Multimed Tools Appl (2018) 77:22159–22171
https://doi.org/10.1007/s11042-018-5704-3

Understanding the effective receptive field in semantic image segmentation

Yongge Liu[1,3,4] **· Jianzhuang Yu**[2] **· Yahong Han**[2]

Received: 13 September 2017 / Revised: 14 December 2017 / Accepted: 22 January 2018 /
Published online: 31 January 2018

Abstract Deep convolutional neural networks trained with strong pixel-level supervision have recently significantly boosted the performance in semantic image segmentation. The receptive field is a crucial issue in such visual tasks, as the output must capture enough information about large objects to make a better decision. In DCNNs, the theoretical receptive field size could be very large, but the effective receptive field may be quite small. The latter is an really important factor in performance. In this work, we defined a method of measuring effective receptive field. We observed that stacking layers with large receptive field can increase the size of receptive field and increase the density of receptive field. Based on the observation, we designed a Dense Global Context Module, which makes the effective receptive field coverage larger and density higher. With the Dense Global Context Module, segmentation model reduces a large number of parameters while the performance has been substantially improved. Massive experiments proved that our Dense Global Context Module exhibits very excellent performance on the PASCAL VOC2012 and PASCAL CONTEXT data set.

✉ Yahong Han
yahong@tju.edu.cn

Yongge Liu
liuyongge@aynu.edu.cn

Jianzhuang Yu
jzyu@tju.edu.cn

[1] School of Computer and Information Engineering, Anyang Normal University, Anyang, China

[2] School of Computer Science and Technology, Tianjin University, Tianjin, China

[3] Henan Key Laboratory of Oracle Bone Inscriptions Information Processing, Anyang Normal University, Anyang, China

[4] Collaborative Innovation Center of International Dissemination of Chinese Language Henan Province (HNIDCL), Henan, China

Keywords Semantic segmentation · Effective receptive field · Dilated convolution

1 Introduction

Deep Convolutional Neural Networks(DCNNs) have achieved great successes in vision recognition tasks, including image classification [15], object detection [10], pose estimation [34], and semantic image segmentation [3, 8, 20]. Because of its ability of feature extraction and context information abstraction, DCNNs have greatly enhanced the performance of semantic segmentation, compared to feature engineering.

There are two main challenges in semantic segmentation task. One is pixel classification, and the other is pixel localization. Before CNN was applied to semantic segmentation, DCNNs [12, 32] achieved very high accuracy on the task of image classification. After CNN is applied to semantic segmentation, pixel-wised classification and localization have became the major challenge. As described in [20, 21], with the deepening of the network, semantic information of features becomes more and more abstract, and meanwhile, the loss of location information is more and more serious.

Many works focused on how to obtain more accurate location information. There are two main directions for solving this problem. One direction is to exploit encoder-decoder structure [1, 9, 16, 24, 27, 29, 30]. The encoder part extracts semantic information, and the decoder part recovers spatial resolution, which is achieved by cascading multiple deconvolutional layers. The other direction [3, 36] is to keep image resolution as much as possible in the encoder part, with a simple interpolation algorithm in the decoder section. In order to maximize the retention of the original pixel location information, T. Pohlen et al. [29] even maintain the size of feature map same as original image during the entire propagation of CNN. Besides, some works [17, 19, 37] through integrating Condition Random Field (CRF) [14] into CNN also can achieve pixel spatial information.

More recently, many works [3, 5, 22, 27, 35, 36] focus on increasing receptive field. Large receptive field could improve the accuracy of pixel-wised localization and classification at the same time. The concepts of receptive field, or field of view (FOV) is a very critical perspective on understanding how DCNNs work. As an output unit of network extracts information from input unit which is within the scope of its receptive field. Any input unit which outside the receptive field could not provide information to the output unit. If the receptive field size of an output unit is not big enough, then it will not be able to obtain enough information to make the right decision. This will directly affect the performance of the experiment. In many visual tasks, especially dense prediction tasks like semantic image segmentation, it is important and necessary to carefully control the size of receptive field.

Some works [3, 27, 35, 36] obtained a relatively large theoretical receptive field by setting network parameters and changing network structure. However, they did not further analyze how the receptive field works. Some works have attempted to propose novel methods to visualize and observe effective receptive field(ERF). Zhou et al. [38] through simplifying image representation and highlighting the elements that lead to the high classification score, observed the effective receptive field of specific class and specific image. Long et al. [21], through associating CNN feature with receptive field to visualize CNN feature, found that convnet features localize at a much smaller scale and have intraclass correspondence, and then applied to keypoint localization task. Luo et al. [22] gived some theoretical derivations of effective receptive field and verification experiments, and analyzed properties of effective receptive field.

The above several works are related to receptive field. Some of them have complex algorithms complicated procedures and lots of computation, and can only roughly observe the receptive field. Or they didn't have an analysis of the relationship between receptive field and experiment performance. It is difficult for us to directly utilize these methods of analyzing receptive field to improve the performance of our own particular tasks. Papers that specialize in the receptive field are rare. The relationship between receptive field and semantic segmentation is still not clear.

In this paper, we hope to find some clues where are the driving force behind the significant improvement of many FCN-based vision tasks recently. We introduce a concise algorithm and a novel visualization technique on effective receptive field (ERF). Unlike other works [21, 25], Our algorithm for calculating ERF is category-independent and has general characteristic. At the same time, the process is simple and the amount of calculation is small. We proposed the concept of Effective Receptive Field Intensity (ERFI). Experiments show that ERFI could directly reflect the relation between effective receptive field and semantic performance. We can also easily use ERFI to analyze other tasks.

In the visualizing experiments, we discovered an unusual problem: if one layer has too large theoretical receptive field in the network, the effective receptive field may come a discrete and sparse regional distribution. We designed a Dense Global Context Module (DGCM) to alleviate such problem. Dense Global Context Module (DGCM) makes ERF cover the whole input field and ERFI denser. Because of our precisely control of ERF, the redundancy of network parameters is relatively small, resulting in a reduction of parameters. After integrated Dense Global Context Module into the network, the performance on semantic segmentation has a significant improvement. Compared to methods fusing multi-scale layer to further promote performance, our Dense Global Context Module has less parameters and be faster in inference.

This paper is organized as follows. We first review the related work in Section 2. In Section 3, we will discuss the relationship between effective receptive field and semantic segmentation model in detail. Section 4 presents experimental setup and results. The paper is concluded in Section 5.

2 Related work

In the following, we review recent advances in semantic segmentation tasks. As mentioned above, it is helpful to improve the segmentation performance through deconvolution, keeping the feature map resolution as much as possible and using the CRF to obtain pixels localization information. In this work, we discuss CNN-based semantic segmentation from four general directions.

Encoder-deconder This structure consists two parts. In the encoder part, CNN extracts feature information from input image, and the feature is gradually abstracted from low-level local information to high-level global semantic information. In this process, the feature map resolution is decreasing. In the decoder part, hight-level semantic information recover the pixel localization information step by step. In this process, the feature map resolution is increasing.

Many works under this structure show two trends. One trend is a complete decoder procedure. Each step of decoder corresponds to every step of encoder. SegNet [1] records the position indexes of subsample operation in the encoder section, and then apply these recorded indexes to upsample in the encoder section. U-Net [30] has similar decoder section with SegNet, except that U-Net increase the feature map resolution through deconvolution instead of upsampling. In the work of DeconvNet [24], a fully convolutional network which is transformed from CNN is used, and each layer of decoder part still is correspond to each layer in the encoder part.

The other trend is a brief decoder procedure. The main characteristic of this trend is that the system do not reduce the feature map resolution too much in the encoder part, and then restore resolution through a simple interpolation algorithm in the decoder part. In this trend, the most representative work is Deeplab [3], and many more innovative work on the basis of this work [4, 26, 36]. In such works, the feature map resolution reduced by 8 times in the encoder part, while the general CNN structure will shrink by 32 times. In order to maximize the retention of pixel spatial information, [29] maintain a stream to store feature map full resolution in the process of encoder and decoder.

Conditional Random Field To further improve the localization ability, CRF is introduced into the semantic segmentation task. DenseCRF [14] utilizes the pixel similarity to make the boundaries of objects finer. Deeplab [3] exploits denseCRF as a post-processing to further refine semantic segmentation results. CRF-RNN [37] integrate CRF into CNN for the first time, and jointly train and predict. But this method of implementation needs a lot of CPU computation. Then, DPN [19] makes a different approximation algorithm to approximate the mean field algorithm as convolution operation and pooling operation. This method realizes the calculation of CRF on GPU completely, and accelerates the calculation speed. Furthermore, Adelaide [17] replaced hand-crafted potential functions with small-size network which consist convolutions and nonlinearity operations.

Multi-scale A picture usually contains more than one object, and different objects have different scales. How to better recognize objects of different scales is also a very important factor in improving performance. Many works [6, 8, 28] use multi-scale images as CNN inputs. This way shares the same network, feeding the network with different input sizes. Large scale inputs can obtain more detailed information about small objects. However, this method is limited by GPU memory. Feeding network with too large inputs is a problem. At the same time, there is a large amount of redundant computation in this multi-scale method. [3, 36] by using the spatial Pyramid pooling at the end of CNN obtain multi-scale information. In this way, Large scale information acquisition is no longer limited. And the computation cost is very small.

Increase Receptive Field The latest few works [5, 27, 36] all mentioned that increasing receptive field size is a key point in improving segmentation performance. Increasing receptive field mainly exist in the following ways. One way is to deepen the network [12, 32]. With stacking more convolutional layers and pooling layers, the network's Receptive Field will be gradually increased. But using this way to improve segment performance is very difficult. [27] could quickly increase receptive field of the network. The problem is that the parameters and computation of the network will increase exponentially. The large pooling kernel also can multiplied the size of receptive field without increase the network parameters and the amount of computation is relatively small. Dilated convolution [3] has the ability to extend the receptive field of one layer to any size.

3 Method

As discussion above, there are many ways to increase the size of ERF. The methods of deepening the network and using large convolution kernel will increase the network complexity, and lead to longer training time. Pooling operation has the problem of losing resolutions. We use dilated convolution, which avoids all of the above problems, as the tool for observing receptive field in this work.

3.1 Dilated convolution

Dilated convolution could skip several elements to convolution, not just adjacent elements. We consider One-dimensional sample first. Let $y\left(i\right)$ be the output of dilated convolution and $x\left(i\right)$ is the 1-D input. The filter is $w\left(k\right)$ with kernel size K. Dilated parameter is d. The dilated convolution is defined as follow:

$$y\left(i\right)=\sum_{k=1}^{K}x\left(i+d\cdot k\right)w\left(k\right) \tag{1}$$

Figure 1 shows dilated convolution operated in two-dimensional space. Supposed that the first three consecutive convolution layers of a network use dilated parameter $\{1, 2, 4\}$ respectively. The maps (a), (b), (c) are the corresponding output of three layers. The maps show the theoretical receptive field size of each layer. Theoretical receptive field size of map (a) is 3. The RF size of next layer increases on the basis of current layer. So the theoretical RF size of (b) is 7, and (c) is 15.

3.2 Effective receptive field

We follow the definition of the Effective Receptive Field given by [22]. The region containing any input pixel with a non-negligible impact on the central output unit is defined as the Effective Receptive Field of that output unit. Let $x^{n}_{i,j}$ donate the (i, j)th input unit of nth layer. The central unit of input or output map is indexed by $(0, 0)$. So x^{n+1} is the $(n+1)$th

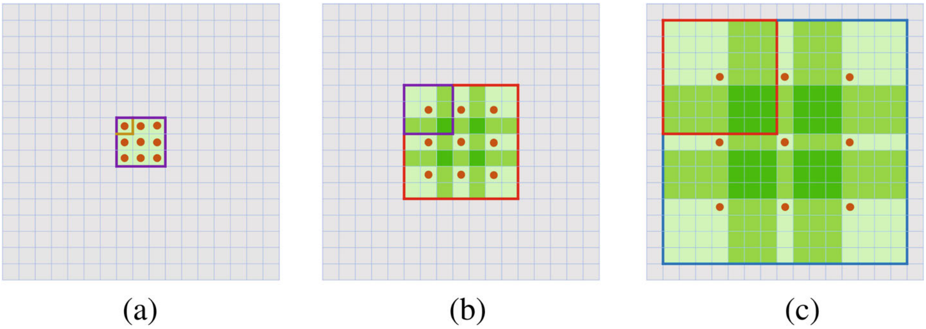


Fig. 1 Illustration of the first three successive convolution layers use dilated parameter $\{1, 2, 4\}$ with kernel size 3. There are two boxes in each map. The small box is the RF size of a unit before convolution, and the large one is the RF size after convolution

layer input map, and also is the nth layer output map. x^0 is the net input map. x^t stands for the network final output map. The partial derivative $\frac{\partial y^t_{0,0}}{\partial x^0_{i,j}}$ is used as evaluation formula for the impact factor of output unit.

The interpretation of computing ERF impact factor using partial derivative is that derivative indicates which input unit need to be changed the least to affect the output unit decision most. We note that similar technique has been previously applied in the context of Bayesian classification [2] and image-specific class saliency [33].

Note that this measurement will not only depend on the weights of the network, but also the input when the network contains non-derivative function processing, such as Pooling and Relu. Therefore, when testing EFR, we also need to describe the input distribution.

The impact factor derivative can be computed by back propagation. The formula for calculating gradients by chain rules in normal back propagation is as follows:

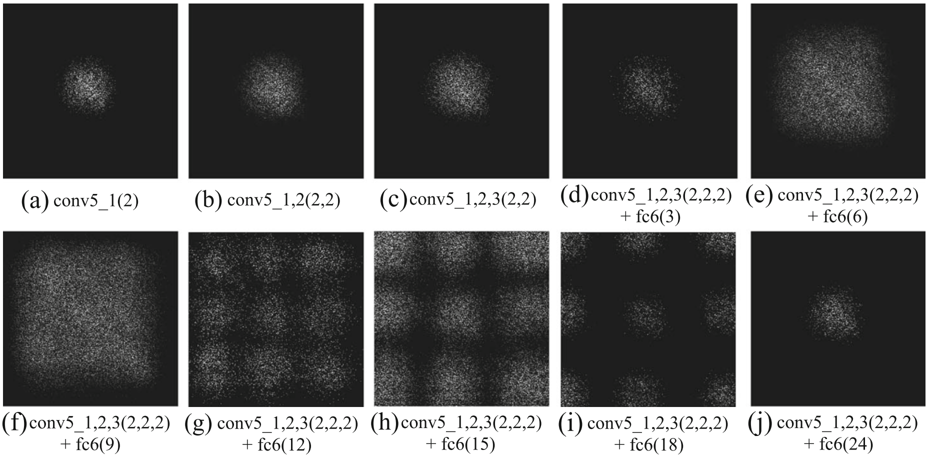
$$\frac{\partial l}{\partial x^0_{i,j}} = \sum_{i',j'} \frac{\partial l}{\partial x^t_{i',j'}} \cdot \frac{\partial x^t_{i'j'}}{\partial x^0_{i,j}} \tag{2}$$

where l stands for the loss function. (i', j') is the passed unit index of intermediate layer while back propagation. In order to achieve impact factor derivatives, we manually set the loss gradient $\frac{\partial l}{\partial x^t_{0,0}} = 1$ and $\frac{\partial l}{\partial x^t_{i,j}} = 0$ for all the rest. After such loss gradient backward to the first layer, we could get the impact factor gradient map of the center output unit.

3.3 Effective receptive field intensity

In above sections, we have provided methods with computing the impact factor of each input unit to the output unit. The value of this factor may be positive or negative. In general, positive gradients contribute to correct classification, whereas negative gradients otherwise. Therefore, when measuring ERF, we only use positive values and abandon negative values. In the experiments, we found that the impact factor values of network initialized by Gauss random algorithm present a Gauss distribution in the two-dimensional space on the map. On measuring two different receptive fields, we need to consider both coverage area of the ERF and the impact factor values of each input unit. Here, we propose the concept of Effective Receptive Field Intensity, which could take into account the both two aspects. Assume same input, and we have obtained raw gradient map after back propagation. First, we remove the negative values of gradient map. Then, the values of the gradient map are normalized to [0, 1]. Last, the sum of the gradient is the Effective Receptive Field Intensity. Note that [22] also gives a method of measuring ERF, but only applies to the ERF with Gauss distribution. The experimental results showed that ERFI performs better on describing relation between ERF and semantic segmentation performance, as shown in Fig. 4.

In the Experiments, it showed that the effective receptive field size is very small compared to theoretical RF size. The dilated convolution can extend the theoretical RF to arbitrary size, but the upper limit size of ERF is the input image size. If the difference of RF size between two adjacent layers is too big, the ERF presented a discrete regional distribution as shows in Fig. 2f-j. Within the range of input image size, the semantic segmentation performance correspondingly increases as the ERF size increases. Although dilated convolution could make RF size zoom to any size, too large RF size would not contribute to segmentation performance improvements.



(a) conv5_1(2) (b) conv5_1,2(2,2) (c) conv5_1,2,3(2,2) (d) conv5_1,2,3(2,2,2) + fc6(3) (e) conv5_1,2,3(2,2,2) + fc6(6)

(f) conv5_1,2,3(2,2,2) + fc6(9) (g) conv5_1,2,3(2,2,2) + fc6(12) (h) conv5_1,2,3(2,2,2) + fc6(15) (i) conv5_1,2,3(2,2,2) + fc6(18) (j) conv5_1,2,3(2,2,2) + fc6(24)

Fig. 2 Illustration of different network ERF with different dilated parameter combination. First, we set the dilated parameter to 1 in all layer. conv5_1_2(2,2) means dilated parameters in conv5_1 layer and conv5_2 are 2, 2 respectively. C_1_2(10,12) means dilated parameter of the first large RF layer is 10, and the second layer is 12

3.4 Dense global context module

In general, the first five-level convolution layers of DCNNs are used to extract basic context information from pixel-level image. The context module is a major portion of enlarging RF size, and also is an important part of further processing context information. We use the large dilated parameter to rapidly expand ERF size in context module. With the large receptive field, context module will be able to utilize more context information to learn, and have a more accurately prediction. Experiments showed that single layer with large dilated parameter could obtain large RF size, but the available RF signal is weak which leads to insufficiently utilize of context information. For the purpose of sufficiently utilizing context information, making ERF bigger and ERFI denser, we proposed the Dense Global Context Module. As shown in Fig. 3, DGCM stacked two layers with large receptive field size, which could ensure that the output unit can perceive global context information with enough signal strength.

4 Experiment

The Deeplab-LargeFOV net presented by [3] as our base network. We use Caffe [13] as our experimental platform to implement our framework.

4.1 ERF visualization

Experiment Setup We use Gaussian random algorithm with std 0.01 to initialize all the convolutional layers weights. In order to avoid difference numerical distribution among input image, all experiments were set to uniform input. All pixel values of the input image are set to 128. In the last layer, we manually set the loss gradient of central output unit to 1, and the rest gradients set to 0. After a BP operation, we obtained the raw impact factor

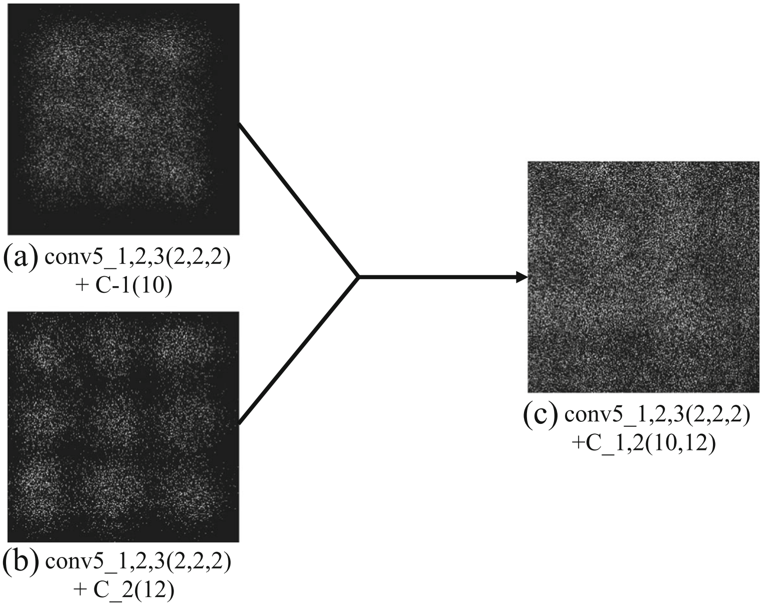


Fig. 3 Illustration of Dense Global Context Module. The dilated paremeter of first large RF layer is 10, and the second is 12

gradient map. Then removing negative values and normalization, we achieved the ERF map. With sum up all the gradients of ERF map, we got the ERFI. Through converting ERF map from color image to gray-scale image, we achieved ERF visualization.

Visualization Results Fig. 2 shows different network ERF map with different dilated parameter combination. First, we set the dilated parameter to 1 in all layers. conv5_1_2(2,2) means dilated parameters in conv5_1 layer and conv5_2 are 2, 2 respectively. C_1_2(10, 12) means dilated parameter of the first large RF layer is 10 and the second layer is 12.

Figure 4 shows the distribution of mIOU and ERFI on theoretical RF among different network. The network used for this experiment is the same network in Fig. 2. Here the results are averaged across 100 runs with different random weights. It can be seen that the distribution of ERFI is closely related to the segmentation performance distribution. This proves that ERFI could measure ERF more accurately.

Because of large RF size gap between two adjacent layers, ERF map appears a discrete regional distribution like Fig. 2f. In this case, ERFI is sparse. We designed a Dense Global Context Module to solve this problem. Figure 3 shows Dense Global Context Module structure. We insert a layer with big dilated parameter between pool5 and fc6 layer. DGCM keeps the network's ERF size large enough and makes ERFI more intensive like Fig. 3c.

We also have tried many more methods, like stacking more large receptive field layers to make ERFI more intensive, and using different dilated to obtain different scales context information to enhance the performance of segmentation model. The experimental results showed that the performance of these methods is less than network with Dense Global Context Module. That is because the DGCM has made each output unit possess a sufficiently large ERF and a dense enough ERFI. In other words More dense ERF won't help, and the role of contextual information has reached its limit.

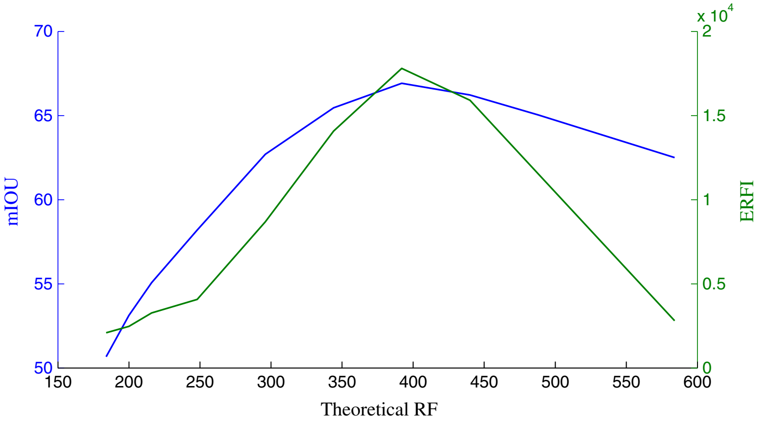


Fig. 4 Distribution of mIOU and ERFI on theoretical RF

4.2 PASCAL VOC2012

4.2.1 Dataset

In this work, we use Pascal VOC2012 [7] segmentation benchmark. The Pascal VOC2012 dataset consists of 1,464 training images, 1,449 validation images, and 1,456 test images, involves 20 foreground object classes and one background class. For training, we used the augmented PASCAL VOC (*PASCAL VOC aug*) training data including 10,582 images provided by Hariharan et al. [11]. The performance is measured in the standard metric of mean pixel intersection over union (mIOU), with the mean taken over all classes, including background.

Experiment Setup We use Imagenet-pretrained VGG-16 [32] model as our initial weights. In order to avoid the scale effects among context module, we only use VGG-16 pre-trained model to initialize the conv1 layer to conv5 layer weights of our network. Layers after fc6 which is the context module section, we use Gaussian random initialization with std 0.01. However, Gaussian random initialization bring us a marked drop in performance. To alleviate such impact as much as possible, we use two step training same as [31]. We used step training policy to train our segmentation network with batch size 10, learning rate 0.001, learning policy ‘poly’, momentum 0.9 and weight decay 0.0005. First step of step training, we fixed conv1 layer to conv5 layer’s weights, only fine-tuning layers after pool5 with 10,000 iterations. Second step, we release the fixed weights. All weights get fine-tuned for 20,000 iterations. Similar to [37], we also exploit pre-training the model on MS-COCO dataset [18].

Experiment Result Table 1 shows segmentation results. We used different training data in the training process. The first group was initialized by VGG-16 and used *PASCAL VOC aug* to train. The second group additionally used MS-COCO dataset to pre-train. We listed the number of parameter within context module among the networks. Experimental results show that our results are higher than the Deeplab-LargeFOV net by 2.9% with only a small amount of parameters added, and higher than Deeplab-ASPP [3] net by 1.7 % with almost just one-third of its parameters. we also list the runtime among different

Table 1 Results on PASCAL VOC2012 val set

NetWork	Context params	Runtime (ms/image)	mIOU
VGG16+VOC			
FCN-8s [20]	16.7M	61.6	62.20
Deeplab-LargeFOV	5.8M	56.6	64.33
Deeplab-ASPP	23.1M	75.4	65.17
DGCM(ours)	**8.1M**	**57.9**	**66.16**
VGG16+VOC+COCO			
Deeplab-LargeFOV	5.8M	56.6	66.92
Deeplab-ASPP	23.1M	75.4	68.11
DGCM(ours)	**8.1M**	**57.9**	**69.80**

Bold are used to highlight the results from our proposed method

networks for a further comparison. Experiments showed that our DGCM exhibits very excellent performance. Figure 5 presents some comparison of segmentation visualization among Deeplab-LargeFOV, Deeplab-ASPP and network with DGCM.

4.3 PASCAL context

We also verified our approaches on the PASCAL Context dataset [23]. The PASCAL Context dataset are relabeled as pixel-wise to the whole scene, which images are from Pascal VOC 2010. This dataset totally contain 4,998 training images and 5,105 validation images, including 459 semantic classes. Following [23], our proposed models are trained with the most frequent 59 classes along with the background category. Since the data set added much more categories, to better identify small objects, we have added another DGCM into

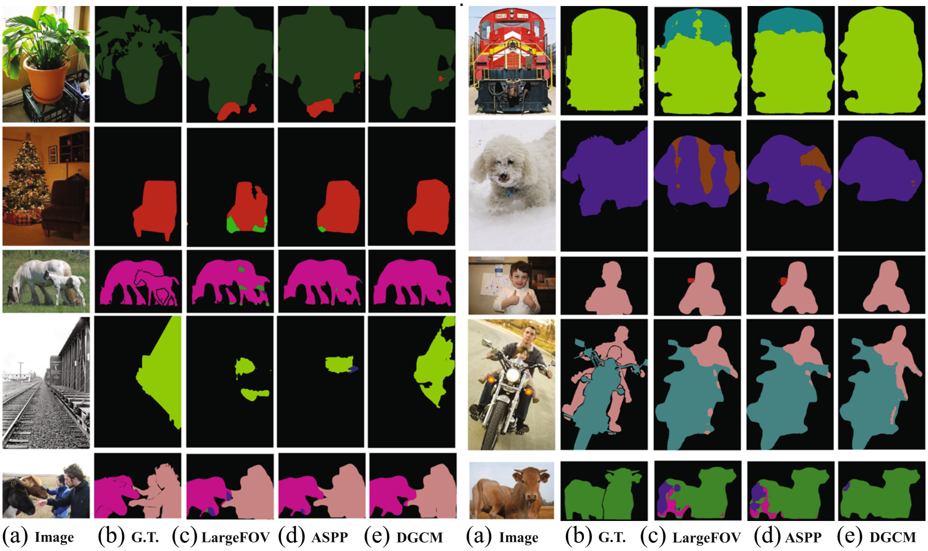


Fig. 5 PASCAL VOC 2012 val results of Deeplab-LargeFOV net, Deeplab-ASPP net and network with DGCM

Table 2 Results on PASCAL-Context val set

NetWork	Context Pparams	Runtime (ms/image)	mIOU
VGG16			
Deeplab-LargeFOV	5.8M	83.0	37.66
FCN-8s [20]	16.7M	91.9	37.80
Deeplab-ASPP	23.1M	101.9	38.55
DGCM(ours)	**8.1M**	**86.1**	38.20
BiDGCM(ours)	16.2M	94.1	**38.69**

Bold are used to highlight the results from our proposed method

the network with dilated parameter (4,6). As shown in Table 2, Our network has the best performance with fewer parameters.

5 Conclusion

In this work, we visualized the ERF of different networks. We proposed the concept of ERFI which can better reflect the relevance of ERF and semantic segmentation performance. A Dense Global Context Module is designed to make ERF size larger and ERFI denser, which has the advantage of less network parameters and higher performance. In future work, we hope to explore shortening the depth of network by reducing base convolution layers, through the control of receptive field.

Acknowledgments Prof. Liu is supported by the Major Projects entrusted by the National Social Science Fund of China (Under Grant 16ZH017A3) and Program for Changjiang Scholars and Innovative Research Team in University (PCSIRT) granted by Ministry of Education, China. Dr. Han is supported by the NSFC (Under Grant U1509206,61472276)

References

1. Badrinarayanan V, Kendall A, Cipolla R (2015) Segnet: A deep convolutional encoder-decoder architecture for image segmentation. arXiv:151100561
2. Baehrens D, Schroeter T, Harmeling S, Kawanabe M, Hansen K, MÃžller KR (2010) How to explain individual classification decisions. J Mach Learn Res 11(Jun):1803–1831
3. Chen LC, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2016a) Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. arXiv:160600915
4. Chen LC, Yang Y, Wang J, Xu W, Yuille AL (2016b) Attention to scale: Scale-aware semantic image segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3640–3649
5. Chen LC, Papandreou G, Schroff F, Adam H (2017) Rethinking atrous convolution for semantic image segmentation. arXiv:170605587
6. Eigen D, Fergus R (2015) Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: Proceedings of the IEEE International Conference on Computer Vision, pp 2650–2658
7. Everingham M, Eslami SA, Van Gool L, Williams CK, Winn J, Zisserman A (2015) The pascal visual object classes challenge: A retrospective. Int J Comput Vis 111(1):98–136
8. Farabet C, Couprie C, Najman L, LeCun Y (2013) Learning hierarchical features for scene labeling. IEEE Trans Pattern Anal Mach Intell 35(8):1915–1929
9. Ghiasi G, Fowlkes CC (2016) Laplacian reconstruction and refinement for semantic segmentation. CoRR, arXiv:1605022641

10. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 580–587
11. Hariharan B, Arbeláez P, Bourdev L, Maji S, Malik J (2011) Semantic contours from inverse detectors. In: IEEE International Conference on Computer Vision (ICCV), 2011. IEEE, pp 991–998
12. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 770–778
13. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: Convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM international conference on Multimedia, ACM, pp 675–678
14. Koltun V (2011) Efficient inference in fully connected crfs with gaussian edge potentials. Adv Neural Inf Process Syst 2(3):4
15. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
16. Lin G, Milan A, Shen C, Reid I (2016a) Refinenet: Multi-path refinement networks with identity mappings for high-resolution semantic segmentation. arXiv:161106612
17. Lin G, Shen C, van den Hengel A, Reid I (2016b) 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
18. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014) Microsoft coco: Common objects in context. In: European Conference on Computer Vision. Springer, pp 740–755
19. Liu Z, Li X, Luo P, Loy CC, Tang X (2015) Semantic image segmentation via deep parsing network. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1377–1385
20. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3431–3440
21. Long JL, Zhang N, Darrell T (2014) Do convnets learn correspondence? In: Advances in Neural Information Processing Systems, pp 1601–1609
22. Luo W, Li Y, Urtasun R, Zemel R (2016) Understanding the effective receptive field in deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp 4898–4906
23. Mottaghi R, Chen X, Liu X, Cho NG, Lee SW, Fidler S, Urtasun R, Yuille A (2014) The role of context for object detection and semantic segmentation in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 891–898
24. Noh H, Hong S, Han B (2015) Learning deconvolution network for semantic segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1520–1528
25. Oquab M, Bottou L, Laptev I, Sivic J (2015) Is object localization for free?-weakly-supervised learning with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 685–694
26. Papandreou G, Chen LC, Murphy KP, Yuille AL (2015) Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1742–1750
27. Peng C, Zhang X, Yu G, Luo G, Sun J (2017) Large kernel matters–improve semantic segmentation by global convolutional network. arXiv:170302719
28. Pinheiro P, Collobert R (2014) Recurrent convolutional neural networks for scene labeling. In: International Conference on Machine Learning, pp 82–90
29. Pohlen T, Hermans A, Mathias M, Leibe B (2016) Full-resolution residual networks for semantic segmentation in street scenes. arXiv:161108323
30. Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, pp 234–241
31. Shimoda W, Yanai K (2016) Distinct class-specific saliency maps for weakly supervised semantic segmentation. In: European Conference on Computer Vision, Springer, pp 218–234
32. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition
33. Simonyan K, Vedaldi A, Zisserman A (2013) Deep inside convolutional networks: Visualising image classification models and saliency maps. arXiv:13126034
34. Tompson JJ, Jain A, LeCun Y, Bregler C (2014) Joint training of a convolutional network and a graphical model for human pose estimation. In: Advances in neural information processing systems, pp 1799–1807

35. Yu F, Koltun V (2015) Multi-scale context aggregation by dilated convolutions. arXiv:151107122
36. Zhao H, Shi J, Qi X, Wang X, Jia J (2017) Pyramid scene parsing network. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition
37. Zheng S, Jayasumana S, Romera-Paredes B, Vineet V, Su Z, Du D, Huang C, Torr PH (2015) Conditional random fields as recurrent neural networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1529–1537
38. Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A (2014) Object detectors emerge in deep scene cnns. arXiv:14126856

Yongge Liu is currently a Professor of Anyang Normal University, China, and the head of the Innovation Team granted by Ministry of Education, China. He received the Master's degree from Northwestern Polytechnical University, China. From 2012 to 2013, he visited the University of California at Los Angeles as a visiting scholar. His current research interests include Oracle Bone inscription information processing and multimedia analysis.

Jianzhuang Yu is currently a graduate student with the School of Computer Science and Technology, Tianjin University, China.

Yahong Han received the Ph.D. degree from Zhejiang University, Hangzhou, China. He is currently a Professor with the School of Computer Science and Technology, Tianjin University, Tianjin, China. His current research interests include multimedia analysis, computer vision, and machine learning.