but also produces dense scale-range features. However, a lager receptive field is unable to benefit all the pixels in image, especially those near the boundaries. A larger receptive field means more information from adjacent objects, which sometimes confuse the model. Inspired by CliqueNet [6], we added the backward connections between every atrous convolution layer to ensure that each pixel is able to focus on its own accurate features, on the premise that it receives a sufficient receptive field.

CycleASPP. CycleASPP is a DenseASPP-like module that contains a series of sequential atrous convolution layers with increasing atrous rates. In particular, there are bidirectional connections in CycleASPP, whereas DenseASPP only has forward connections.

As depicted in Fig. 1, CycleASPP consists of two parts. In the Part I, input features are concatenated with output of previous layers, and then all the features are utilized to update the next layer. In part I, each atrous layer can be defined as:

$$Y_j^0 = f_{r_j}(concate[Y_{j-1}^0, Y_{j-2}^0, \cdots, Y_0^0]) \tag{2}$$

where Y^0 is the output of atrous convolution in part I, r_j represents the $j-th$ atrous convolution layers in CycleASPP, and $concate[\cdots]$ is concatenation operation.

After that, feedback features from Part I is used to refine the atrous convolution layers. In the part II, all the features from the Part I are concatenated as input except for the output of current layer. What is more noteworthy is that the

atrous convolution layers are updated sequentially, so some of aggregate features are from Part I and others are from Part II, which can be formulated as:

$$Y^k_{j,j\neq 1} = f_{r_j}(concate[Y^{k-1}_J, Y^{k-1}_{J-1}, \cdots, Y^{k-1}_{j+1}, Y^k_{j-1}, \cdots, Y^k_1]) \tag{3}$$

where k denotes feedback times in Part II. $k = 0$ represents only forward connections.

In CycleASPP, the latest outputs from each atrous convolution layer are used together to generate the final feature maps. This recurrent structure has two main benefits: the first is to refine the convolution filters to attain more accurate representative features, and the second is to maximize information flow.

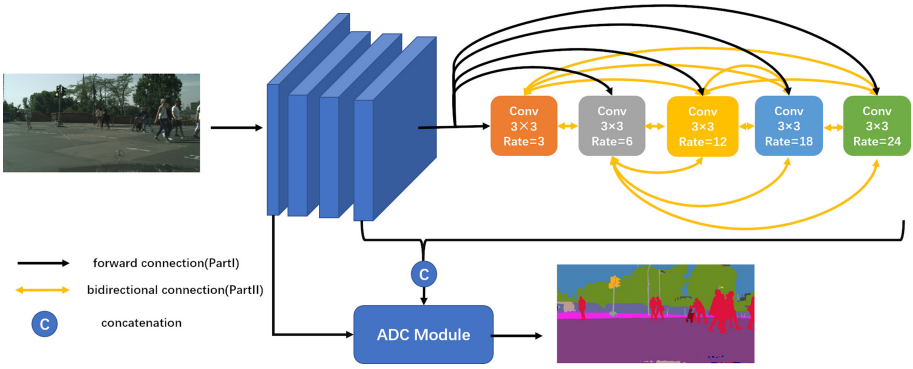

Fig. 1. Overview of CycleNet. Given an input image, we use a CNN model to generate high-level features. Then, CycleASPP including a series of atrous convolution layers with different rates is used to learn rich and accurate contextual features. The outputs of CycleASPP are concatenated with low-level features. To align multi-scale features, we proposed Alignment with Deformable Convolution (ADC) module.

3.2 Alignment with Deformable Convolution (ADC)

Restoration of image resolution caused by pooling layers is an inevitable procedure of semantic segmentation. At the decoding stage, the low-resolution feature maps firstly are interpolated to the same size of high-resolution ones, and then a simple concatenating way results in spatial misalignment. Considering that the deformable convolution layers are able to automatically learn the position offset which enhance the different features fusion. We exploit modulated deformable module [18] and it is obtained by:

$$Y[l_0] = \sum_{n=1}^{N} w_n \cdot X[l_0 + l_n + \triangle l_n] \cdot \triangle m_n \tag{4}$$

N is sampling positions, w_n and l_n respectively denote the weight and the predefined offsets for l_0. Supposing a 3×3 deformable convolution, N is 9 and

$l_n \in [(-1,-1),(-1,0),\cdots,(1,1)]$. Besides, the offset $\triangle l_n$ and the modulation scalar $\triangle m_n$ are based on data.

In this module, we firstly perform bilinear interpolation on the features X_{high} generated by CycleASPP to the same size with low-level features X_{low}. Then, these features are combined, followed by several convolutions to generate the learnable offset $\triangle l_n$ and modulation scalar $\triangle m_n$ which are required by the modulated deformable module. Finally, our aligned context features $Y_{context}$ could be defined as follow:

$$Y_{context} = f(concate[X_{low}, \widetilde{X}_{high}]) \tag{5}$$

where $\widetilde{X}_{high}$ denotes the aligned features by deformable convolution, and f is conventional convolution layers (Fig. 2).

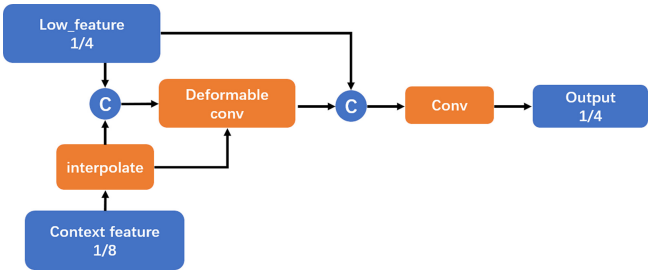

Fig. 2. An elaborate show of Alignment with Deformable Convolution (ADC) module. It takes two parts features respectively from Backbone and CycleASPP, and then exploit deformable convolution to align these features.

4 Experimental Evaluation

A large number of experiments are conducted to evaluate the effectiveness of CycleNet on two benchmark datasets Cityscapes [7] and ADE20K [8]. Results are evaluated with mean of class-wise Intersection over Union (mIoU) and pixel accuracy.

4.1 Implementation Details and Datasets

Network Structure. Our method adopts ResNet101 [19] pre-trained on ImageNet [20]. The last two pooling layers and the fully-connected layers of model are removed. At the same time, the convolution layers in the last two blocks are replaced by atrous convolution layers with atrous rates of 2 and 4, respectively.

Training Settings. We train our model with stochastic gradient descent training method. The initial learning rate is initialized as 1e−2 for Cityscapes and 2e−2 for ADE20K. The momentum and weight decay are set as 0.9 and 1e−4, respectively. According to the prior work [5,10], we utilize a poly learning rate policy where the initial learning rate is multiplied by 0.9. Synchronized Batch Normalization [21] is employed to synchronize the mean and standard variation. For Cityscapes (only use 2975 finely annotated images), we train the model with 8 mini-batch size and 180 epochs. The input is randomly cropped to 796796 from the original image. For ADE20K, we train the model with 16 mini-batch size, 120 epochs and the input is cropped to 512512. During training, data augmentation including random horizontal flipping, random cropping, and random scaling in the range of [0.75, 2] are used to avoid overfitting. As for loss, we adopt the auxiliary supervision, as [22,23].

Cityscapes. The Cityscapes dataset is designed for understanding of urban street scenes. It contains 5000 images with finely annotations and 20000 images with coarse annotations. The annotations include 30 categories such as road, tree and person. Only 19 categories are commonly used for training and evaluation. The 5000 finely annotated images are split into 2975 images for training, 500 images for validation and 1525 images for testing.

ADE20K. ADE20K is a complex scene parsing dataset including 150 categories involved objects and stuff. It contains 25000 images which consist of 20000 training images, 2000 validation images and 3000 testing images.

4.2 Experiments on Cityscapes

Ablation Study. To evaluate the effectiveness of proposed model, we implement ablation experiments on Cityscapes validation set. We choose atrous ResNet101 mentioned above as the baseline network which down-samples the input size to 1/8 of its original size. The baseline model reaches 76.25% mIoU. All components are based on baseline model and continuously improve the performance.

Effectiveness of CycleASPP. All evaluation of CycleASPP are equal without ADC module. There are two important components in CycleASPP, one is atrous convolution layers and the other is recurrent connections. First, we compare several methods of atrous convolution layers with different numbers at different dilation rates, as shown in Table 1. It is obvious that adding more atrous convolution layers and increasing dilation rates can both improve the performance, because the model achieves larger receptive fields and gains more contextual information. As DenseASPP [3] demonstrates when the receptive field goes larger than the feature map, the results begin to decrease. Thus, there is no need to add more convolutions. For subsequent evaluation of the recurrent connections, which is proposed to refine filters. In order to be fair, we compare

Table 1. The contrast experiments between ASPP, DenseASPP and CycleASPP with various atrous dilation rates.

Method	Backbone	mIoU(%)	GFLOPs
ASPP (6, 12, 18)	ResNet101	78.18	–
DensASPP (3, 6, 12, 18, 24)	ResNet101	78.45	539.1
DenseASPP (3, 6, 12, 18, 24, 30)	ResNet101	78.57	555.5
CycleASPP (6, 12, 18, 24)	ResNet101	78.54	530.8
CycleASPP (3, 6, 12, 18, 24)	ResNet101	78.93	551.5
CycleASPP (3, 6, 12, 18, 24, 30)	ResNet101	78.95	574.0

the results from DenseASPP and CycleASPP with same dilation rates. The performance in Table 1 shows that segmentation results with same receptive fields from CycleASPP outperform these from DenseASPP. CycleASPP (6, 12, 18, 24) achieves the almost same result as DenseASPP (3, 6, 12, 18, 24, 30) while at low GFLOPs. In other words, CycleASPP improves the accuracy of our method without much loss of speed. To ensure the follow-up experiments' performance, we utilize atrous convolution layers with dilation rates (3, 6, 12, 18, 24, 30) and only once feedback connection for the further experiments. As is illustrated in Table 2, the CycleASPP module brings 2.70% mIoU improvements compared with baseline, proving the effectiveness of the introduced module.

Effectiveness of ADC. The use of ADC module to retrieve location information from high-level features missing from the pooling layers has been detailed in Sect. 3.2. We select the low-level features from block1 of atrous ResNet101, which are 1/4 the size of the input image. Compared to the previous models in Table 2, the performance gains 1.60% mIoU improvement when adding this part.

Effectiveness of extra trick. In order to boost the results, we also incorporate a trick used in many works, like [22,23]. Multi-scale inference (MS): this trick is only used in inference. The final segmentation results are obtained by averaging the output probability maps at different scales which vary between [0.75, 1, 1.25, 1.5, 1.75]. From Table 2, we can see that MS brings 0.69% mIoU improvements.

Table 2. Ablation experiments on Cityscapes validation dataset. We evaluated the improvements of each proposed model.

ResNet101	CycleASPP	ADC	MS	mIoU(%)
✓	–	–	–	76.25
✓	✓	–	–	78.95
✓	✓	✓	–	80.55
✓	✓	✓	✓	81.26

Compare with State of the Arts. We compare our CycleNet with previous state-of-arts works such as DenseASPP [3], CCNet [5], DANet [10] and so on Cityscapes test set in Table 3. For fair comparison, we only train the model with fine annotated dataset and evaluate the results by the evaluation server. The CycleNet consists of CycleASPP which set the dilation rates as (3, 6, 12, 18, 24, 30), only once feedback connection and ADC module. Then, we boost the performance by MS. Finally, our approach achieves 82.0% mIoU which outperforms DANet 0.5% mIoU.

Table 3. Results on Cityscapes test dataset.

Method	Backbone	mIoU(%)
RefineNet [24]	ResNet101	73.6
PSPNet [4]	ResNet101	78.4
BiSeNet [25]	ResNet101	78.9
DSSPN [26]	ResNet101	77.8
PSANet [22]	ResNet101	80.1
DenseASPP [3]	DenseNet161	80.6
CCNet [5]	ResNet101	81.4
DANet [10]	ResNet101	81.5
CycleNet (ours)	ResNet101	82.0

4.3 Experiments on ADE20K

Compare with State of the Arts. Here, we further experiment with our method on ADE20K. As shown in Table 4, we compare our work with PSPNet [4], EncNet [21], DSSPN [26], PSANet [22], CCNet [5] and SPNet [23] on the ADE20K validation set. We also adopt atrous ResNet101 as our backbone, and the dilation rates of CycleASPP are set as (3, 6, 12, 18, 24) because of the small input size. Both 46.14% mIoU and 82.20% pixel accuracy are achieves state-of-art results.

5 Visualization

The CycleNet mainly benefits from two modules mentioned above, CycleASPP and ADC. To further analyze the reason, we visualize features'similarity maps and class activation maps to realize a clear understanding.

Table 4. Results on ADE20K validation dataset.

Method	Backbone	mIoU(%)	Pixel Acc.(%)
PSPNet [4]	ResNet101	43.29	81.39
EncNet [21]	ResNet101	44.65	81.69
DSSPN [26]	ResNet101	43.68	81.13
PSANet [22]	ResNet101	43.77	81.51
CCNet [5]	ResNet101	45.22	–
SPNet [23]	ResNet101	45.60	82.09
DenseASPP [3]	ResNet101	43.03	80.73
CycleNet (ours)	ResNet101	46.14	82.20

5.1 Results of Two Datasets

We visualize some results under different settings of the proposed approach in Fig. 3. The red square show the more difficult to distinguish regions. Obviously to find, CycleASPP can correct the misclassified pixels in DensASPP. Such as 'truck' or 'car' in the second example and 'building' or 'wall' in third example. Then, as in first example, CycleASPP can make sure that the pixels on the

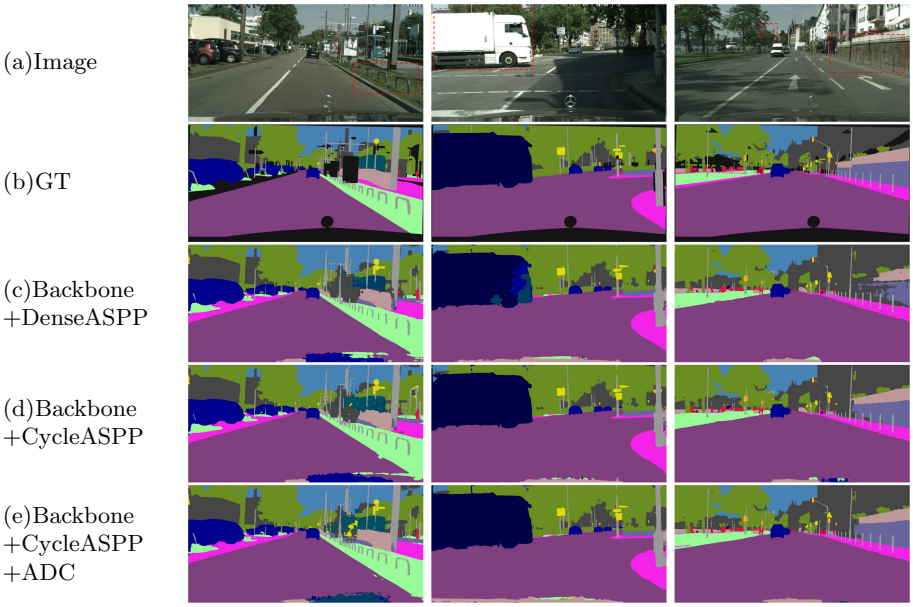

Fig. 3. Visualization of segmentation results among different approaches on Cityscapes validation sets. The first to the fifth rows respectively are original image, ground truth results, results from Backbone+DenseASPP, results from Backbone+CycleASPP, results from Backbone+CycleASPP+ADC.

edge are not disturbed by the rich information from the large receptive fields. From the third example, compared the segmentation results of 'pole' from different methods, obviously, Backbone+CycleASPP equipped with ADC is able to exactly segment tiny objects.

5.2 Features in Different Parts

In CycleASPP, each layer benefits from backwards high-level information. In order to show the refinement more clearly, we calculated the similarity maps, as shown in Fig. 4(b) and (c). With the help of backward connections, features of the same objects are more similar and features of different objects are more discriminative. Grad-CAM [27] are used to visualize the activation maps of two

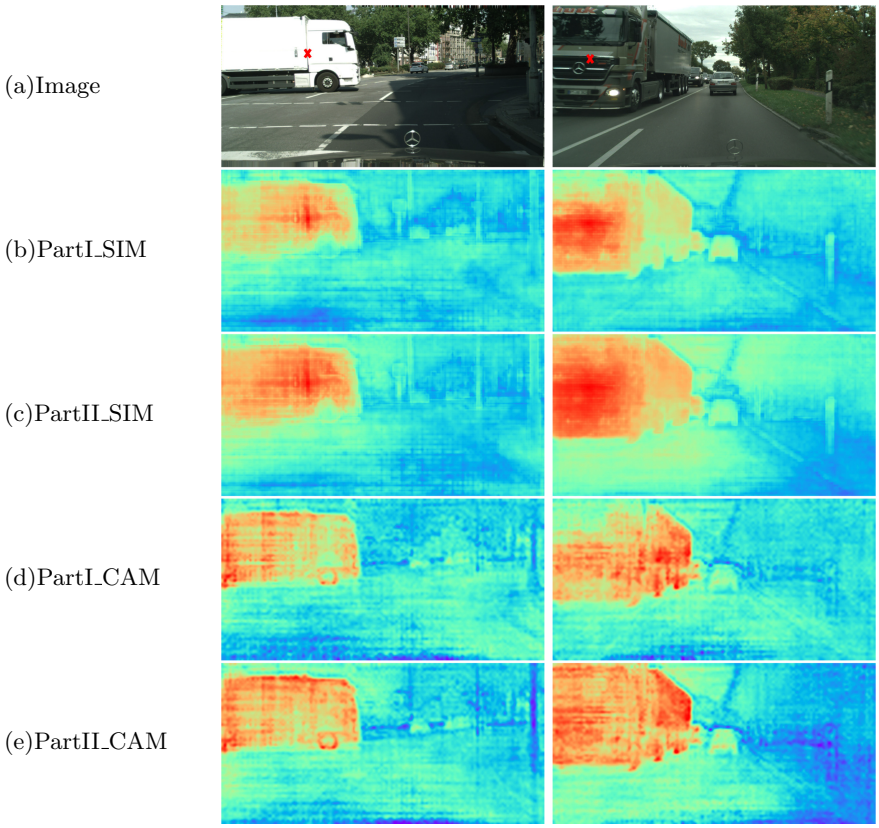

Fig. 4. Visualization of features. (a) is the original image. (b) and (c) are similarity maps with red pixel in original images. Hotter color denotes more similar in feature level. (d) and (e) are class activation maps. Hotter color means larger in the degree of activation.

parts. As shown in Fig. 4(d) and (e), we only use one pixel exactly as red symbol in Fig. 4(a) to produce a class activation map. It is obvious that our module can see targets.

6 Conclusion

In this work, we present CycleNet to deal with the semantic segmentation task in complex scene. CycleNet contains two significant parts, CycleASPP and ADC. CycleASPP adds recurrent connections to dense forward connections like DenseASPP that help model gain more accurate information. Since deformable convolution can collect the information from unfixed positions, ADC develops the decoding procedure that is different from the simple interpolation. As a result, the possibility maps can better aligned with input image. Experiments on Cityscapes and ADE20K demonstrate the effectiveness of the proposed approach.

Acknowledgements. This research was supported in part by the National Key Research and Development Program of China (Grant No. 2018AAA0102702), the National Natural Science Foundation of China (Grant Nos. 61876146, 62036007, 61871311).

References

1. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122 (2015)
2. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 (2017)
3. Yang, M., Yu, K., Zhang, C., Li, Z., Yang, K.: DenseASPP for semantic segmentation in street scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3684–3692 (2018)
4. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2881–2890 (2017)
5. Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., Liu, W.: CCNet: criss-cross attention for semantic segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 603–612 (2019)
6. Yang, Y., Zhong, Z., Shen, T., Lin, Z.: Convolutional neural networks with alternately updated clique. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2413–2422 (2018)
7. Cordts, M., et al.: The Cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3213–3223 (2016)
8. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ADE20K dataset. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 633–641 (2017)
9. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708 (2017)

10. Fu, J., et al.: Dual attention network for scene segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3146–3154 (2019)
11. Zaremba, W., Sutskever, I., Vinyals, O.: Recurrent neural network regularization. arXiv preprint arXiv:1409.2329 (2014)
12. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014)
13. Caswell, I., Shen, C., Wang, L.: Loopy neural nets: imitating feedback loops in the human brain. Technical report (2016)
14. Visin, F., et al.: ReSeg: a recurrent neural network-based model for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 41–48 (2016)
15. Li, X., Zhao, H., Han, L., Tong, Y., Yang, K.: GFF: gated fully fusion for semantic segmentation. arXiv preprint arXiv:1904.01803 (2019)
16. Huang, Z., Wei, Y., Wang, X., Shi, H., Liu, W., Huang, T.S.: AlignSeg: feature-aligned segmentation networks. arXiv preprint arXiv:2003.00872 (2020)
17. Tian, Z., He, T., Shen, C., Yan, Y.: Decoders matter for semantic segmentation: data-dependent decoding enables flexible feature aggregation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3126–3135 (2019)
18. Zhu, X., Hu, H., Lin, S., Dai, J.: Deformable ConvNets V2: more deformable, better results. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9308–9316 (2019)
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
20. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. IEEE (2009)
21. Zhang, H., et al.: Context encoding for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7151–7160 (2018)
22. Zhao, H., et al.: PSANet: point-wise spatial attention network for scene parsing. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11213, pp. 270–286. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01240-3_17
23. Hou, Q., Zhang, L., Cheng, M.M., Feng, J.: Strip pooling: rethinking spatial pooling for scene parsing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4003–4012 (2020)
24. Lin, G., Milan, A., Shen, C., Reid, I.: RefineNet: multi-path refinement networks for high-resolution semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1925–1934 (2017)
25. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: BiSeNet: bilateral segmentation network for real-time semantic segmentation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11217, pp. 334–349. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01261-8_20

26. Liang, X., Zhou, H., Xing, E.: Dynamic-structured semantic propagation network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 752–761 (2018)
27. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-CAM: visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 618–626 (2017)