Improving Residual Block for Semantic Image Segmentation

Fei Liu[1,2], Jing Liu[1 *], Jun Fu[1,2], Hanqing Lu[1]
[1]*National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences*
[2]*University of Chinese Academy of Sciences*
liufei2017@ia.ac.cn, jliu@nlpr.ia.ac.cn, jun.fu@nlpr.ia.ac.cn, luhq@nlpr.ia.ac.cn

***Abstract*—Currently, most state-of-the-art semantic segmentation methods employ residual network as base network. Residual network is composed of residual blocks. In this paper, we present an improved residual block called pyramid residual block to explicitly exploit context information and enhance useful features. In contrast to the standard residual block, the proposed pyramid residual block contains two newly added components: pyramid pooling module and attention mechanism. The former aggregates different-region-based context information. And the latter is able to adaptively re-calibrate feature responses through element-wise multiplication operation, thus enhancing useful features and suppressing less useful ones. Our proposed pyramid residual block demonstrates outstanding performance in PASCAL VOC 2012 segmentation datasets, and improve the segmentation accuracy by a large margin over the standard residual block.**

***Index Terms*—semantic segmentation, residual block, context information, attention mechanism**

I. Introduction

Semantic image segmentation is a fundamental topic in the field of computer vision. The task aims to assign every single pixel in the image a category label. In recent years, fully convolutional networks [13], which are adapted from the classification networks through replacing the last few fully connected layers by convolutional layers to output score maps instead of classification scores, have been broadly applied in pixel-wise semantic segmentation tasks, making remarkable progress due to the integrated multi-level hierarchical features and end-to-end trainable frameworks.

Since the pioneering deep CNN, AlexNet [11], won the first place in ILSVRC-2012, many efforts have been made to construct more powerful CNNs [8] [16] by varying the depth and breadth of network architectures. Learning more discriminative features through increasingly more layers of feature representation has shown to be very effective on ILSVRC object classification tasks. The field of semantic segmentation also benefits from deeper architectures of fully convolutional networks. Remarkable advances [4] in mean Intersection-over-Union (mIoU) scores on PASCAL VOC 2012 dataset [6] were reported when the 101-layer ResNet [8] model replaced the 16-layer VGG-16 model [16]; using 152-layer ResNet model yields further improvements. ResNet is specially remarkable due to its depth and the introduction of residual blocks. The residual blocks are helpful for overcoming the vanishing gradients problem by introducing identity skip connections. Currently, most state-of-the-art semantic segmentation methods employ ResNet as building blocks for segmentation architectures. In this paper we devote to improving the residual block of ResNet, thus obtaining improvement on the segmentation accuracy. We describe in detail our proposed pyramid residual block in Section III.

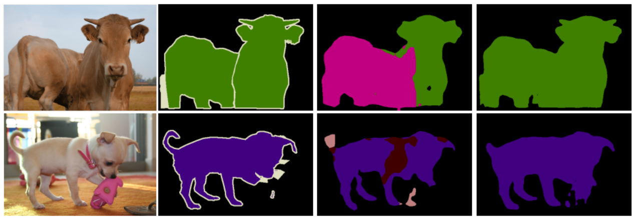

Fig. 1. Our method adds more context information to help clarify the local confusion. From left to right: input image, ground truth, the output of standard ResNet, our output.

For the task of semantic image segmentation, more context information results in more accurate segmentation results. The standard ResNet [8] has the capacity to capture context information, but there is still room for improvement. As shown in Fig. 1, due to the lack of enough context information for large objects, the standard ResNet model [8] confuses cow with horse within a local area on the top row. However, by adding more context information, it can greatly help correctly classify each pixel in the image. Several approaches [15] [21], introducing contextual information, have been proposed. Although these methods can mitigate the issues caused by the absence of context information, they require to build extra contextual modules on the top of encoder networks, by which context features are incorporated. As a result, the network architecture can become exorbitantly complex and large-scale hyperparameter tuning is often wasteful of time and resources. Different from the forementioned methods, we introduce pyramid pooling module into residual block to effectively exploit context from regions of various sizes, thus generating more reliable predictions (see Fig. 2 and Section III A for details).

For each convolutional layer, a set of filters are expected to extract informative features. However, not all elements of feature responses have the same contribution. For those feature responses that contribute more, we should enhance them. Otherwise, we should suppress them. Based on this idea, we embed attention mechanism into the standard residual block.

*Corresponding author

978-1-5386-5321-0/18/$31.00 ©2018 IEEE

Specifically, we use the *sigmoid* function and convolutional layer to learn an attention probability distribution from the input feature maps. Then an element-wise product operation between the feature maps and the attention probability maps is performed, thus selectively emphasising useful features and suppressing less useful ones.

We improve the conventional residual block, making it better serve the need of pixel-level semantic segmentation. The technical details will be discussed in Section III. Extensive experiments in Section IV demonstrate the effectivity of the pyramid residual block. The main contributions of this paper are as follows:

- We introduce the pyramid pooling module into standard residual block to gather multi-scale context information.
- An attention mechanism is embedded into the standard residual block in order to enhance beneficial features.
- Our proposed pyramid residual block is simple yet effective. It is not limited to the current architecture or tasks, while should be a generalized method that can be applied to other architectures or tasks based on ResNet.

II. Related Work

In the field of semantic segmentation, many efforts have been made to collect context information more effectively. DeepLab-V2 [4] proposed dilated convolution with variable dilation rate to enlarge local receptive fields. The method of [5] supplemented global context information by adding global pooling operation. IFCN [15] proposed a method of capturing context information through stacked contextual module. Pyramid pooling module was introduced by PSPNet [21] to model context information of feature maps. The key component of pyramid pooling module is multi-level pooling with pooling kernels of different sizes. Our proposed method (i.e. pyramid residual block) also uses pyramid pooling module, but is different from the method of [21]. The pyramid pooling module of [21] is attached at the top of ResNet, while ours is embedded into residual blocks of ResNet. Besides, there are some differences in the technical details, for example, our pyramid pooling module uses *sum* instead of *concat* operation applied in PSPNet.

Attention has been shown to improve performance in many visual tasks [2] [9]. In recent work, the SENet proposed by *Hu et al.* [9] presented an attention mechanism which is similar to ours. They also added a new branch to learn the attention probability distribution in the standard residual block. However, there are two main differences between the two attention mechanisms. First, the attention mechanism of SENet focuses on channels, achieved by channel-wise multiplication operation, while ours focuses on channels as well as spatial, achieved by element-wise multiplication operation. Second, theirs is designed for image classification, while ours is designed for semantic segmentation.

III. Method

Compared to the standard residual block, our proposed pyramid residual block contains two newly added components: pyramid pooling module and attention mechanism. In this section, we provide a detailed description of the two components and overall method.

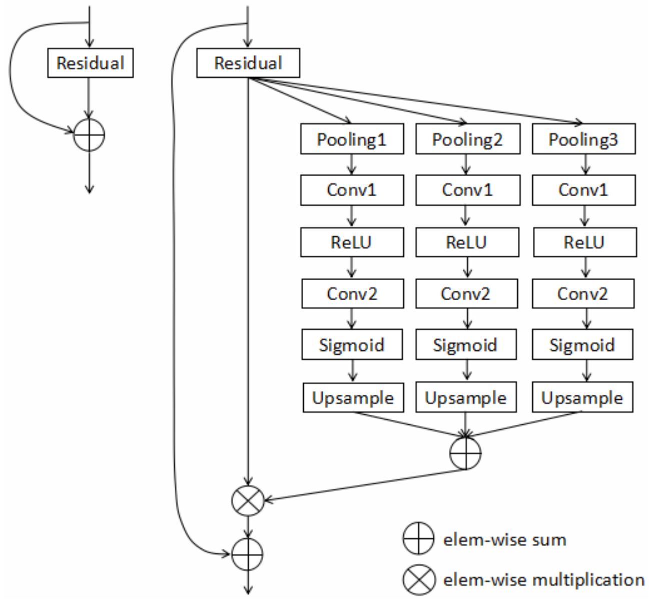

Fig. 2. The schema of the standard resisual block (left) and the pyramid residual block (right).

A. Pyramid Pooling Module

Context information [20] [12] is known to be very useful for improving the segmentation performance. To be concrete, fine-grained or local information helps to achieve good pixel-level accuracy, and global context of the image is able to clarify local ambiguities. Besides, in a deep neural network, although theoretically features from higher level layers already have very large receptive fields that are beyond the size of input image, in practice the size of receptive fields is much smaller than the theoretical one, as shown in [23]. This prevents the segmentation networks from sufficiently incorporating the significant context information. To address this issue, we introduce the pyramid pooling module into the standard residual block.

As illustrated in Fig. 2, the pyramid pooling module takes the feature maps from the main branch of residual block as inputs. It has three parallel branches and each branch contains one pooling layer, one ReLU activation function, two convolutional layers (1×1 convolution), one sigmoid function and one upsampling operation (i.e. bilinear interpolation). The bin sizes of pooling in the three branches are set to 1×1, 2×2 and 4×4 respectively. And we adopt the average pooling operation. The first convolutional layer after pooling operation is used for dimensionality reduction. The second convolutional layer after pooling operation is used for dimensionality increase, recovering the original channel number. After upsampling operation, the feature maps from three parallel branches are fused via element-wise sum, resulting in the attention probability maps.

TABLE I
PERFORMANCE COMPARISONS OF DIFFERENT DIMENSIONALITY REDUCTION SETTINGS. 'DIMENSIONALITY' DENOTES THE NUMBER OF CHANNELS IN THE DIMENSIONALITY-REDUCTION LAYER.

Network	**Dimensionality**	**mIoU(%)**
baseline	-	64.45
ResNet-50	1024	69.70
ResNet-50	512	69.96
ResNet-50	256	70.24
ResNet-50	128	68.88

TABLE II
PERFORMANCE COMPARISONS OF DIFFERENT LEVELS OF POOLING.

Network	**Dimensionality**	**# the levels of pooling**	**mIoU(%)**
baseline	-	-	64.45
ResNet-50	256	1	67.52
ResNet-50	256	2	68.74
ResNet-50	256	3	70.24
ResNet-50	256	4	70.88

B. Attention Mechanism

We expect to enhance useful features and suppress less useful ones. To fulfill this, we embed the attention mechanism into residual block. In each branch of pyramid pooling module, for the activation function after the second convolution, we choose the sigmoid function instead of the frequently used ReLU function to normalize every element of feature maps. The pyramid pooling module outputs the attention probability maps, and then an element-wise multiplication operation is performed between the feature maps from the main branch of residual block and the attention probability maps. In this simple way, greater weight is assigned to more useful feature response, helping to boost feature discriminability.

C. Pyramid Residual Block

We integrate the pyramid pooling module and attention mechanism into the standard residual block, forming our pyramid residual block. As illustrated in Fig. 2, the feature maps from the main branch of residual block pass through the pyramid pooling module, obtaining the attention probability maps, and then the feature maps and attention probability maps perform element-wise multiplication.

The proposed pyramid residual block can naturally replace the standard block of ResNet framework. It not only retains the primary advantage of residual learning [8] but also supplements aggregated context information and adaptive feature recalibration. In addition, it is designed for end-to-end learning; thus the pyramid pooling module and feature re-calibration can be optimized jointly.

IV. EXPERIMENTS

We carry out comprehensive experiments on PASCAL VOC 2012 dataset [6]. Experimental results demonstrate that our proposed pyramid residual block is able to bring a significant improvement on the performance of the network.

TABLE III
EVALUATION ON THE EFFECTIVITY OF SINGLE COMPONENT.

Method	**mIoU(%)**
None (baseline)	64.45
Only pyramid pooling module	68.50
Only attention mechanism	67.34
Ours	70.24

TABLE IV
PERFORMANCE COMPARISONS OF RESNET-50 AND RESNET-101 WITH MORE PYRAMID RESIDUAL BLOCKS ON PASCAL VOC 2012 VAL SET.

Network	**Usage of pyramid residual block (# blocks)**	**mIoU(%)**
ResNet-50	none (0, baseline)	64.45
ResNet-50	res5c (1)	70.24
ResNet-50	res5c & res4f (2)	71.15
ResNet-50	res5c & res4f,c (3)	71.86
ResNet-101	none (0, baseline)	71.27
ResNet-101	res5c (1)	74.24
ResNet-101	res5c & res4b22 (2)	74.90
ResNet-101	res5c & res4b22,19 (3)	75.38

A. Implementation

Our implementation is based on the public Caffe [10] framework. Weight decay and momentum parameters are set to 0.0005 and 0.9 respectively. Similar to [12], we adopt the poly learning rate strategy where current learning rate equals to the initial one multiplying $(1 - \frac{iter}{maxiter})^{power}$. We set the initial learning rate to 0.00025 and power to 0.9. We train our models using stochastic gradient descent (SGD) with a batch size of 10 and the maximum number of 20K iterations. For data augmentation, we just employ random mirror and random cropping for all training images. We use ResNet-50 or ResNet-101 [8] networks that have been pretrained on the ImageNet dataset as our base models. Here, we remove the last downsampling operation and adopt dilated convolution, resulting in a downsampling factor of 16. The output from the last residual block is processed by a 1×1 convolutional layer and softmax nonlinearity to produce the final pixel-wise segmentation result.

B. Dataset and Measure

PASCAL VOC 2012 [6] is the most widely used dataset for semantic segmentation. It contains 20 object categories plus one background class. We augment the data with the extra annotations provided by [7], resulting in 10582, 1449 and 1456 images for training, validation and testing, respectively. The performance is measured by pixel intersection-over-union (IoU) averaged across the 21 classes.

C. Ablation Study

Dimensionality Reduction. To explore the influence of dimensionality reduction on performance, we replace the last residual block (res5c) of ResNet-50 with pyramid residual block, and the channel number of dimensionality-reduction layer (i.e. 'Conv1' in Fig. 2) are set to 1024, 512, 256 and 128, respectively. The results are listed in TABLE I. When

TABLE V
COMPARISONS OF MODEL SIZE AND TESTING TIME.

Model	Model size (*M*)	Testing time (*s*)	mIoU(%)
Baseline	165.48	1.252	71.27
Ours (one)	178.97	1.288	74.24
Ours (two)	191.36	1.319	74.90
Ours (three)	202.74	1.348	75.38
Ours (three) + Aug	202.74	1.348	**76.32**
DeepLab-V2 [4]	517.64	2.986	74.54

TABLE VI
COMPARISON WITH OTHER EXCELLENT METHODS ON PASCAL VOC 2012 TEST SET. AUG: DATA AUGMENTATION BY RANDOMLY RESCALING INPUTS.

Method	Aug	CRF	mIoU(%)
CRF-RNN [22]		√	74.7
Context + CRF-RNN [20]		√	75.3
DT-EdgeNet [3]		√	76.3
H-ReNet + DenseCRF [19]	√	√	76.8
Oxford_TVG_HO_CRF [1]		√	77.9
Context + Guidance CRF [14]		√	78.1
Adelaide_VeryDeep_FCN [18]	√		79.1
DeepLab-V2 [4]	√	√	79.7
Ours			80.54
Ours	√		**82.35**

adopting the dimensionality-reduction layer with 256 channels, the model attains performance of 70.24% and outperforms the baseline by an absolute improvement of 5.79%. Compressing the dimensionality of features can increase the segmentation accuracy in a certain degree; this may be caused by the removal of redundancies in features. However, when the dimensionality is reduced too much, some discriminative features may be lossed, as shown in the fifth row of TABLE I.

Multi-level Pooling. Our proposed pyramid residual block adopts three-level pooling with bin sizes of 1×1, 2×2 and 4×4 respectively. We also construct the models which adopt one-level pooling (i.e. only global pooling), two-level pooling (with bin sizes of 1×1 and 2×2) and four-level pooling (with bin sizes of 1×1, 2×2, 4×4 and 6×6). The comparative results are listed in TABLE II. When we increase the levels of pooling, the performance is constantly improved, from 67.52% to 70.88%. More levels of pooling operations with various bin sizes are able to obtain richer context information, thus resulting in higher mIoU. However, it also leads to more computational stress and harder optimization, so we adopts three-level pooling as more levels of pooling bring too slight improvements.

The Effectivity of Single Component. Our proposed pyramid residual block contains two newly added components, pyramid pooling module and attention mechanism. To evaluate the effectivity of single component, we retain one of the two components and remove the another. Specifically, when retaining attention mechanism, we replace pyramid pooling module with one *conv-relu* and one *conv-sigmoid*, while with the same number of parameters. When retaining pyramid pooling module, we remove element-wise multiplication, and add the feature maps from the pyramid pooling module to the feature maps from the identity branch of residual block. The experimental results are presented in TABLE III. With only pyramid pooling module, the model improves mIoU by 4.05% over the baseline. Adding attention mechanism brings an improvement of 2.89% on mIoU over the baseline. Our method combines the pyramid pooling module and attention mechanism, improving mIoU by 5.55% over the baseline.

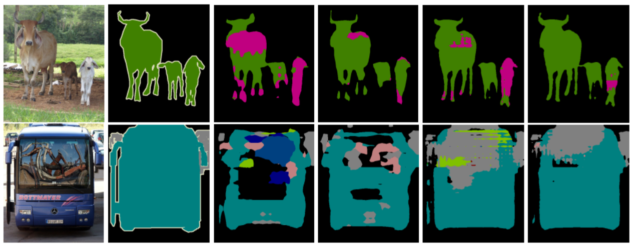

Fig. 3. Using more pyramid residual blocks makes the results more accurate. From left to right: input image, ground truth, baseline model, the model with one pyramid residual block, the model with two pyramid residual blocks, and the model with three pyramid residual blocks.

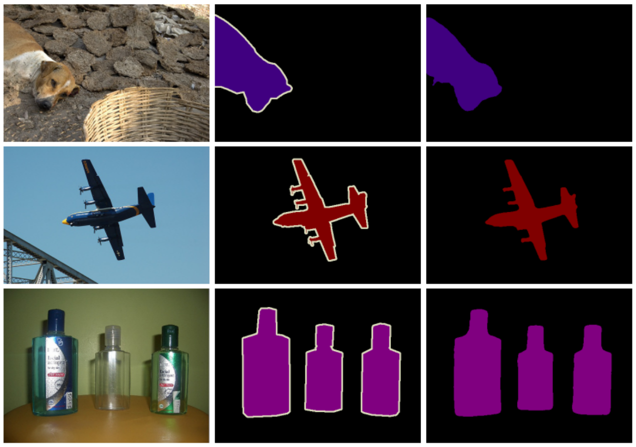

Fig. 4. From left to right: input image, ground truth and semantic segmentation results.

More Pyramid Residual Blocks. Since ResNet-50 [8] with one pyramid residual block (res5c) largely outperforms the baseline model, we experiment with more residual blocks. We replace two blocks (res5c and res4f) and three blocks (res5c, res4f and res4c) of the original ResNet-50 with pyramid residual block, respectively. As shown in TABLE IV, the model with two pyramid residual blocks brings 0.91% improvement over the one with only one pyramid residual block, the model with three pyramid residual blocks further improves the performance by 0.71%. Therefore, our proposed pyramid residual block is simple yet effective for improving semantic segmentation systems based on ResNet. We believe that the segmentation performance can be further improved when more pyramid residual blocks are used. Some visual results are shown in Fig. 3.

Deeper ResNet. Previous works [17] [4] have shown that deeper neural networks are beneficial to pixel-wise semantic segmentation due to more powerful capability of modeling feature representation. We adopt pre-trained ResNet-101 with the same modification as ResNet-50. In the same way, one block (res5c), two blocks (res5c and res4b22), and three blocks (res5c, res4b22 and res4b19) are replaced by pyramid residual block, respectively. The experimental results are showed in TABLE IV. We can see that ResNet-101 largely outperforms ResNet-50 under the same settings. Besides, when the number of the replaced blocks increases, the performance improves from 71.27% to 75.38%. Deeper ResNet also benefits from more pyramid residual blocks. Some semantic segmentation results of our ResNet-101 model with three pyramid residual blocks are shown in Fig. 4.

Model Size and Testing Time. We compare our ResNet-101 model with DeepLab-V2 [4] in terms of model size and testing time. The time cost is measured on one GTX TITAN X GPU. The comparative results are shown in TABLE V. In the first column, the number in parentheses denotes the number of pyramid residual blocks. ”Aug” denotes data augmentation by randomly rescaling inputs. When we use more pyramid residual blocks to replace the standard residual blocks, the model size and testing time also increase accordingly. Our ResNet-101 model with three pyramid residual blocks (+ Aug) outperforms DeepLab-V2 by an absolute improvement of 1.78%, but the model size is only about 40% of that of DeepLab-V2 and the testing time is only about 45% of that of DeepLab-V2.

Comparison with Other Methods. We also compare our method with other excellent methods on PASCAL VOC 2012 test set. Specifically, we fine tune our best model on PASCAL VOC 2012 trainval set, and submit our test results to the official evaluation server. The results are shown in TABLE VI. Other methods employ some common strategies (e.g. data augmentation by randomly rescaling inputs, CRF) to enhance the segmentation performance. However, our model attains the performance of 80.54% without using any strategies. When we use data augmentation by randomly rescaling inputs during training phase, the performance is further improved to 82.35%.

V. Conclusion

We have proposed a pyramid residual block for semantic segmentation. Our block consists of two key components, pyramid pooling module and attention mechanism. The former provides context information from regions of different sizes and the latter selectively enhances useful features. Experiments on the PASCAL VOC 2012 datasets show that our method significantly improves the segmentation performance.

Acknowledgment

This work was supported by National Natural Science Foundation of China (61332016 and 61472422).

References

[1] A. Arnab, S. Jayasumana, S. Zheng, and P. H. Torr, “Higher order potentials in end-to-end trainable conditional random fields,” *CoRR, abs/1511.08119*, vol. 4, p. 8, 2015.

[2] C. Cao, X. Liu, Y. Yang, Y. Yu, J. Wang, Z. Wang, Y. Huang, L. Wang, C. Huang, W. Xu *et al.*, “Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2956–2964.

[3] L.-C. Chen, J. T. Barron, G. Papandreou, K. Murphy, and A. L. Yuille, “Semantic image segmentation with task-specific edge detection using cnns and a discriminatively trained domain transform,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4545–4554.

[4] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs,” *arXiv preprint arXiv:1606.00915*, 2016.

[5] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, “Rethinking atrous convolution for semantic image segmentation,” *arXiv preprint arXiv:1706.05587*, 2017.

[6] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes challenge: A retrospective,” *International journal of computer vision*, vol. 111, no. 1, pp. 98–136, 2015.

[7] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, and J. Malik, “Semantic contours from inverse detectors,” in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 991–998.

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

[9] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” *arXiv preprint arXiv:1709.01507*, 2017.

[10] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 675–678.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[12] W. Liu, A. Rabinovich, and A. C. Berg, “Parsenet: Looking wider to see better,” *arXiv preprint arXiv:1506.04579*, 2015.

[13] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[14] F. Shen and G. Zeng, “Fast semantic image segmentation with high order context and guided filtering,” *arXiv preprint arXiv:1605.04068*, 2016.

[15] B. Shuai, T. Liu, and G. Wang, “Improving fully convolution network for semantic segmentation,” *arXiv preprint arXiv:1611.08986*, 2016.

[16] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[17] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell, “Understanding convolution for semantic segmentation,” *arXiv preprint arXiv:1702.08502*, 2017.

[18] Z. Wu, C. Shen, and A. v. d. Hengel, “Bridging category-level and instance-level semantic image segmentation,” *arXiv preprint arXiv:1605.06885*, 2016.

[19] Z. Yan, H. Zhang, Y. Jia, T. Breuel, and Y. Yu, “Combining the best of convolutional layers and recurrent layers: A hybrid network for semantic segmentation,” *arXiv preprint arXiv:1603.04871*, 2016.

[20] F. Yu and V. Koltun, “Multi-scale context aggregation by dilated convolutions,” *arXiv preprint arXiv:1511.07122*, 2015.

[21] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” *arXiv preprint arXiv:1612.01105*, 2016.

[22] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, “Conditional random fields as recurrent neural networks,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1529–1537.

[23] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, “Object detectors emerge in deep scene cnns,” *arXiv preprint arXiv:1412.6856*, 2014.