DEVILS IN SEMANTIC SEGMENTATION

Tao Hu[1]*, Yao Wang*[1]*, Yisong Chen*[1]*, Peng Lu*[2]*, Guoping Wang*[1]

[1]School of Electronics Engineering and Computer Science, Peking University
[2]School of Computer Science, Beijing University of Posts and Telecommunications
{taohu,yaowang95,chenyisong,wgp}@pku.edu.cn, lupeng@bupt.edu.cn

ABSTRACT

Semantic segmentation is the basic task in computer vision. Recently many deep learning based methods greatly boosted the result of this task. However, there are some details that significantly matter the result of semantic segmentation but are ignored by most approaches. Our paper goes through the details about the image size choice setting, reducible convolution, and dealing with batch normalization in semantic segmentation. Our result reaches state of the art in Pascal VOC 2012 without any bells and whistles.

Index Terms— Semantic segmentation

1. INTRODUCTION

Image segmentation is an important task of current computer vision, which has been applied to many interesting fields: autonomous driving, medical imaging, to name a few. Convolutional Neural Network(CNN) has pushed the performance of computer vision systems from coarse-grained to fined-grained tasks: classification[1], detection[2] or localization, semantic and instance segmentation[3]. Most of them are designed as an end-to-end manner that delivers strikingly better result than those traditional hand-crafted feature-based methods.

Several years ago, the CNN is only used in the classification task which maps the input R^n to output R. Nowadays, the most important fundamental deep learning techniques for image segmentation derives from the Fully Convolutional Network(FCN)[4], which mainly uses deconvolution to upsample the feature map to output a same size probability map. Recently, Liang-Chieh Chen et al. put forward a solid work named deeplab[5], it comes up with a new convolution method named Atrous Convolution that greatly enlarges the receptive field of the CNN framework. Conditional Random Field(CRF) is often deployed to smoothen the noisy segmentation map, which serves as a post-process procedure for image segmentation.

Batch Normalization[6] performs an important role in current deep learning field, which can solve the internal covariate shift, realize regularization and speed up the total training process. However, the mean, variance of each layer of the neural network are to a large extent determined by the batch size, which is subject to GPU memory. In this paper, we will discuss how to coordinate the batch size with image crop size to reach a better result.

On the other side, crop size is often neglected by many methods. Inappropriate crop size often leads to bad results. Even if a moderate crop size is chosen, there is also the "irreducible convolution problem"(we will introduce it in later section 2.3) in the fully convolution network. A practical solution on cropping is theoretically and quantitatively demonstrated.

Last but not least, there are many data augmentation methods in semantic segmentation such as horizontal flipping, image rotation, image resizing. In this work, it is shown that the nearest neighbor resizing is better than bilinear resizing via quantitatively analysis. Some underlying explanations about it are also given.

Improvements of our basic network architecture are introduced in Section 2. Details of our ablation study and our state of the art result in Pascal VOC 2012 dataset is demonstrated in Section 3 followed by conclusion and future work in Section 4.

2. METHOD

2.1. Network structure

Our network architecture is based on deeplabv1[5], as indicated in Fig 1. Our basic model is resnet101[7] with conv3.x all equipped with dilation = 2 convolution to enlarge receptive field and keep the feature map size in conv4.x the same with the conv3.x. After the residual part, we upsample our feature map by 16 times and generate the final prediction score map. Our loss is typical multi cross entropy loss.

2.2. Image Resizing Augmentation

The general resize strategy is bilinear resizing for image and Nearest Neighbor(NN)[8] resizing for ground truth label. In this paper, we find that nearest neighbor resizing for image has a better effect than the bilinear resizing.

Nearest Neighbor(NN) is the simplest and fastest implementation of image resize technique. Unlike NN, other

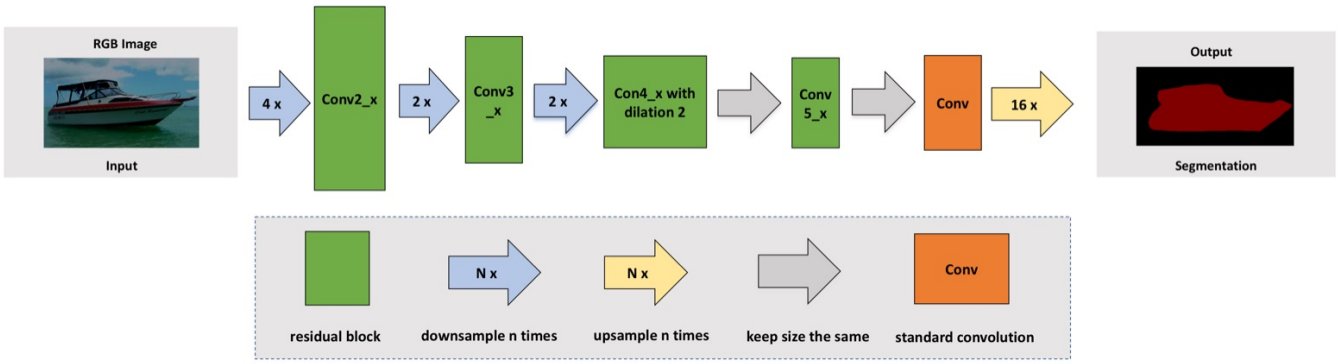


Fig. 1. Network Architecture

complex variation of scaling algorithms like bilinear, bicubic, spline and sinc uses interpolation of neighboring pixels, resulting in smoother image.

The principle of image scaling is to have a reference image and using this image as the base to construct a new scaled image. The constructed image will be smaller, larger, or equal in size depending on the scaling ratio. When enlarging an image, we are actually introducing empty spaces in the original base picture. From the Fig 2, an image with dimension (w1 = 4, h1 = 4) is to be enlarged to (w2 = 8, h2 = 8). The black pixels represent empty spaces where interpolation is needed. The NN and bilinear interpolation results are shown afterwards.

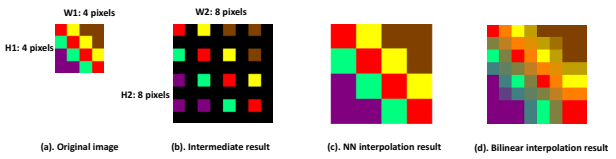


Fig. 2. Nearest Neighbor and Bilinear Interpolation

The image gradient, which can be detected by traditional edge detectors like canny[9], has a high degree of coincidence with the semantic contour [10][11]. The bilinear interpolation smooths the image gradient, which inevitably blurs semantic contours and reduces the accuracy of pixels on the edges. While NN keeps the semantic contour salient and steady.

2.3. Reducible Convolution

The task of semantic segmentation can be interpreted as two parts: (1). pixel classification, which outputs a pixel-level classification probability map. (2). pixel localization. The pixels of the same semantic should be aligned suitably in the final prediction score map. we find that *Reducible Convolution* is beneficial to the semantic localization.

In Convolution Neural Network, we often proceed downsampling based on stride=2 pooling or stride=2 convolution for the following reasons: (1). To enlarge the receptive field of the deeper neural units. (2). To reduce the GPU memory occupation so that we could have a larger batch size for a better optimization result and a better statistics of Batch Normalization(BN).

Our network have an output_stride=16 which means that it proceeds downsampling for 4 times via convolution. As the convention of Convolution operation, the kernel size is often odd which is more convenient for alignment. Except this, we find that when output_stride=16, one uses inputs with spatial dimensions that fits $16k+1, k \in \{0,1,2,..\}$. We use the term *Reducible Convolution* as a shorthand for this criterion. In this case the feature maps at the output will have spatial shape $[\frac{height-1}{output\_stride}+1, \frac{width-1}{output\_stride}+1]$ with corners exactly aligned to the input image corners, which greatly facilitates alignment of the features to image. In our later ablation study we find that this can bring considerable improvement for semantic localization.

2.4. Batch Normalization

Batch Normalization[6] is an important method in computer vision. The detail framework of BN is in Algorithm 1. β, γ are learned via back propagation while the $\mu_{\mathcal{B}}, \sigma^2_{\mathcal{B}}$ are calculated from the batch of feature map. The data source dissimilarity and GPU memory limitation significantly hinders the parameter learning and population statistics stability in BN.

In the task of semantic segmentation, we often use ImageNet[1] pretrained model for fine tuning, which is time- and resource-consuming. When the segmentation source owns the same distribution as the ImageNet data source, we should cherish the weights learned from ImageNet which is a million-level dataset.

3. EXPERIMENT

We carry out experiments on the PASCAL VOC 2012 segmentation dataset [12], which contains 20 object categories and one background class. Following the procedure of [13],

Algorithm 1 Framework of Batch Normalization

Input:
Values of x over a mini-batch: $\mathcal{B} = x_{1...,m}$;
Parameter to be learned: β, γ
Output: $\{y_i = BN_{\gamma,\beta(x_i)}\}$
$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}$
$\sigma^2_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2$
$\hat{x_i} \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}}$
$y_i \leftarrow \gamma\hat{x_i} + \beta = BN_{\beta,\gamma}(x_i)$

we use augmented data with the annotation of [11] resulting 10,582, 1,449 and 1,456 images for training, validation and testing.

To validate the batch normalization effect on different dataset, we also proceed some experiments on the CamVid road scenes dataset[14]. This dataset is small, consisting of 367 training and 233 testing RGB images (day and dusk scenes) at 360×480 resolution. The challenge is to segment 11 classes such as road, building, cars, pedestrians, signs, poles, side-walk etc.

3.1. Training

The ablation study experiments are conducted on single GPU Titan X Pascal. The code is written in Mxnet framework[15]. We use the SGD[16] optimizer with weight decaying parameter : $5e-5$. Learning rate is initialized to $1e-3$ and halved when the objective is stuck in some plateau. For Pascal VOC 2012 and Camvid, we respectively iterate 50K and 10K times. The single GPU training can cost 12 hours.

3.2. Ablation Study

In the following part, we will show some ablation study results on the crop size, reducible convolution, batch normalization. As convention, we use mean IoU, mean accuracymean acc., pixel accuracypixel acc. as the main metric for ablation study.

3.2.1. Coordinate crop size with batch size

As indicated in Table 1, when crop size is 321×321, the batch size is larger, the mIoU is higher. Another interesting fact is that when we set the crop size as 473×473, the max batch size can only be 11 as the limit of GPU memory. the result is better than the result of crop size=321.

Since that the larger crop size can bring larger receptive field for semantic segmentation. Meanwhile, the batch size will drop down due to GPU memory, which will influence the SGD optimization. Therefore, we don't try crop sizes larger than 473.

Table 1. crop size difference in the validation result of Pascal VOC12 dataset

crop size/batch size	mean IoU	mean acc.	pixel acc.
$321 \times 321/10$	69.44	79.58	93.10
$321 \times 321/23$	70.02	81.56	93.16
$473 \times 473/11$	71.10	80.86	93.50

3.2.2. Reducible Convolution

From the result in Table 2, when the architecture is reducible convolution, the result is often higher than the non-reducible convolution in both cases of crop size 321 and 473. We carry the experiment both on Camvid and Pascal VOC12 dataset to eliminate the dataset influence.

The cause of gain on reducible convolution is that the reducible convolution greatly facilitates the alignment of the feature map in the neural network and finally improves the semantic localization ability.

Table 2. reducible convolution difference in the validation result of Pascal VOC12 and Camvid dataset

Dataset	crop size/batch size	mean IoU	mean acc.	pixel acc.
Pascal VOC12	$320 \times 320/23$	68.49	79.72	92.79
	$321 \times 321/23$	70.02	81.56	93.16
	$472 \times 472/11$	70.18	79.55	93.32
	$473 \times 473/11$	71.10	80.86	93.50
Camvid	$320 \times 320/23$	62.04	70.56	91.50
	$321 \times 321/23$	62.73	71.22	91.62
	$472 \times 472/11$	64.77	73.31	92.11
	$473 \times 473/11$	64.51	73.23	92.30

3.2.3. Nearest Neighbor Resizing

In the experiment, we proceed some basic ablation studies between Nearest Neighbor resizing(NN) and bilinear resizing(bilinear) as Table 4. Our experiment is based on Pascal VOC12 and Camvid. We both resize the image by a ratio of 0.7 to 1.5, after the resizing, a 473×473 random crop is applied to generate the input.

From the result, we can see nearest neighbor can bring 1% mIoU gain in both Pascal VOC12 and Camvid. For this phenomenon, we argue that we should not imagine the neural network's perception ability as human being, even though the bilinear resized image is more acceptable for human being, the neural network may process the nearest neighbor resized image better since its jaggy property is more favorable to the classification of the neural network.

3.2.4. batch normalization strategy

We explore different strategies on Pascal VOC12 and Camvid dataset. As we mentioned before, our network is fine-tuned

Table 3. Test result on Pascal VOC12 dataset (Asterisk ($*$) denotes the algorithms that also use Microsoft COCO for training.)

Method	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbk	person	plant	sheep	sofa	train	tv	mean
Hypercolumn[17]	88.9	68.4	27.2	68.2	47.6	61.7	76.9	72.1	71.1	24.3	59.3	44.8	62.7	59.4	73.5	70.6	52.0	63.0	38.1	60.0	54.1	59.2
FCN8s[4]	91.2	76.8	34.2	68.9	49.4	60.3	75.3	74.7	77.6	21.4	62.5	46.8	71.8	63.9	76.5	73.9	45.2	72.4	37.4	70.9	55.1	62.2
Deconvnet	92.7	85.9	42.6	78.9	62.5	66.6	87.4	77.8	79.5	26.3	73.4	60.2	70.8	76.5	79.6	77.7	58.2	77.4	52.9	75.2	59.8	69.6
Deeplabv1-CRF	93.1	84.4	54.5	81.5	63.6	65.9	85.1	79.1	83.4	30.7	74.1	59.8	79.0	76.1	83.2	80.8	59.7	82.2	50.4	73.1	63.7	71.6
* BoxSup	93.6	86.4	35.5	79.7	65.2	65.2	84.3	78.5	83.7	30.5	76.2	62.6	79.3	76.1	82.1	81.3	57.0	78.2	55.0	72.5	68.1	71.0
Our method	94.05	86.40	40.99	84.31	60.94	70.73	89.60	84.92	86.90	35.50	80.08	58.00	79.30	79.94	85.52	84.23	62.17	83.60	53.23	81.92	68.86	73.87

Table 4. nearest neighbor resizing in the validation result of Pascal VOC12 dataset

dataset	method	mean IoU	mean acc.	pixel acc.
VOC12	bilinear	70.05	79.99	93.36
	NN	71.21	80.50	93.57
Camvid	bilinear	64.16	72.86	92.24
	NN	65.27	74.24	92.27

on ImageNet dataset classification task. And the data distribution of ImageNet is more close to Pascal VOC 12 than Camvid Road Scene dataset.

We summaries three strategies about BN. (1). "free", $\beta, \gamma, \mu_{\mathcal{B}}, \sigma^2_{\mathcal{B}}$ are all calculated in immediately. (2). "fix β, γ". which means we use the learned β, γ in ImageNet dataset. and the $\mu_{\mathcal{B}}, \sigma^2_{\mathcal{B}}$ are calculated in the segmentation dataset. (3). "fix all". all the $\beta, \gamma, \mu_{\mathcal{B}}, \sigma^2_{\mathcal{B}}$ are all kept the same as the ImageNet dataset learned weight.

As the result in Table 5. we can find "fix all" is the best strategy for Pascal VOC12 dataset, and "free" is the best strategy for Camvid dataset. It conforms with the fact the Pascal VOC12 is more similar with ImageNet so that it nearly doesn't need BN parameters tuning. While Camvid strongly need the BN parameter tuning and can obtain 5% mIoU gain compared with "fix all" strategy.

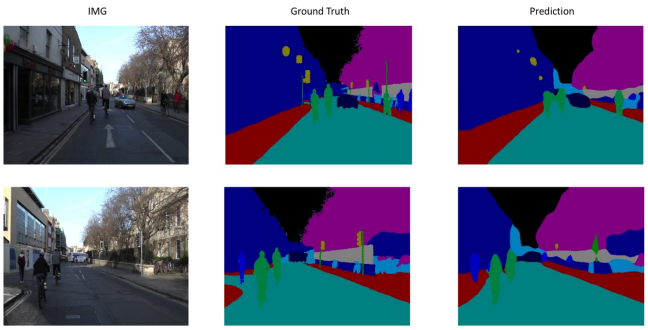
Table 5. BN strategy difference in validation result of Pascal VOC12 and Camvid dataset

Dataset	method	mean IoU	mean acc.	pixel acc.
Pascal VOC12	free	70.05	79.99	93.36
	fix β, γ	71.10	80.86	93.50
	fix all	71.30	80.62	93.57
Camvid	free	64.16	72.86	92.24
	fix β, γ	63.99	72.66	92.11
	fix all	59.2	68.98	91.09

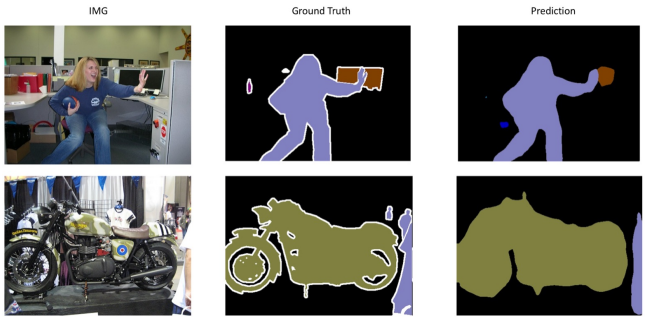
3.3. Result

In our ablation study, a new model is introduced and the state of art performance in test result of Pascal VOC12 dataset is indicated in Table 3. Based on our improvements, we also apply multi gpu training for our final model, which greatly stabilizes the optimization and BN statistics via moving average. Compared with model deeplabv1-CRF, our result can obtain 2.27% mIoU gain. Please notice that deeplab-v1 baseline has a CRF post-processing while our result doesn't. But still we can reach a competitive result.

Some of our test evaluation results are shown in Fig 3.



(a) Camvid Evaluation Result



(b) Pascal VOC12 Evaluation Result

Fig. 3. Evaluation Result

4. CONCLUSION

In this paper, we try some improvements in semantic segmentation including crop size choice, reducible convolution, batch normalization freeze choice and nearest neighbor resizing augmentation. Based on our improvements on those devil details and under the same architecture our work favorably compares with the state of the art.

In our future work, we will try to address the large GPU occupation of batch normalization and the stability and generalization of the batch normalization.

5. REFERENCES

[1] 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.

[2] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Neural Information Processing Systems (NIPS)*, 2015.

[3] Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei, "Fully convolutional instance-aware semantic segmentation," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2359–2367.

[4] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *arXiv preprint arXiv:1606.00915*, 2016.

[6] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*, 2015, pp. 448–456.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[8] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft, "When is nearest neighbor meaningful?," in *International conference on database theory*. Springer, 1999, pp. 217–235.

[9] John Canny, "A computational approach to edge detection," in *Readings in Computer Vision*, pp. 184–203. Elsevier, 1987.

[10] Mukta Prasad, Andrew Zisserman, Andrew Fitzgibbon, M Pawan Kumar, and Philip HS Torr, "Learning class-specific edges for object detection and segmentation," in *Computer Vision, Graphics and Image Processing*, pp. 94–105. Springer, 2006.

[11] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik, "Semantic contours from inverse detectors," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 991–998.

[12] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results," http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[13] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia, "Pyramid scene parsing network," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2881–2890.

[14] Gabriel J Brostow, Julien Fauqueur, and Roberto Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[15] Tianqi Chen, Mu Li, Yutian Li, Min Lin, Naiyan Wang, Minjie Wang, Tianjun Xiao, Bing Xu, Chiyuan Zhang, and Zheng Zhang, "Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems," *arXiv preprint arXiv:1512.01274*, 2015.

[16] Léon Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*, pp. 177–186. Springer, 2010.

[17] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 447–456.