Deeplab: semantic image segmentation with DCNN, atrous convs and CRFs

(2016)

Chen et al.
Resume

February 28, 2019

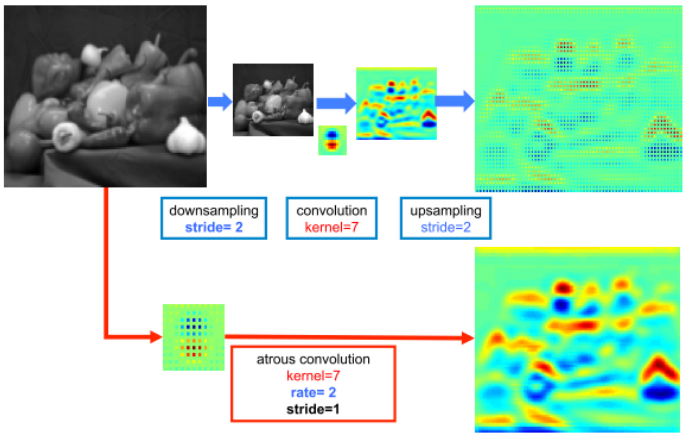
---

1 Introduction

In this work, they present:
- Atrous convolution, a powerful tool for dense predictions.
- Atrous spatial pyramid pooling (ASSP), to segment the image at multiple scales.
- Using fully connected CRFs to combine the improve DCNN acacuracy by refining the segmentation masks.

2 Atrous convolutions

In most recent DCNN architecture, we see a heavy use of maxpooling, to add some translation invariance to the network and also reduce the computation overhead by reducing the spatial dimensions and distilling the iformation while adding more depth to the network and enlarging the receptive field. In semantic segmentation, in addition to the global context, the low level information is as important to obtain sharp segmentation masks.

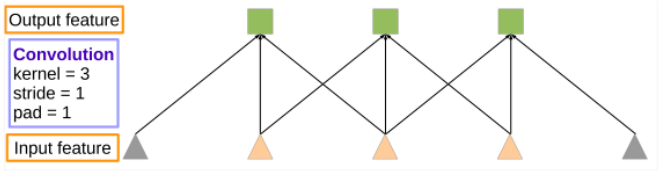


One solution is to use atrous convolutions, such that with the same amount of computation (we don't multiply by the zeros in the filters) we can obtain similar receptive field without any reduction in the spatial dimensions, one downside is the amount of memory require to store the feature maps increases, for that the authors only add atrous convolutions in the last layers, replacing the max

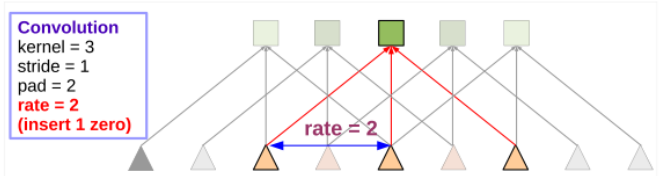
pooling layers in the last two to three layers, in resnet for example, instead of reducing the spatial dimensions by a factor of 32, they replacing the strided convolution in the last two blocks with dilated convolutions, and thus the resulting feature maps are 1/8 of the original size, and with simple bilinar upsampling the results are much better that using the upsamling directely on the 1/32 volume, and without any need for learned filters to upsample the output.

Atrous convolution formulation: Considering one-dimensional signals, the output $y[i]$ of atrous convolution 2 of a 1-D input signal $x[i]$ with a filter $w[k]$ of length K is defined as:

$$y[i] = \sum_{k=1}^{K} x[i + r \cdot k] w[k]$$



(a) Sparse feature extraction



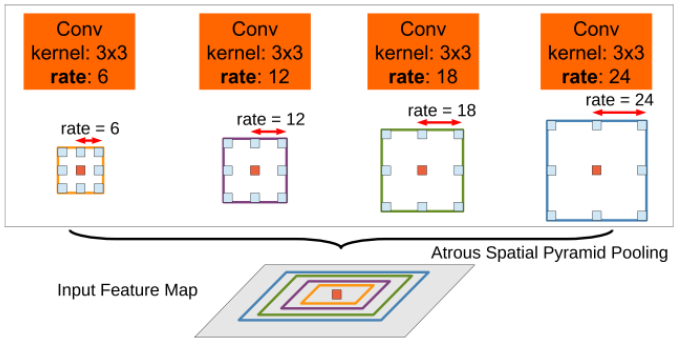
So Atrous convolution with rate r introduces $r - 1$ zeros between consecutive filter values, effectively enlarging the kernel size of a $k \times k$ filter to $k_e = k + (k - 1)(r - 1)$ without increasing the number of parameters or the amount of computation.

3 Atrous spatial pyramid pooling (ASSP)

DCNN are capable learing multiple scales, by simply being trained on detecting the same objects in different scales that are present in the dataset, but to better help the network, the authors propose two techniques to handle the multiscale segmentation:

1- pass three resclaed version of the image through the DCNN (x1, x0.75 and x0.5), upsample the ouputs of x0.75 and x0.5 to x1 and take the max over all three volumes.

2- use multiple parallel branches with atrous filters with different rates, so the features extracted for each sampling rate are further processed in separate branches and fused to generate the final result.



4 Fully connected CRFs

To overcome the limitations of short-range CRFs, the authors integrate a fully connected CRF model into the network, taking as input the predicted maks by the DCNN. The employed the energy function is:

$$E(\boldsymbol{x}) = \sum_i \theta_i(x_i) + \sum_{ij} \theta_{ij}(x_i, x_j)$$

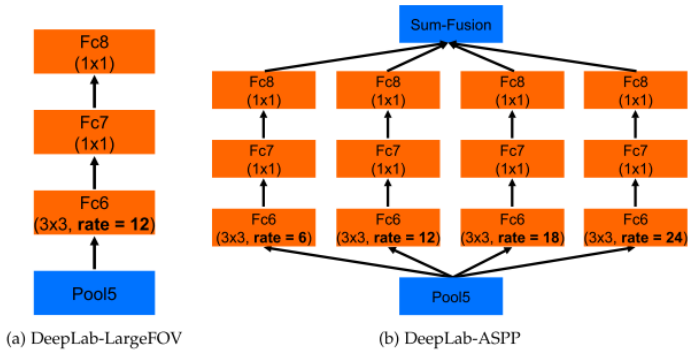
- x in the label assignemnet for pixels. - $\theta_i(x_i) = -\log P(x_i)$ is the negative cross entropy of the predicted labels by the DCNN. - No, the fully connected part comes from scoring all paris of pixels in the image: $\theta_{ij}(x_i, x_j)$:

$$\theta_{ij}(x_i, x_j) = \mu(x_i, x_j)\left[w_1 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\alpha^2} - \frac{\|I_i - I_j\|^2}{2\sigma_\beta^2}\right) + w_2 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\gamma^2}\right)\right]$$

the term μ is used to not take into consideration (no loss) a pair of the same pixel, with two gaussian kernels in difference spaces (possitions denoted as p and the pixels intensity I), the first term forces the pixels close by to have the same colors, and the second term only considers the proximity to add some smoothness in the transitions.

5 Expetiments

models:



(a) DeepLab-LargeFOV (b) DeepLab-ASPP

- **loss:** CR entropy between the outputs (1/8 of the original image) and the labels (subsampled by a factor of 1/8), and the unlabed pixels (255) are ignored.
- **learning rate:** $\left(1 - \frac{iter}{max_iter}\right)^{power}$), power = 0.9.
- **dilation rates:** for the ASSP: ASSP-L = 2, 4, 8, 12 and ASSP-S = 6, 12, 18, 24.
- **backbones:** VGG16 and resnet 101, pretrained on MS-COCO.

6 results

Method	before CRF	after CRF
LargeFOV	65.76	69.84
ASPP-S	66.98	69.73
ASPP-L	68.96	71.57

MSC	COCO	Aug	LargeFOV	ASPP	CRF	mIOU
						68.72
✓						71.27
✓	✓					73.28
✓	✓	✓				74.87
✓	✓	✓	✓			75.54
✓	✓	✓		✓		76.35
✓	✓	✓		✓	✓	77.69

Method	MSC	COCO	Aug	LargeFOV	ASPP	CRF	mIOU
VGG-16							
DeepLab [38]				✓			37.6
DeepLab [38]				✓		✓	39.6
ResNet-101							
DeepLab							39.6
DeepLab	✓		✓				41.4
DeepLab	✓	✓	✓				42.9
DeepLab	✓	✓	✓	✓			43.5
DeepLab	✓	✓	✓		✓		44.7
DeepLab	✓	✓	✓		✓	✓	45.7
O_2P [45]							18.1
CFM [51]							34.4
FCN-8s [14]							37.8
CRF-RNN [59]							39.3
ParseNet [86]							40.4
BoxSup [60]							40.5
HO_CRF [91]							41.3
Context [40]							43.3
VeryDeep [93]							44.5