CANNET: CONTEXT AWARE NONLOCAL CONVOLUTIONAL NETWORKS FOR SEMANTIC IMAGE SEGMENTATION

Lingyan Ran, Yanning Zhang

School of Computer Science
Northwestern Polytechnical University
Xian, Shaanxi, China

Gang Hua[*]

Department of Computer Science
Stevens Institute of Technology
Hoboken, NJ, USA

ABSTRACT

Semantic segmentation has long been a hot topic, most methods are the region based method, which lost connection information to their neighbors. In this paper we propose to encode context information into convolutional networks on this semantic labeling task. Firstly, we propose the nonlocal convolution kernel, which extracts feature from larger neighbor regions without introducing more parameters. Then we build up a context aware module, which takes both local patch and nonlocal neighbor information into account. At last we embed the module into convolutional networks and tested the improvement on benchmark datasets.

Index Terms— Semantic segmentation, sparse kernel, context aware module

1. INTRODUCTION

Semantic segmentation has long been a hot topic and a companying task with object detection and recognition. It aims at pixel-wise classification of images and generating meaningful partitioning areas with objects and scene labels. Many methods have gained great success in the past a few years. Most of them are region based supervised learning methods. These learning methods try to find relevant image features that can help classify regions into different categories.

The state-of-the-art methods like RCNN [1] and SDS[2] are all those region-based method. They first partition input images into regions with candidate extraction method like CPMC [3], MCG [4], and selective search [5]. Then extract features using convolutional neural networks (CNN) on those candidate regions. Further, classifiers such as SVM are trained on those features to generate class-specific strong classifiers. At last, assign labels to those regions and we can get the semantic meaning of our input image.

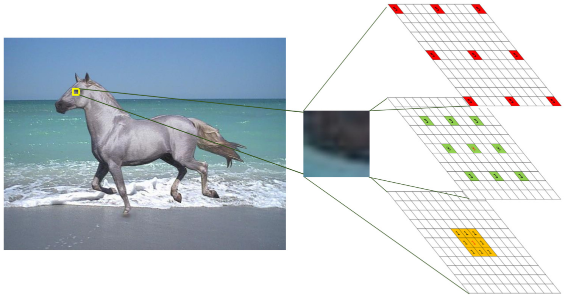

Fig. 1: The nonlocal convolutional filter, compared with the local filters, our proposed filter can sample a much larger range and hence give better performance without introducing more parameters. Grids illustrate the proposed nonlocal kernel with stride 1(i.e. a local kernel), stride 3 and stride 5.

Region based methods work very well on consistent object patches but not for objects with occlusions or objects unexpectedly grouped in candidate generation procedure. Texture or context based methods, on the other hand, can handle this case. For the region based method, when extracting features from a region, we are losing the occurrence information of their neighbors. Context based methods like [6, 7] are proposed under that consideration.

One basic fact is that images, especially those captured by photographers, always contain not only one specific object, but also some accompanies and natural scenery. Context based methods then make judgment from both region of interest and its neighbors. For example, one region of horse is most likely accompanied by a grass region other than a road region. Methods like autocontext [6] are taking advantage of those neighboring pixels in prediction. Their performance can be further boosted with more discriminative features.

Recently, there is a rapidly growing literature on convolutional networks, which demonstrates its great capability in extracting meaningful features, especially in the task of object recognition [8, 9]. Though it has also achieved state-of-the-art

[*]This work is conducted when Lingyan Ran is a Visiting Ph.D. student at Stevens Institute of Technology. Dr. Gang Hua was partly supported by US National Science Foundation Grant IIS 1350763. Lingyan Ran was partly supported by the National Natural Science Foundation of China (No.61231016(Key Project)), NPU New People and New Directions Foundation (No.13GH014604), The Fundamental Research Funds for the Central Universities (No. 3102015), Soaring Star of NPU (No.12GH0311), NPU Foundation for Fundamental Research (NPU-FFRJCT20130109), and Foundation of China Scholarship Council (No.201303070083, No.201306290090).

performance on other tasks like object detection [8, 10, 11], pose estimation [12, 13], and scene parsing [14], not that much work has been done on the task of pixel level object segmentation. For example, RCNN [1] uses convolutional network at the patch level, which in fact is a patch recognition problem. In this paper, we attempt to build a context aware convolutional network and focus on pixel-wise image segmentation task.

The receptive field is an essential factor to a convolutional network's performance. To maintain good accuracy, one straightforward thought is to increase the convolutional filter size, which, may not give good results in practice. Large filter size does not necessarily leads to more discriminative models and it is easy to over-fitting because of the co-adaption problem. That is why Dropout [15] is adopted in enhancing the generalization of deep learning models. Another way to achieve that goal is to build a deeper network [16]. Despite their cheering performance, the computational cost will be significantly increased. Our work is based on a new way of increasing the representation power of the neural network, by first adding one 1×1 convolutional layer [17], and further build more complex network structures [8].

Our work makes three key contributions: (1) we define a nonlocal convolutional kernel, which is very easy to implement and embed into existing networks. This sparse kernel gives neurons a larger receptive field without increasing the number of parameters. (2) we design a context aware module, this module adopts both local and nonlocal convolutional kernels together. This module have the capacity to sample both local neighbor pixels and nonlocal ones. Moreover, by parallel local and nonlocal kernel together as a pyramid, this module has the potential to deal with scale various problems. (3) we build a convolutional network with this context aware module and apply it to the problem of image segmentation. Experiments on three publicly available datasets demonstrate the good performance of our proposed models.

2. NONLOCAL CONVOLUTIONAL NETWORK

In this section, we give detailed description on how we design the nonlocal convolutional kernel, the context aware module, and the entire network architecture.

Nonlocal Convolutional Kernel. Convolutional kernel is the basis of CNN. A conventional convolutional filter is represented as Equation 1. For a confidence f, its value is the nonlinear warping result of a receptive field $x \in X$. x has the same size as the filter, which is $w \times h$. Function σ is an activation function and it can be, for example, a Sigmoid, Tanh, or ReLU. Each filter is working on a consistent patch, we call it the local kernel.

$$f = \sigma(\sum_{i=1}^{w}\sum_{j=1}^{h}\omega_{ij} \cdot x_{ij} + b) \tag{1}$$

A convolutional layer generates feature maps by warping a region with a linear filter kernel and then feed them into a nonlinear activation function. Each single value in the feature map is an activation response of one receptive field in the previous layer. In general, a larger receptive field would result in a better performance after filter warping. However, it also means more parameters, and hence more prone to overfitting. That is why Dropout [15] and Rectified Units are designed, to maintain that only part of those nodes are optimized each time. One interesting question to ask is, could we maintain the same level of performance with limited number of parameters and yet keep a relatively large field?

To solve the problem, we propose the nonlocal convolutional kernel. That is, for a given region, we are not using all those pixels as input but just choosing part of them with a given pattern. This filter pattern samples from part of the receptive field and combine them together with a linear function before feeding to a nonlinear activation function, which works similar as conventional local convolutional layers.

Eq. 2 shows the nonlocal convolutional kernel with the same number of parameters as Eq. 1. The parameter s is a step size chosen by hand, which determines the sparsity of a model. σ is also a non-linear warping function. Here we are using the sigmoid function for the confidence map, where the output values are in the range of (0,1).

$$f^{'} = \sigma(\sum_{i=1}^{w}\sum_{j=1}^{h}\omega^{'}_{i,j} \cdot x_{i\times s,j\times s} + b^{'}) \tag{2}$$

While we are using the same number of parameters, the receptive field of our kernel would be $ws\times hs$. In other words, we could now sampling on a s times larger receptive field than a local kernel. This, as demonstrated in later experiments, can contribute a lot on enhancing the network's performance.

Concatenate Context Aware Module. Another property of a convolutional network is that it dose not sample from one single image channel, but from a set of feature maps. And those feature maps are all filtering results of previous ones. In this section, we build a module that combines different sparse filters together, as show in Fig. 2.

The work of auto-context [6] gives a good example of how to learn an efficient and effective context model from a set of training images and their corresponding pixel-wise label maps. First, they would train a set of weak classifiers on local image patches and generate a set of confidence maps. Then sampling from those confidence maps to get the context feature. Combining both the context feature and the image patch feature together to train a new classifier. After that, use the new classification maps and again the image patches to get another classifier. This process is iteratively conducted until the algorithm converges to the ground truth. And at last, the budget of classifiers composes the auto-context segmenter.

Fig. 2 gives the structure of our module. In our proposed context aware network module, we use the same assumption

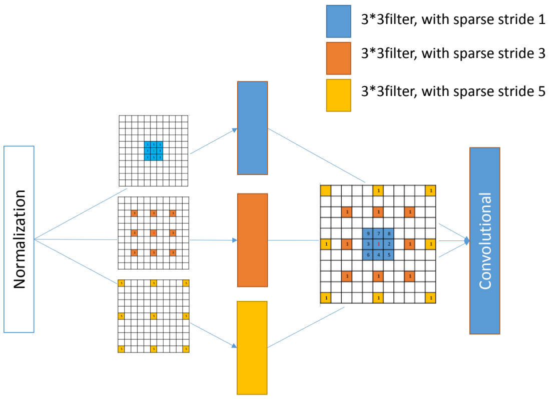
3*3filter, with sparse stride 1
3*3filter, with sparse stride 3
3*3filter, with sparse stride 5
Normalization
Convolutional

Fig. 2: The context aware module we use in our CanNet. It's composed of one local convolutional kernel (stride 1) and two nonlocal kernel (stride 3, stride 5). For a local only module with kernel size 3×3, the receptive field is 3×3, while in our module it is 11×11.

that classifiers trained on feature maps should also have the ability to describe the context information. This works very much like the "Inception" module proposed by Szegedy et al. in [8]. The input images would be first enhanced using algorithms like local contrast normalization. Then a local convolutional layer and two non-local convolutional layers are added in parallel. Those layers would extract features from a relatively sparse region. In our experiment, we use the stride of 1, 3, 5 respectively. At last, a convolutional layer is added again. This layer is for the combination of both local and nonlocal feature maps, and works in the same way as the NiN [17]. For each single confidence value in the module, it gets a expanded receptive field taking advantage of the nonlocal part.

Nonlocal Convolutional Network. Our context aware nonlocal convolutional network (CanNet) mainly contains 3 modules. The first module is a conventional convolutional layer followed by a max-pooling layer. The second and third module are the proposed context aware modules.

Our base layer is designed for the goal of image noise removal and feature enhancement. The following two stacked context aware modules work for extracting features and generating confidence maps. The very last layer outputs our prediction map and it's the same size as ground truth image.

Training the CanNet. Our objective is to get a confidence map from the network of whether one pixel in the image belongs to a specific object or not. For each pixel, it's confidence value is the probability of this pixel taking label l, while $l \in L$ is the labels in target space. We denote the value here as $p(y_i = l|x) \in [0, 1]$. This value then can be seen as a result of a regression problem. Given a set of training images $D = (x, y)^N$, with $x, y \in R^{W \times H}$, the problem now is to find a set of parameters Ω that can minimize the cross-entropy of the predicted value and the ground truth, see Equation 3.

F now is a set of nonlinear warping functions using the non-local structure as described in Equation 2.

$$\min_{\Omega} \sum_{(x,y)\in D} -F(x)\log y - (1 - F(x))\log(1 - y) \qquad (3)$$

Training our non-local deep network requires estimating all the weights and biases of each neuron with limited training dataset. We follow the pipeline of learning a convolutional network, i.e. first initialize the parameters with random values and then update them all with fine-tuning. During training, weights are updated using stochastic gradient descent method. As we know, this is really time consuming in reality, especially for tasks like deep neural networks with thousands of parameters.

Our network does not have that much convoluitonal layers like [8], so we do not take much considerations about the gradients vanishing problem. However, improving the training speed is still of interest. As described in [18], pre-training is very important for feature learning efficiency. Hence, we feed the idea of layer-wise training into our network as [19, 20] does.

Firstly, we train the network modules one by one by fixing its previous layers. The output of each module is bounded to the ground truth. Then, after each module is trained separately, we do a back-propagation on the whole network.

3. EXPERIMENTS

To verify the performance of our proposed non-local convolutional kernel and the context aware module, we run several tests on three publicly available datasets, i.e., the Weizmann horse dataset [21], the Graz02 dataset [22], and the PASCAL VOC 2012 segmentation dataset [23].

We test the comparative performance of our proposed CanNet and a conventional network with only local convolutional kernels, we quote it as 'CanNet-local' on those datasets. Both networks have the same number of parameters and layer-out structures. Moreover, they are initialized with the same weights.

Results on Horse Dataset. The Wizemann horse dataset consists of 328 horse images with pixel-wise annotated label maps. We randomly extract half of those images as training data and the rest half as the testing data. For comparison, we choose the equal F-measure as the evaluation metric, following what most work does, i.e.,

$$F_{1score} = \frac{2 \times Precision \times Recall}{(Precision + Recall)} \qquad (4)$$

Fig. 3 shows sample result from local only and nonlocal convolutional kernels. Notice that networks with only local convolutional kernels face a challenge problem that the horse trunk is not well detected. That's because this part does not have enough texture for the nets to give a good prediction. The nonlocal kernel, on the other hand, can handle this with support information from neighboring regions.

Table 1: Results on PASCAL VOC 2012 val. dataset on AP. We get a 5 percent improvement on average.

	aero	bike	bird	boat	botl	bus	car	cat	chair	cow	table	dog	horse	moto	pers	plant	sheep	sofa	train	tv	mAP
O_2P[24]	56.5	19.0	23.0	12.2	11.0	48.8	26.0	43.3	4.7	15.6	7.8	24.2	27.5	32.3	23.5	4.6	32.3	20.7	38.8	32.3	25.2
SDS[2]	68.4	49.4	**52.1**	32.8	**33.0**	67.8	**53.6**	**73.9**	19.9	43.7	25.7	60.6	55.9	58.9	56.7	28.5	55.6	32.1	64.7	**60.0**	49.7
CanNet	**71.6**	**55.2**	41.4	**47.4**	24.4	**73.3**	45.7	68.4	**24.9**	**71.9**	**50.2**	**65.5**	**69.7**	**66.9**	**59.6**	**39.7**	**70.0**	**39.5**	**73.7**	35.0	**54.7**

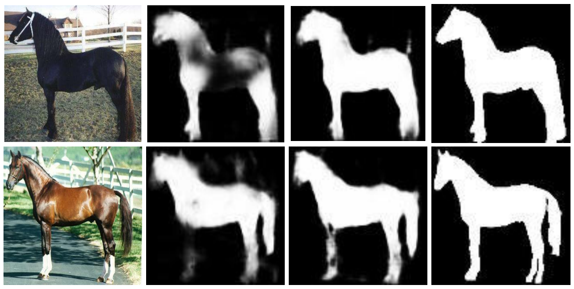

Fig. 3: Comparison results of conventional local kernel and our nonlocal kernel. Column #1 source images; #2 conventional kernel; #3 our kernel; #4 ground truth labels. The nonlocal kernel's vast sampling region maintains the performance improvement.

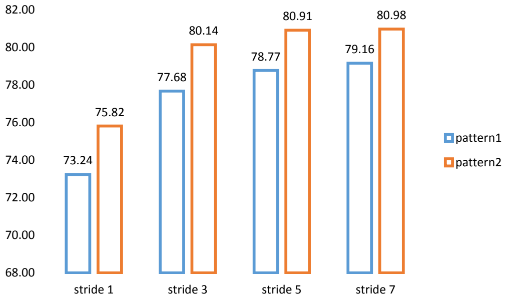

Fig. 4: Results on Horse dataset with various kernel and stride size using CanNet. Networks from pattern 1 and 2 are of different filter size 7*7, 11*11 respectively. Note that we can boost the performance with fixed number of parameters by just increasing stride size.

Table 2: Results on the Graz02 dataset in terms of F1score(%). Note that all those methods is confronted by bikes, which is a challenging category, but not CanNet nor CanNet-local. That results from the fact that convolutional filters are in fact edge or corner detectors. The more complicated an object, the better performance.

	Bikes	Cars	People	Avg.
Aldavert et al. [25]	58.6	62.9	71.9	64.5
CanNet-local	72.1	69.4	59.2	66.9
Kuettel et al. [26]	63.2	74.8	66.4	68.1
Fulkerson et al. [27]	66.1	72.2	72.2	70.2
CanNet	78.0	69.0	65.5	70.8
Lempitsky et al. [28]	83.7	84.9	82.5	83.7

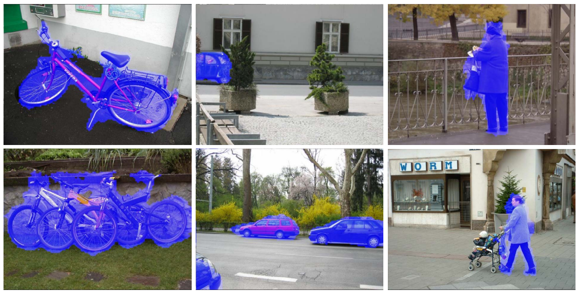

Fig. 5: Sample segmentation results from Graz02 dataset. From left, the segmentation of bikes, cars and people.

Fig. 4 presents the comparative tests. We have convolutional network with filter size of 7 × 7 and 11 × 11, we name those two networks as pattern 1 and pattern 2 respectively. It is easy to tell that local convolutional kernel (i.e. stride = 1) works bad on this segmentation task. And when we increase the filter sparsity stride size, we get a better performance.

Results on Graz Dataset. The Graz02 dataset is designed for testing scale and pose changes in visual recognition. In total, it has three object categories, including 365 images for bikes, 420 images for cars, and 311 images for people. Each object is marked out with a mask and some other annotations, such as occlusion information. In this work, we consider generating global context message of an object from the image, hence the occluded part is also marked as part of the object.

We use the predefined train-test protocol to specify the test and train sets. The results are presented in Table 2. In general, our method can get a relatively higher average value. And the nonlocal CanNet gives improvement over the CanNet-local. One thing to notice is that, for those traditional methods [25, 26, 27], the segmentation of bikes is the worst among those three tasks. Convolutional methods, both CanNet and CanNet-local, get the best on bikes other than the rest two. That demonstrate the fact that convolutional kernels preserve a good capacity in extracting edges, corners, and objects with rich textures. Fig. 5 shows some segmentation samples from our test result.

Results on PASCAL Dataset. In total the dataset has 20 object categories and 2913 images for the task of class segmentation. In practice, we are using the pre-splitted 1464 training images for training and 1449 validation images for testing.

Table 1 shows the detailed comparison result with two state-of-the-art semantic segmentation algorithms O_2P [24] and SDS [2]. This table shows score of the average precision (AP), which is computed by measuring the area under a precision recall curve. As presented in the table, our proposed method performs the best in 15 out of 20 object categories and reaches a total average precision of 54.7%. These strong support the efficacy of our proposed model.

4. CONCLUSION

We propose a novel convolutional kernel and build up a context aware module on the task of semantic segmentation. This structure enables us to sample a input image region in a much larger field without increasing the number of filter parameters. Experiments on the semantic segmentation dataset demonstrate the efficacy of our proposed model.

5. REFERENCES

[1] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[2] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik, “Simultaneous detection and segmentation,” in *Proc. European Conf. Computer Vision (ECCV)*. 2014, pp. 297–312, Springer.

[3] Joao Carreira and Cristian Sminchisescu, “Cpmc: Automatic object segmentation using constrained parametric min-cuts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1312–1328, 2012.

[4] Pablo Arbeláez, Jordi Pont-Tuset, Jonathan T Barron, Ferran Marques, and Jitendra Malik, “Multiscale combinatorial grouping,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2014.

[5] Jasper RR Uijlings, Koen EA van de Sande, Theo Gevers, and Arnold WM Smeulders, “Selective search for object recognition,” *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.

[6] Zhuowen Tu and Xiang Bai, “Auto-context and its application to high-level vision tasks and 3d brain image segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 10, pp. 1744–1757, 2010.

[7] Jamie Shotton, Matthew Johnson, and Roberto Cipolla, “Semantic texton forests for image categorization and segmentation,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2008, pp. 1–8.

[8] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, “Going deeper with convolutions,” *arXiv preprint arXiv:1409.4842*, 2014.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” in *Proc. European Conf. Computer Vision (ECCV)*. 2014, pp. 346–361, Springer.

[11] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun, “Overfeat: Integrated recognition, localization and detection using convolutional networks,” *arXiv preprint arXiv:1312.6229*, 2013.

[12] Georgia Gkioxari, Bharath Hariharan, Ross Girshick, and Jitendra Malik, “R-cnns for pose estimation and action detection,” *arXiv preprint arXiv:1406.5212*, 2014.

[13] Alexander Toshev and Christian Szegedy, “Deeppose: Human pose estimation via deep neural networks,” *arXiv preprint arXiv:1312.4659*, 2013.

[14] Clément Farabet, Camille Couprie, Laurent Najman, and Yann LeCun, “Learning hierarchical features for scene labeling,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1915–1929, 2013.

[15] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[16] Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman, “Return of the devil in the details: Delving deep into convolutional nets,” *arXiv preprint arXiv:1405.3531*, 2014.

[17] Min Lin, Qiang Chen, and Shuicheng Yan, “Network in network,” *arXiv preprint arXiv:1406.5726*, 2013.

[18] Pulkit Agrawal, Ross Girshick, and Jitendra Malik, “Analyzing the performance of multilayer neural networks for object recognition,” in *Proc. European Conf. Computer Vision (ECCV)*. 2014, pp. 329–344, Springer.

[19] Geoffrey E Hinton and Ruslan R Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[20] Zhenyao Zhu, Ping Luo, Xiaogang Wang, and Xiaoou Tang, “Deep learning identity-preserving face space,” in *The IEEE International Conf. on Computer Vision (ICCV)*. IEEE, 2013, pp. 113–120.

[21] Eran Borenstein, Eitan Sharon, and Shimon Ullman, “Combining top-down and bottom-up segmentation,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshop (CVPRW)*. IEEE, 2004, pp. 46–46.

[22] Andreas Opelt, Axel Pinz, Michael Fussenegger, and Peter Auer, “Generic object recognition with boosting,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 3, pp. 416–431, 2006.

[23] Mark Everingham, Luc van Gool, Chris Williams, John Winn, and Andrew Zisserman, “The pascal visual object classes challenge 2012 (voc2012) results,” http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[24] João Carreira, Rui Caseiro, Jorge Batista, and Cristian Sminchisescu, “Semantic segmentation with second-order pooling,” in *Proc. European Conf. Computer Vision (ECCV)*. 2012, pp. 430–443, Springer.

[25] David Aldavert, Arnau Ramisa, Ramon Lopez de Mantaras, and Ricardo Toledo, “Fast and robust object segmentation with the integral linear classifier,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 1046–1053.

[26] Daniel Kuettel and Vittorio Ferrari, “Figure-ground segmentation by transferring window masks,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 558–565.

[27] Brian Fulkerson, Andrea Vedaldi, and Stefano Soatto, “Class segmentation and object localization with superpixel neighborhoods,” in *The IEEE International Conf. on Computer Vision (ICCV)*. IEEE, 2009, pp. 670–677.

[28] Victor Lempitsky, Andrea Vedaldi, and Andrew Zisserman, “Pylon model for semantic segmentation,” in *Advances in neural information processing systems*, 2011, pp. 1485–1493.