Pyramid Scene Parsing Network
(CVPR 2017)

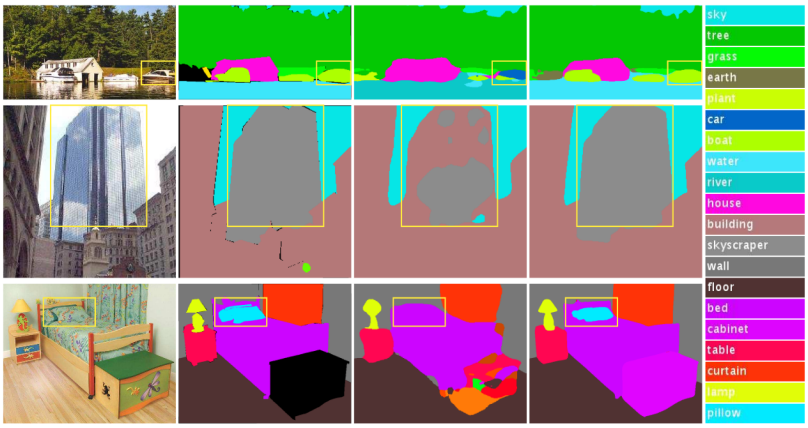
Zhao et al.
Resume

February 22, 2019

Key Contributions:

- Propose pyramid pooling module to aggregate the context.
- Use auxiliary loss

1 Observations



(a) Image (b) Ground Truth (c) FCN (d) PSPNet (e) ColorMap

There is several issues with FCN scene parsing, summarized as follows:

- **Mismatched Relationship:** Context relationship is universal and important especially for complex scene understanding, in the figure above, first row, FCN predict a car based on the appearance, but we know the a car cannot be over a river.
- **Confusion Categories:** as seen in the second row, the FCN predicts the object in the box as part of skyscraper and part of building, These re-sults should be excluded so that the whole object is either skyscraper or building, but not both.
- **Inconspicuous Classes:** Several small-size things, like streetlight and signboard, are hard to find while they may be of great importance. in the third row, the pillow has similar appearance with the sheet. Overlooking the global scene category may fail to parse the pillow. To improve performance for remarkably small or large objects, one should pay much attention to different sub-regions that contain inconspicuous-categories.

Many errors are partially or completely related to contextual relationship and global information for different receptive fields.

2 Pyramid pooling module

Global scene categories matter because it provides clues on the distribution of the segmentation classes. Pyramid pooling module captures this information by applying large kernel pooling layers.

Dilated convolutions are used as in dilated convolutions paper to modify Resnet and a pyramid pooling module is added to it. This module concatenates the feature maps from ResNet with upsampled output of parallel pooling layers with kernels covering whole, half of and small portions of image.

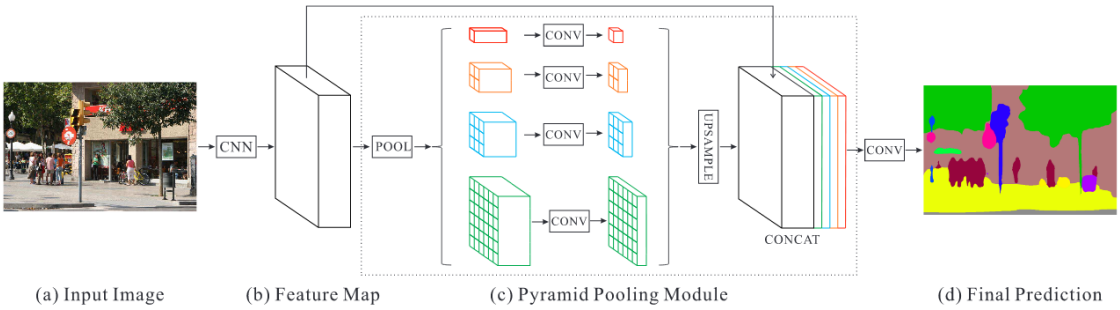


Figure 3. Overview of our proposed PSPNet. Given an input image (a), we first use CNN to get the feature map of the last convolutional layer (b), then a pyramid parsing module is applied to harvest different sub-region representations, followed by upsampling and concatenation layers to form the final feature representation, which carries both local and global context information in (c). Finally, the representation is fed into a convolution layer to get the final per-pixel prediction (d).

"The pyramid pooling module fuses features under fourdifferent pyramid scales. The coarsest level highlighted in red is global pooling to generate a single bin output. The following pyramid level separates the feature map into different sub-regions and forms pooled representation for different locations. The output of different levels in the pyramid pooling module contains the feature map with varied sizes. To maintain the weight of global feature, we use 11 convolution layer after each pyramid level to reduce the dimension of context representation to 1/N of the original one if the level size of pyramid is N. Then we directly up-sample the low-dimension feature maps to get the same sizefeature as the original feature map via bilinear interpolation. Finally, different levels of features are concatenated as the final pyramid pooling global feature."

3 Auxiliary loss

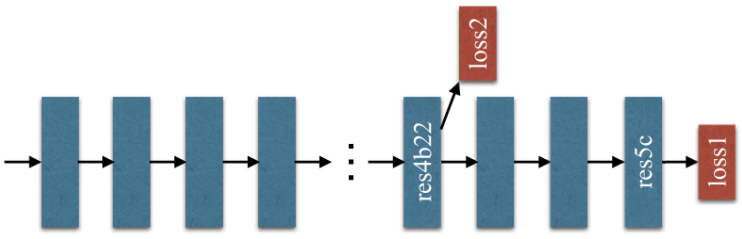


Figure 4. Illustration of auxiliary loss in ResNet101. Each blue box denotes a residue block. The auxiliary loss is added after the res4b22 residue block.

An auxiliary loss, additional to the loss on main branch, is applied after the fourth stage of ResNet (i.e input to pyramid pooling module). This idea was also called as intermediate supervision elsewhere.

4 Architecture details

The network is independt on the back-bone used, as long as we use the pyramid pooling and the auxiliary loss, less take the implementation of the network using a resnet, in resnet, at the start of the network we down sample our image with a factor of 1/4, using a conv 7x7 with a stride of two, and a max pool, and then we have 4 stages:

- First stage with three blocks, each one with similar structure (conv – bn x3 – Relu), 3 conv : 1x1, 3x3, and 1x1 all with stride 1 & and just for the first block where the #C changes, an identity connection with one conv 1x1 to adjust the channels (64 ¿ 256 followed by 256 – 64 – 256 x 2).
- Second stage with same structure with 4 blocks and the 3x3 conv in the first block is with stride (256 – 512 followed by (512 – 128 – 512) x 3).
- Second stage with same structure with 6 blocks and the 3x3 conv in the first block is with stride 2 (256 – 512 followed by (512 – 128 – 512) x 3)
- Second stage with same structure with 3 blocks and the 3x3 conv in the first block is with stride (1024 – 2048 followed by (2048 – 512 – 2048) x 3)

Now, in the resnet, we downsmaple 1/4 in the begining, and then 1/2 x 3 (strides) in the stages 2-4, now in PSPnet the ouput is 1/8 of the size of the input. because we replace stides the conv 3x3 in the stages 3 and 4 with a dilation of 2 to get the same receptive field without any further reduction of the spatial dimensions.

5 Results

Method	Mean IoU(%)	Pixel Acc.(%)
ResNet50-Baseline	37.23	78.01
ResNet50+B1+MAX	39.94	79.46
ResNet50+B1+AVE	40.07	79.52
ResNet50+B1236+MAX	40.18	79.45
ResNet50+B1236+AVE	41.07	79.97
ResNet50+B1236+MAX+DR	40.87	79.61
ResNet50+B1236+AVE+DR	**41.68**	**80.04**

Table 1. Investigation of PSPNet with different settings. Baseline is ResNet50-based FCN with dilated network. 'B1' and 'B1236' denote pooled feature maps of bin sizes $\{1 \times 1\}$ and $\{1 \times 1, 2 \times 2, 3 \times 3, 6 \times 6\}$ respectively. 'MAX' and 'AVE' represent max pooling and average pooling operations individually. 'DR' means that dimension reduction is taken after pooling. The results are tested on the validation set with the single-scale input.

Loss Weight α	Mean IoU(%)	Pixel Acc.(%)
ResNet50 (without AL)	35.82	77.07
ResNet50 (with $\alpha = 0.3$)	37.01	77.87
ResNet50 (with $\alpha = 0.4$)	**37.23**	**78.01**
ResNet50 (with $\alpha = 0.6$)	37.09	77.84
ResNet50 (with $\alpha = 0.9$)	36.99	77.87

Table 2. Setting an appropriate loss weight α in the auxiliary branch is important. 'AL' denotes the auxiliary loss. Baseline is ResNet50-based FCN with dilated network. Empirically, $\alpha = 0.4$ yields the best performance. The results are tested on the validation set with the single-scale input.

Method	Mean IoU(%)	Pixel Acc.(%)
PSPNet(50)	41.68	80.04
PSPNet(101)	41.96	80.64
PSPNet(152)	42.62	80.80
PSPNet(269)	**43.81**	**80.88**
PSPNet(50)+MS	42.78	80.76
PSPNet(101)+MS	43.29	81.39
PSPNet(152)+MS	43.51	81.38
PSPNet(269)+MS	**44.94**	**81.69**

Table 3. Deeper pre-trained model get higher performance. Number in the brackets refers to the depth of ResNet and 'MS' denotes multi-scale testing.