



# Attention Forest for Semantic Segmentation

Jingbo Wang<sup>✉</sup>, Yajie Xing<sup>✉</sup>, and Gang Zeng<sup>(✉)</sup><sup>✉</sup>

Key Laboratory of Machine Perception, Peking University, Beijing 100871, China  
{wangjingbo1219,yajie\_xing}@pku.edu.cn,  
zeng@cis.pku.edu.cn

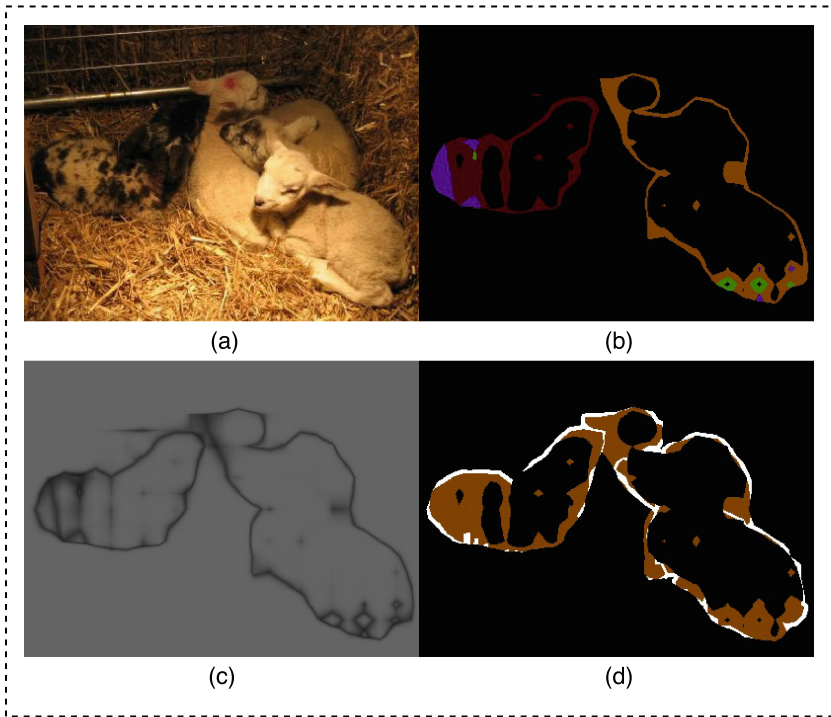
**Abstract.** Semantic segmentation is a classical task in computer vision. In this paper, we target to address the *low confidence regions* which traditional CNN can not solve very well in semantic segmentation task. Depending on different characteristics of *low confidence regions*, an adaptive and robust attention mechanism is important to focus on the informative regions but ignore the noisy parts in the image. Intuitively, one attention map only is not sufficient to model the interaction between the *low confidence regions* and its surrounding patches. Thus, in this paper, we propose an Attention Forest structure, a novel and robust attention mechanism, to handle the *low confidence regions*. Each Attention Tree structure can capture more interactions between current patches with its adjacent regions. Experiments on PASCAL VOC 2012 Dataset validate the effectiveness of our proposed algorithm.

**Keywords:** Semantic segmentation · Deep learning  
Attention mechanism

## 1 Introduction

Recently, fully convolutional networks (FCN) [16] is widely adopted as the general framework for semantic segmentation. FCN usually depends on the advanced deep network architectures, such as the ResNet, and classify every pixel in an image by shared convolution. It is difficult to predict all pixel correctly with high confidence in these structures. As shown in Fig. 1, the areas with different RGB surface are very easy to be mis-classified. According to the further experiments, their scores on probability map are also low. Thus, we define *low confidence regions* with prediction probability lower than  $\rho$  and *high confidence regions* on the counterpart. The FCN has a outstanding performance in the *high confidence regions*, but meet some trouble in *low confidence regions*, as shown in Table 1.

As shown in [4, 6, 10, 24, 26], the neural network only select some most representative regions but ignore other regions. However, in semantic segmentation task, the network should have the ability to adapt the different significations of all objects in the image. It is a challenge for the classification network architectures only based on the origin FCN network's presentation. This defect of FCN



**Fig. 1.** This figure show the *low confidence regions* problem of semantic segmentation task we observed on PASCAL VOC 2012 [5] dataset. (b) is the prediction of FCN base network in *low confidence regions* and (d) is the ground truth in same region. (c) is the probability map of the FCN predicted categories. Shadow regions shown in (c) are the low confidence regions. In (a), boundary and complex regions always confuse the FCN.

network causes some *low confidence regions* could not be “focused on” and result in the *low confidence regions* having a terrible performance.

To adapt more object’s significations and to solve the *low confidence regions* problem, we propose our Attention Forest network. Unlike others’ attention mechanism, we not only generate the attention enhanced by the origin network layer by layer, but also generate the reverse attention of origin attention. Reverse attention can capture the objects or the parts which have the different significance of origin FCN network. By employing the reverse attention, the FCN network can focus on more objects. The interaction between *low confidence regions* on different objects and surroundings patches can modeled better with our sufficient attention maps and reverse attention maps. The Attention Module also can obtain large context information. So the network structure can pay close attention to all *low confidence regions* and classify these regions better. We embed this attention module into a binary tree structure and each node of the tree can generate an attention and a reverse attention. Furthermore, the Attention Forest is the combination of

**Table 1.** The result of *low confidence regions* and *high confidence regions* with different threshold  $\rho$  based on ResNet-101 [9]. The metric is the standard *mIOU* on PASCAL VOC 2012 [5] validation dataset.

$\rho$	Low confidence regions(%)	High confidence regions(%)
0.90	39.79	80.46
0.95	43.62	81.82
0.985	49.35	83.61

different Attention Trees for creating more robust attention system. In our “*search-classify*” modeling approach, we make a progress on the *low confidence regions* problem.

Our contributions are summarized as follows: (1) we propose the *low confidence regions* problem in semantic segmentation task. (2) we propose an Attention Module that has origin attention and reverse attention with large context information.

## 2 Related Work

*Semantic Segmentation:* In recently years, studies of the semantic segmentation always employ deep convolution neural networks [2,12,16] instead of the hand-crafts features [11]. In this task the most common method is enlarge the receptive field and embedding different receptive context information. In [2,3,12,24], convolution layer with dilation can capture larger receptive field information than the ordinary convolution layer. Driven by the image pyramid, mutli-scale feature ensemble is always employed in semantic segmentation to capture different scope context information. In [2], an “ASPP” module is applied for ensemble multi-scale feature and in [3] improved the “ASPP” module. [25] applied different scales average pooling in their pyramid pooling module instead of dilation convolution layers.

In [1,7,16,17] also use different level feature of the base network. They refine the outputs of the base network by using before level layer’s context information. In [14] has a multi-path refine structure using different level features. In [18], a large kernel method is employed in CNN with encoder-decoder structure.

*Attention in CNN:* Attention mechanism is a import process in CNN to use top information guiding the feed-forward network [6,26]. In semantic segmentation task, attention mechanism is always used like a signification detection of the image. In [13], attention of CNN depends on the scale of input image. And in [19], the attention is used like a sign to let the network learn what not belong to the signification of the origin network.

### 3 Approach

In this section, we propose a novel network structure called Attention Forest to solve the *low confidence regions* problem in semantic segmentation task. We introduce the Attention Tree structure and its sub-module Attention Module in Sect. 3.1. In Sect. 3.2, we propose the Attention Forest structure. At last, we define our whole framework of Attention Tree and Attention Forest in Sect. 3.3.

#### 3.1 Attention Tree

After getting the feature of the base network, the Attention Tree consists of two modules: the Binary Tree module and the Attention module. In the Binary Tree module, each branch of the tree creates refine features of the upper level features. In the Attention module, we create the attention which the network should focus on and the attention which the feature of the upper level doesn't focus on.

*Definition.* Let  $I \in I$  donate the input image and  $F$  be the base FCN network. Feature  $f$  is created by the network as function  $f = F(I)$ . Let  $Att$  donate the attention map generated by attention network from feature  $f$ . The network structure of creating attention map can be written as  $F_{Att}$ .

$$Att = F_{Att}(f) \quad (1)$$

In Attention Tree module, we create reverse attention through the attention map  $Att$  from the network  $F_{Rev}$ .

$$Rev = F_{Rev}(Att) \quad (2)$$

We define the module which creates the attention feature map  $Att$  and reverse attention feature map  $Rev$  as Attention module. In attention tree, the structure is defined like a binary tree. The  $i$ th layer  $j$ th father node of the binary tree is  $FC_{ij}$ . Based on  $FC_{ij}$ , we create attention  $Att_{ij}$  and reverse attention  $Rev_{ij}$ .  $LC_{ij}$  donates the left child node feature of  $FC_{ij}$  and  $RC_{ij}$  is the right child node feature. As shown in Fig. 3(b), we multiply  $LC_{ij}$  by  $Att_{ij}$  and multiply the  $RC_{ij}$  by  $Rev_{ij}$  to create the feature map pay attention to different regions of input image  $I$ . Each node of the binary tree structure can be a father node to create a sub-tree network like above method and then we can get a multi-layer binary tree network. We define this module as our Attention Tree module.

*Attention Module:* To capture the signification of different objects or parts in the image, we propose our Attention module with reverse attention in this paragraph. In Attention module, we use stack of convolution layers to estimate the function  $F_{Att}$ . As shown in Fig. 2, we use three convolution layers to create a spatial wise and channel wise attention map. Each convolution layer has a  $3 \times 3 \times C$  kernel. We set BN [21] layer after each convolution layer and ReLU [8] layer only behind first two convolution layers. At the end of these layer we use a no-linear

normalization function to normalize the output. Mathematically , the value of the  $i$ th layer  $j$ th attention can be wrote as

$$Att_{ij} = F_{Norm}(F_{Conv}(FC_{ij})) \tag{3}$$

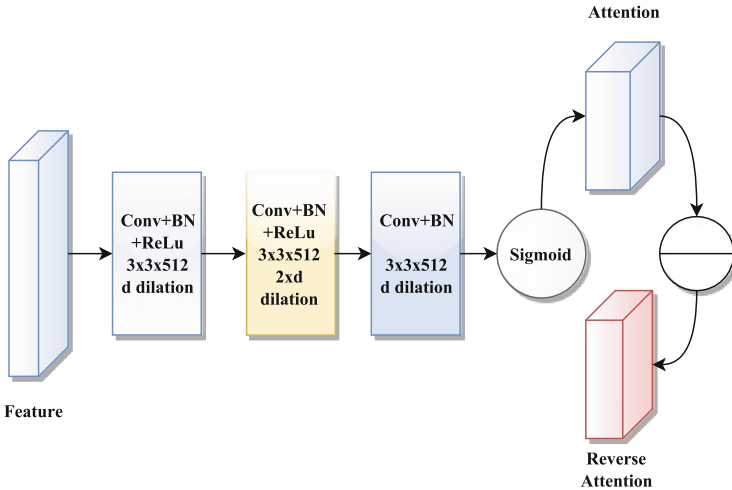
In our experiment, we set the  $C = 512$  and the  $F_{Norm}$  as *Sigmoid* function. So  $Att_{ij}$  can be wrote as

$$Att_{ij} = \frac{1}{1 + e^{-F_{Conv}(FC_{ij})}} \tag{4}$$

At the same time, we generate the reverse attention in this module. Reverse attention is the reverse signal of the origin attention. By using reverse attention, the network can capture semantic information or meaningful parts that the origin attention doesn't pay attention to. So we can solve the low confidence regions problem by finding the low confidence regions the network doesn't focus on firstly. To create the reverse attention, we  $F_{Rev}$  can be wrote as a simple function.

$$F_{rev}(x) = 1 - x \tag{5}$$

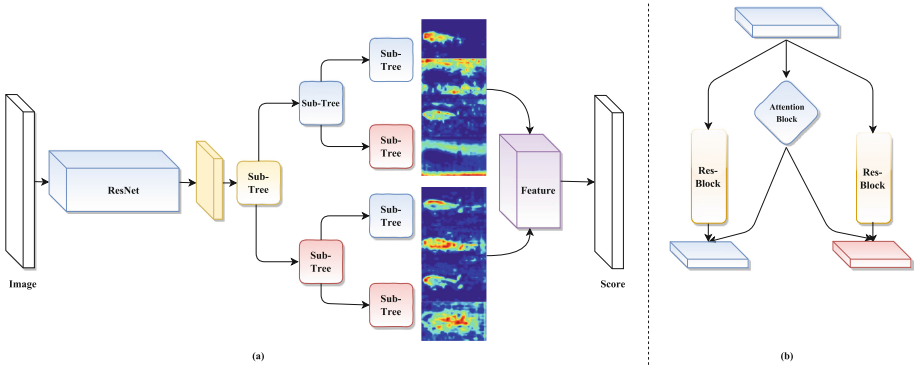
We suppose that the network only cares about the value of each pixel larger than 0.5. This function can create the reverse attention that care about different region from the origin attention. In our approach, we create the attention can cover all low confidence regions for semantic segmentation.



**Fig. 2.** Overview of our proposed attention module. We employ three ‘*atrous convolution*’ layers to capture the attention of the given input feature. Different dilation rate can get different significations of objects or object parts. We create the reverse attention in this module.

*Multi-Grid Dilation:* To classify the *low confidence regions* correctly, we should use larger scale context information besides “focusing on” it. In [2], they develop the ‘*atrous convolution*’ to capture larger context information. In our approach, we also use this attention module to capture larger scale context of the image. So as [3,23], the convolution layers in Attention module have *Multi-Grid* dilation rate and set a dilation multiplier  $d_i$  for *ith* layer’ Attention module. For example, we can set the dilation grid  $(1,2,1)$  and the  $d_i = 2$ . In this setting, the three convolution layers in Attention module has dilation rate  $(2,4,2)$ . We can set different dilation multipliers to Attention Module in different layers. With the network going deeper, the network can capture larger extent context. The reverse attention generates different attention with the receptive field changing. In our attention tree, we set  $d_{i+1} = 2 \times d_i$ .

*Binary Tree Module:* In binary tree module, each branch of the binary tree is a Bottleneck block in ResNet-50 [9]. We set this branch to refine the feature map of the binary tree module’s father node. In our model, we create a three layers binary tree for semantic segmentation task. The father node of the binary tree is the output of the base FCN model and the output of each branch is the father node of next sub binary tree module. In the third layer, we set the half of the before two layer’s channel number in attention module to reduce the calculating.



**Fig. 3.** (a) shows the whole pipeline of our Attention Tree Module. Given an input image, we first employ the CNN to get the output feature by the last convolution layer. Then we use a multi-layer Attention Tree module to capture part information of the input feature. The attention can pay attention to more parts of the objects than the base CNN and each branch can emphatically **solved a part region segmentation by our reverse attention mechanism**. In the Attention Tree Module, each sub-tree is shown in (b) and the Attention Block is shown in Fig. 2. The Res-Block is created same as the bottleneck module in [9].

### 3.2 Attention Forest

The Sect. 3.1, we particularly describe how to create an attention tree. Influenced by the random forest algorithm, we build an Attention Forest to improve the performance. The key of creating different attention trees is designing different attention module. We use different *multi-grid* rate, different base dilation rate and Global pooling attention module for designing attention forest to capture different receptive fields context information.

*Atrous Convolution Attention Tree:* As shown in Sect. 3.1, we employ three *atrous convolution* [2] in our attention module. Different dilation grid can capture different receptive filed and create different attention. In our attention forest, we create another two different attention trees with *atrous convolution* attention from the attention tree we proposed in Sect. 3.1 which we named *Tree<sub>1</sub>*. Firstly, we can change the number of convolution in attention module. In *Tree<sub>2</sub>*, We reduce one convolution layer and set the dilation grid (1, 1) in *Tree<sub>2</sub>*'s attention module. In *Tree<sub>3</sub>*, we employ the same dilation gird but set a base dilation rate half of *Tree<sub>1</sub>*.

*Pooling Attention Tree:* To make differences from the other trees, we create *Tree<sub>4</sub>* without *atrous convolution*. We replace the *atrous convolutions* in attention module by a global pooling and a  $1 \times 1$  kernel convolution layer to capture global context but different from *atrous convolution*. This global context will enhance all point in the feature map by the same signal.

### 3.3 Framework

Our Attention Tree model is shown in Fig. 3. We use the pre-trained ResNet [9] as our feature network. After the feature network, we obtain the coarse segmentation feature map. We send this feature map to our Attention Tree. The outputs of our Attention Tree are 8 feature maps which have 128 channel. We concatenate these feature maps and use a  $1 \times 1 \times 512$  convolution layer, a BN [21] layer and a ReLU [8] layer (*conv-bn-relu*) to ensemble different feature. Then we use a  $1 \times 1 \times 21$  convolution layer and the Softmax function to obtain the prediction score map.

In our Attention Forest, as same as the Attention Tree, we send the output feature map of ResNet to each Attention tree and use the same *conv-bn-relu* block to capture different context information. We concatenate the coarse segmentation feature map and each Attention Tree's output feature map to obtion the prediction score map.

## 4 Experiment

In this section, we will introduce our experiment with Attention Tree and Attention Forest. We evaluate our approach on standard benchmark PASCAL VOC 2012 [5]. We choose the ResNet-101 (pre-trained on ImageNet [20]) as our base model for fine tuning. We use SGD optimization algorithm with batch size 16, momentum 0.9 and weight decay  $1 \times 10^{-5}$ , in our training process. We also set the a 'poly' learning rate (as in [12]) with initial learning  $1 \times 10^{-2}$  and 0.9 *power*. The performance is measured by standard mean intersection-over-union(IoU). Our baseline is the ResNet with  $16 \times$  downsample by setting the last block a 2 dilation in  $3 \times 3$  convolution layers.

In next subsection, we will enumerate a series of ablation experiments to evaluation the performance of our approach and show the function of the Attention Forest and Attention Tree. Then we will report the full results of our approach on PASCAL VOC 2012 test dataset.

### 4.1 Ablation Studies

In this subsection, we will firstly compare the results of different layer Tree structure model. Then, we will examine the effort of the attention and the reverse attention on our baseline network. Besides these, different Attention Trees and different combinations of Attention Forest will be compared.

*Layer matters:* In Sect. 3.1, we propose that an attention tree has multi-layer structure rather than single-layer structure. The key of creating multi-layer Attention Tree is creating the different receptive field of attention in different layer. Like [3], we set gradually larger dilation rate with the Attention Tree going deeper. For example, if the attention module has three convolution layers, we can apply the multi-grid method to this module. *Mult-Grid* =  $(r_1, r_2, r_3)$  are applied for each attention module and the dilation rate( $d_i$ ) is multiplied by 2 with the attention adding one layer.

**Table 2.** The result of our model base on resnet101. The first column is the mean IOU with the whole Attention tree and the second is the tree structure without attention module in Table 2. The mean IOU is improved with the Attention tree going deeper. Compare with the tree structure has attention module or not, the attention module improve the performance in each layer.

Method	With Attention(%)	W/O Attention(%)
Baseline	73.02	73.02
Layer One	75.60	74.23
Layer Two	76.51	74.26
Layer Three	77.62	74.00



From Table 2, we can find that employing in ResNet-101, the performance of our model become better with the layer going deeper. We also compare the tree structure which has the Attention Module or not in each layer.

*Reverse Attention matters:* In our attention tree, the attention module is the key module of capture different region and enlarge the receptive. In our proposed model, we compare our attention module with the attention module doesn't have the reverse attention. As shown in 3, employing ResNet-101, our Attention Module achieve a better performance both in mIOU or the *low confidence regions* mIOU. So from Table 3, our reverse attention supplement the origin attention and both of them solve the low confidence regions jointly, better than only use the origin attention.

**Table 3.** Compare the mIOU and the *low confidence regions* mIOU based on ResNet-101, The first line is the baseline of ResNet-101 in mIOU and hard IOU, the second line is the result of the Attention Tree module without the reverse attention. The third line is the result of our Attention Tree.

Method	mIOU(%)	LCR mIOU(%)
Baseline	73.02	39.79
W/O Reverse	76.50	50.31
With Reverse	77.62	54.08

Further more, in Sect. 3.2, we create an Attention Forest model for segmentation task. The forest based on different tree with different Attention Module. Different Attention Module can capture different size receptive field context information and different type context information. As shown in Sect. 3.2, we compare different combination of these Attention Trees in Table 4.

In Table ??, we find that our Attention Tree which defines in Sect. 3.1 achieve the best performance of the four Attention Tree. But in our Attention forest, we just want to use some weak feature extractor and make a strong feature extractor. We compare the different combination of these trees. From the Table 4, we find that the performance in mIOU and Hard mIOU is improved by combining more

**Table 4.** Compare the mIOU and the *low confidence regions* mIOU of different attention tree combination based on ResNet-101.

Method	mIOU(%)	LCR mIOU(%)
Baseline	73.02	39.79
Tree-1	77.62	54.08
Tree-1,2	78.15	55.53
Tree-1,2,3,4	78.52	56.94



**Fig. 4.** Examples of our prediction on Pascal VOC 2012 validation dataset. We can find that the low confidence regions can be solved and more objects can be found in our method.

Attention Trees. The whole Attention Forest can achieve 78.52% in mIOU and 56.94% in hard mIOU, which has a 0.9% and 2.86% improvement in mIOU and hard mIOU than the single Attention Tree.

## 4.2 Experiment

In this subsection, we will discuss our experiment on PASCAL VOC 2012 dataset. We use flip both in training and evaluation for our network.

*PASCAL VOC 2012:* We split our experiment into three stages. (1) *stage-1*, we mix up PASCAL VOC 2012 images and SBD for training, like ablation study. (2) *stage-1*, we only employ PASCAL VOC 2012 dataset and fine-tune the pre-train model in *stage-2*. We achieve a 80.52% mIOU on validation dataset and 79.97% on test dataset. (3) We fine-tune our model on MS-COCO [22] dataset and finally achieve 84.60% mIoU. We choose some prediction on PASCAL VOC 2012 validation dataset and show in Fig. 4 (Table 5).

**Table 5.** Results on PASCAL VOC 2012 testing set. [\*] means pre-trained on MS-COCO dataset.

Method	mIOU(%)
FCN [16]	62.2
Deep [12]	71.6
CRF-RNN [27]	72.0
Piecewise [15]	75.3
Ours	79.97
Ours[*]	84.60

## 5 Conclusions

Our proposed Attention Forest is the combination of different types of Attention Tree. Each Attention Tree can capture large receptive context information and its reverse information. This structure can find all objects in the image and solve the *low confidence regions* problem. In our ablation experiment, we find the large receptive attention and our reverse attention in Attention Forest structure can enhance the performance in “*low confidence regions*”. We do experiments on PASCAL VOC 2012 and achieve a comparable result against the *state-of-the-art* methods.

**Acknowledgements.** This work is supported by the National Key Research and Development Program of China (2017YFB1002601) and National Natural Science Foundation of China (61375022,61403005,61632003).

## References

1. Badrinarayanan, V., Handa, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. *IEEE Trans. Pattern Anal. Mach. Intell.* (2017)
2. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **40**, 834–848 (2018)
3. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587* (2017)
4. Walther, D., Itti, L., Riesenhuber, M., Poggio, T., Koch, C.: Attentional selection for object recognition — a gentle way. In: Bülthoff, H.H., Wallraven, C., Lee, S.-W., Poggio, T.A. (eds.) *BMCV 2002. LNCS*, vol. 2525, pp. 472–479. Springer, Heidelberg (2002). [https://doi.org/10.1007/3-540-36181-2\\_47](https://doi.org/10.1007/3-540-36181-2_47)
5. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. <http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html>
6. Wang, F., et al.: Residual attention network for image classification. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3156–3164 (2017)

7. Ghiasi, G., Fowlkes, C.C.: Laplacian pyramid reconstruction and refinement for semantic segmentation. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 519–534. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46487-9\\_32](https://doi.org/10.1007/978-3-319-46487-9_32)
8. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (2011)
9. 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)
10. Gregor, K., Danihelka, I., Graves, A., Rezende, D.J., Wierstra, D.: Draw: a recurrent neural network for image generation (2015)
11. Kae A, Sohn K, Lee, H.: Augmenting CRFs with Boltzmann machine shape priors for image labeling, pp. 2019–2026 (2013)
12. Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected CRFs (2015)
13. Chen, L.-C., Yang, Y., Wang, J., Xu, W., Yuille, A.L.: Attention to scale: scale-aware semantic image segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3641–3649 (2016)
14. Lin, G., Milan, A., Shen, C., Reid, I.: RefineNet: Multi-path refinement networks with identity mappings for high-resolution semantic segmentation (2017)
15. Lin, G., Shen, C., van den Hengel, A., Reid, I.: Efficient piecewise training of deep structured models for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3194–3203 (2016)
16. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
17. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1520–1528 (2015)
18. Peng, C., Zhang, X., Yu, G., Luo, G., Sun, J.: Large kernel matters-improve semantic segmentation by global convolutional network (2017)
19. Huang, Q., et al.: Semantic segmentation with reverse attention (2017)
20. Russakovsky, O., et al.: ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis. (IJCV)* **115**(3), 211–252 (2015). <https://doi.org/10.1007/s11263-015-0816-y>
21. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift, pp. 448–456 (2015)
22. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-10602-1\\_48](https://doi.org/10.1007/978-3-319-10602-1_48)
23. Wang, P., et al.: Understanding convolution for semantic segmentation. arXiv preprint [arXiv:1702.08502](https://arxiv.org/abs/1702.08502) (2017)
24. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. arXiv preprint [arXiv:1511.07122](https://arxiv.org/abs/1511.07122) (2015)
25. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network (2017)
26. Zhao B, Wu X, F.J.: Diversified visual attention networks for fine-grained object classification. [arXiv:1606.08572](https://arxiv.org/abs/1606.08572) (2016)
27. Zheng, S., et al.: Conditional random fields as recurrent neural networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1529–1537 (2015)