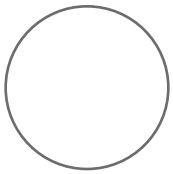


Qure.ai Blog

Revolutionizing healthcare with deep learning

A 2017 Guide to Semantic Segmentation with Deep Learning

Sasank Chilamkurthy | July 5, 2017

At Qure, we regularly work on segmentation and object detection problems and we were therefore interested in reviewing the current state of the art.

In this post, I review the literature on semantic segmentation. Most research on semantic segmentation use natural/real world image datasets. Although the results are not directly applicable to medical images, I review these papers because research on the natural images is much more mature than that of medical images.

Post is oraganized as follows: I first explain the semantic segmentation problem, give an overview of the approaches and summarize a few interesting papers.

In a later post, I'll explain why medical images are different from natural images and examine how the approaches from this review fare on a dataset representative of medical images.

What exactly is semantic segmentation?

Semantic segmentation is understanding an image at pixel level i.e, we want to asign each pixel in the image an object class. For example, check out the following images.

Left: Input image. *Right*: It's semantic segmentation. Source.

Apart from recognising the bike and the person riding it, we also have to delineate the boundaries of each object. Therefore, unlike classification, we need dense pixel-wise predictions from our models.

VOC2012 and MSCOCO are the most important datasets for semantic segmentation.

What are the different approaches?

Before deep learning took over computer vision, people used approaches like TextonForest and Random Forest based classifiers for semantic segmentation. As with image classification, convolutional neural networks (CNN) have had enormous success on segmentation problems.

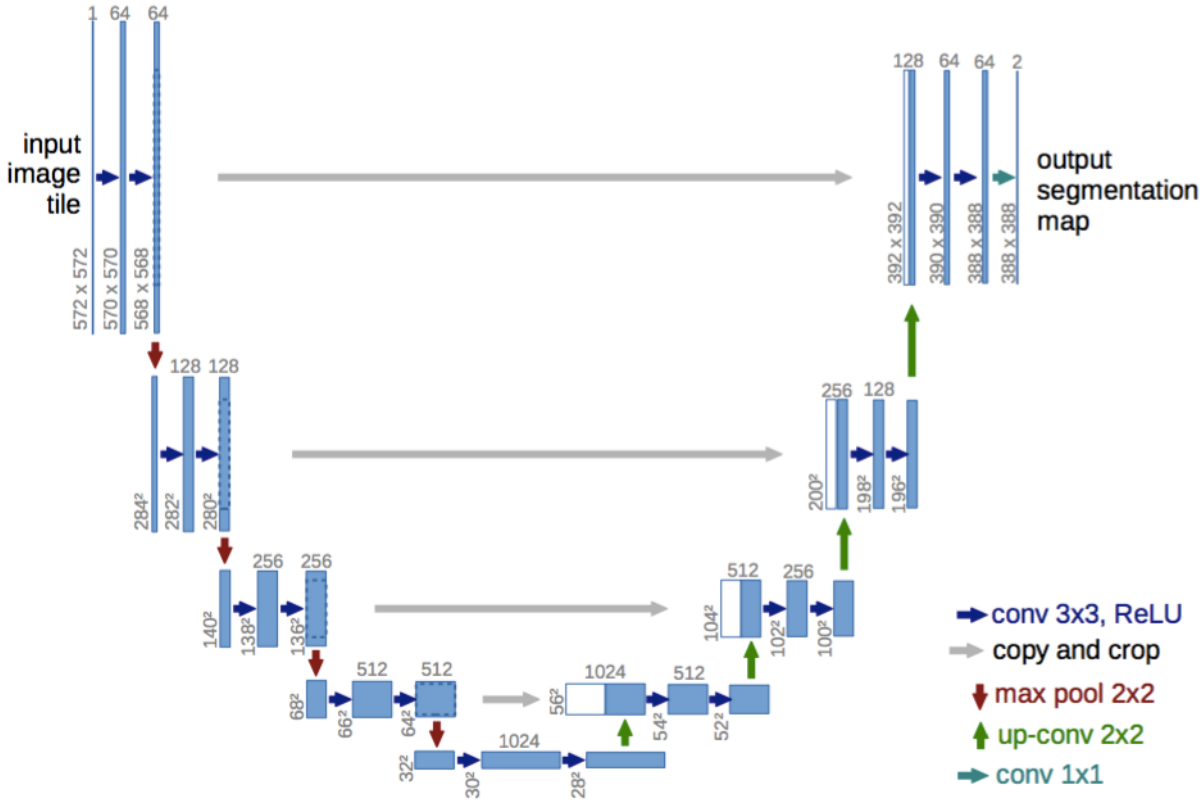
One of the popular initial deep learning approaches was patch classification where each pixel was separately classified into classes using a patch of image around it. Main reason to use patches was that classification networks usually have full connected layers and therefore required fixed size images.

In 2014, Fully Convolutional Networks (FCN) by Long et al. from Berkeley, popularized

CNN architectures for dense predictions without any fully connected layers. This allowed segmentation maps to be generated for image of any size and was also much faster compared to the patch classification approach. Almost all the subsequent state of the art approaches on semantic segmentation adopted this paradigm.

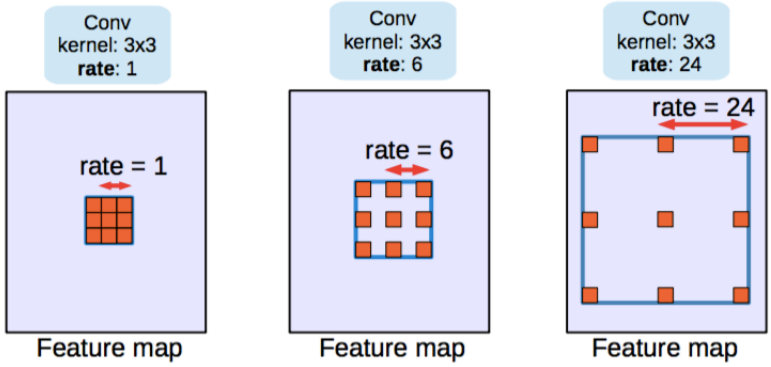
Apart from fully connected layers, one of the main problems with using CNNs for segmentation is *pooling layers*. Pooling layers increase the field of view and are able to aggregate the context while discarding the 'where' information. However, semantic segmentation requires the exact alignment of class maps and thus, needs the 'where' information to be preserved. Two different classes of architectures evolved in the literature to tackle this issue.

First one is encoder-decoder architecture. Encoder gradually reduces the spatial dimension with pooling layers and decoder gradually recovers the object details and spatial dimension. There are usually shortcut connections from encoder to decoder to help decoder recover the object details better. U-Net is a popular architecture from this class.



U-Net: An encoder-decoder architecture. Source.

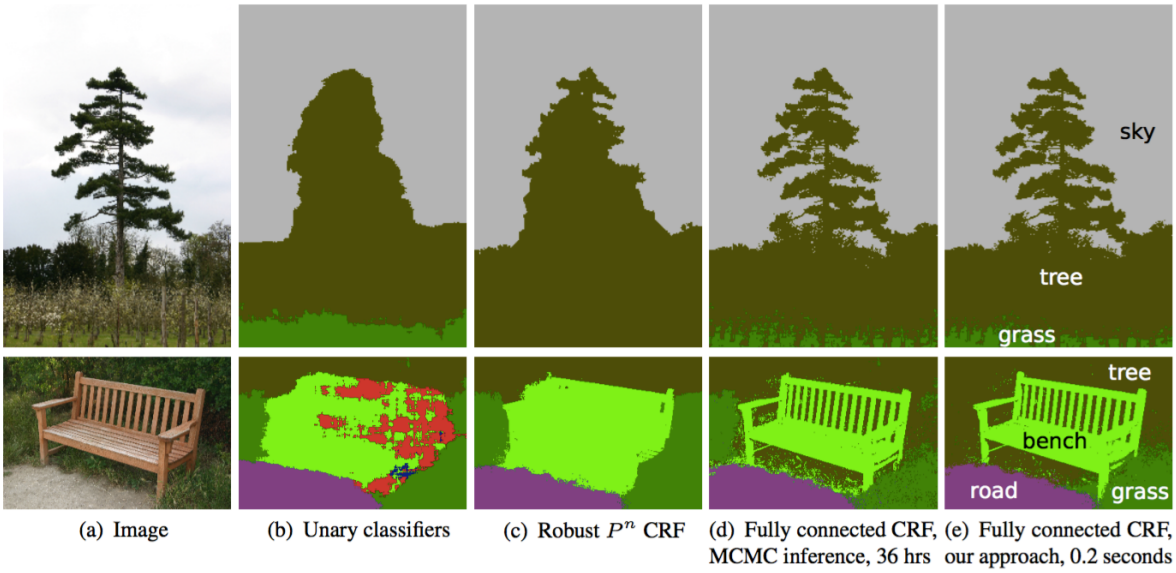
Architectures in the second class use what are called as dilated/atrous convolutions and do away with pooling layers.



Dilated/atrous convolutions. rate=1 is same as normal convolutions. Source.

Conditional Random Field (CRF) postprocessing are usually used to improve the segmentation. CRFs are graphical models which 'smooth' segmentation based on the

underlying image intensities. They work based on the observation that similar intensity pixels tend to be labeled as the same class. CRFs can boost scores by 1-2%.



CRF illustration. (b) Unary classifiers is the segmentation input to the CRF. (c, d, e) are variants of CRF with (e) being the widely used one. Source.

In the next section, I'll summarize a few papers that represent the evolution of segmentation architectures starting from FCN. All these architectures are benchmarked on VOC2012 evaluation server.

Summaries

Following papers are summarized (in chronological order):

1. FCN
2. SegNet
3. Dilated Convolutions
4. DeepLab (v1 & v2)
5. RefineNet
6. PSPNet
7. Large Kernel Matters
8. DeepLab v3

For each of these papers, I list down their key contributions and explain them. I also

show their benchmark scores (mean IOU) on VOC2012 test dataset.

FCN

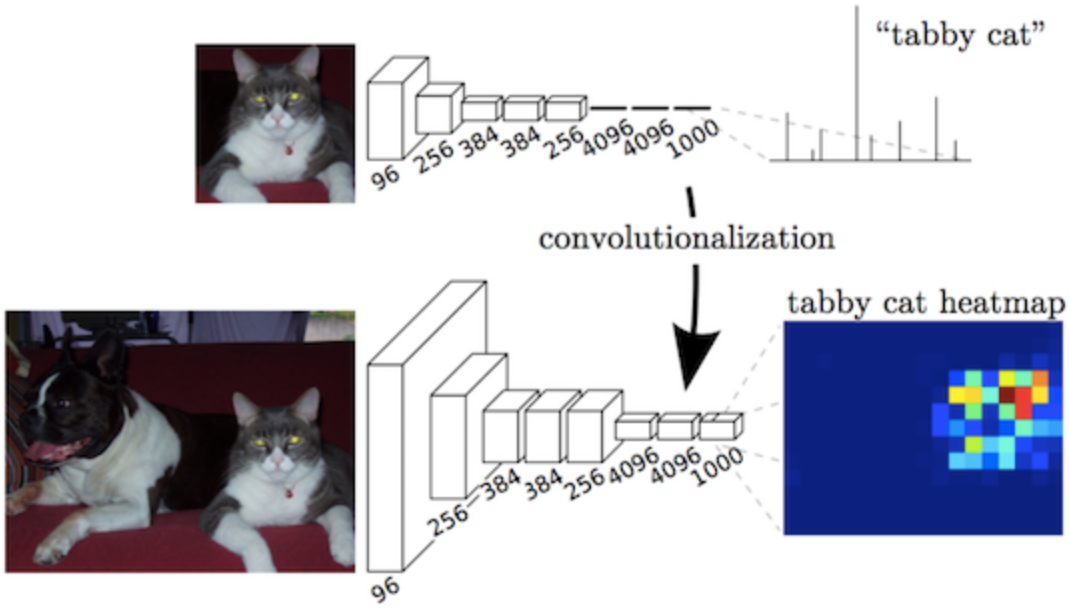
Fully Convolutional Networks for Semantic Segmentation
Submitted on 14 Nov 2014
Arxiv Link

Key Contributions:

- Popularize the use of end to end convolutional networks for semantic segmentation
- Re-purpose imagenet pretrained networks for segmentation
- Upsample using *deconvolutional* layers
- Introduce skip connections to improve over the coarseness of upsampling

Explanation:

Key observation is that fully connected layers in classification networks can be viewed as convolutions with kernels that cover their entire input regions. This is equivalent to evaluating the original classification network on overlapping input patches but is much more efficient because computation is shared over the overlapping regions of patches. Although this observation is not unique to this paper (see overfeat, this post), it improved the state of the art on VOC2012 significantly.



Fully connected layers as a convolution. Source.

After convolutionalizing fully connected layers in a imagenet pretrained network like VGG, feature maps still need to be upsampled because of pooling operations in CNNs. Instead of using simple bilinear interpolation, *deconvolutional layers* can learn the interpolation. This layer is also known as upconvolution, full convolution, transposed convolution or fractionally-strided convolution.

However, upsampling (even with deconvolutional layers) produces coarse segmentation maps because of loss of information during pooling. Therefore, shortcut/skip connections are introduced from higher resolution feature maps.

Benchmarks (VOC2012):

Score	Comment	Source
62.2	-	leaderboard
67.2	More momentum. Not described in paper	leaderboard

My Comments:

- This was an important contribution but state of the art has improved a lot by now though.

SegNet

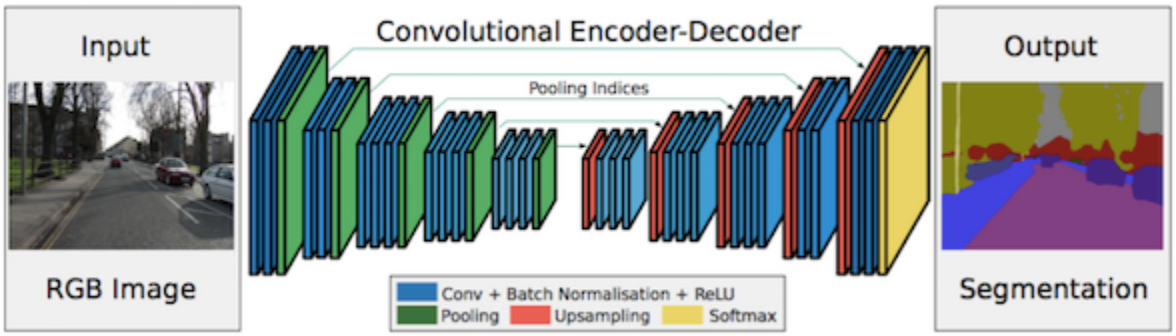
SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation
Submitted on 2 Nov 2015
Arxiv Link

Key Contributions:

- Maxpooling indices transferred to decoder to improve the segmentation resolution.

Explanation:

FCN, despite upconvolutional layers and a few shortcut connections produces coarse segmentation maps. Therefore, more shortcut connections are introduced. However, instead of copying the encoder features as in FCN, indices from maxpooling are copied. This makes SegNet more memory efficient than FCN.



Segnet Architecture. Source.

Benchmarks (VOC2012):

Score	Comment	Source
59.9	-	leaderboard

My comments:

- FCN and SegNet are one of the first encoder-decoder architectures.
- Benchmarks for SegNet are not good enough to be used anymore.

Dilated Convolutions

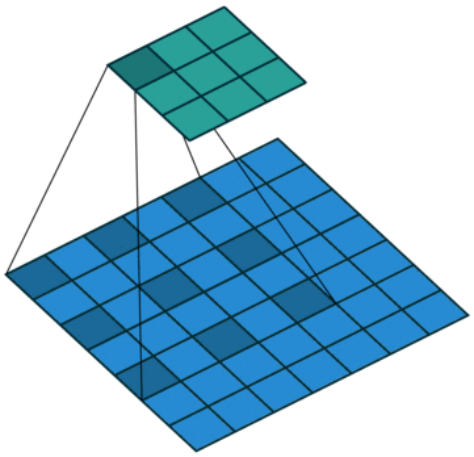
Multi-Scale Context Aggregation by Dilated Convolutions
Submitted on 23 Nov 2015
Arxiv Link

Key Contributions:

- Use dilated convolutions, a convolutional layer for dense predictions.
- Propose 'context module' which uses dilated convolutions for multi scale aggregation.

Explanation:

Pooling helps in classification networks because receptive field increases. But this is not the best thing to do for segmentation because pooling decreases the resolution. Therefore, authors use *dilated convolution* layer which works like this:



Dilated/Atrous Convolutions. Source

Dilated convolutional layer (also called as atrous convolution in DeepLab) allows for

exponential increase in field of view without decrease of spatial dimensions.

Last two pooling layers from pretrained classification network (here, VGG) are removed and subsequent convolutional layers are replaced with dilated convolutions. In particular, convolutions between the pool-3 and pool-4 have dilation 2 and convolutions after pool-4 have dilation 4. With this module (called *frontend module* in the paper), dense predictions are obtained without any increase in number of parameters.

A module (called *context module* in the paper) is trained separately with the outputs of frontend module as inputs. This module is a cascade of dilated convolutions of different dilations so that multi scale context is aggregated and predictions from frontend are improved.

Benchmarks (VOC2012):

Score	Comment	Source
71.3	frontend	reported in the paper
73.5	frontend + context	reported in the paper
74.7	frontend + context + CRF	reported in the paper
75.3	frontend + context + CRF-RNN	reported in the paper

My comments:

- Note that predicted segmentation map's size is 1/8th of that of the image. This is the case with almost all the approaches. They are interpolated to get the final segmentation map.

DeepLab (v1 & v2)

v1 : Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs
Submitted on 22 Dec 2014
Arxiv Link

v2 : DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs
Submitted on 2 Jun 2016
Arxiv Link

Key Contributions:

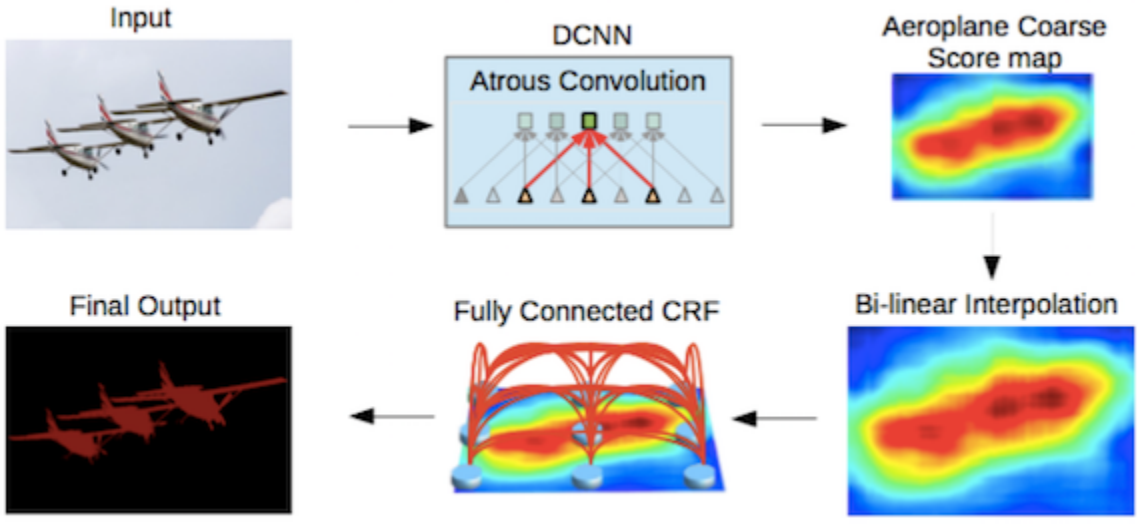
- Use atrous/dilated convolutions.
- Propose atrous spatial pyramid pooling (ASPP)
- Use Fully connected CRF

Explanation:

Atrous/Dilated convolutions increase the field of view without increasing the number of parameters. Net is modified like in dilated convolutions paper.

Multiscale processing is achieved either by passing multiple rescaled versions of original images to parallel CNN branches (Image pyramid) and/or by using multiple parallel atrous convolutional layers with different sampling rates (ASPP).

Structured prediction is done by fully connected CRF. CRF is trained/tuned seperately as a post processing step.



DeepLab2 Pipeline. Source.

Benchmarks (VOC2012):

Score	Comment	Source
79.7	ResNet-101 + atrous Convolutions + ASPP + CRF	leaderboard

RefineNet

RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation
Submitted on 20 Nov 2016
Arxiv Link

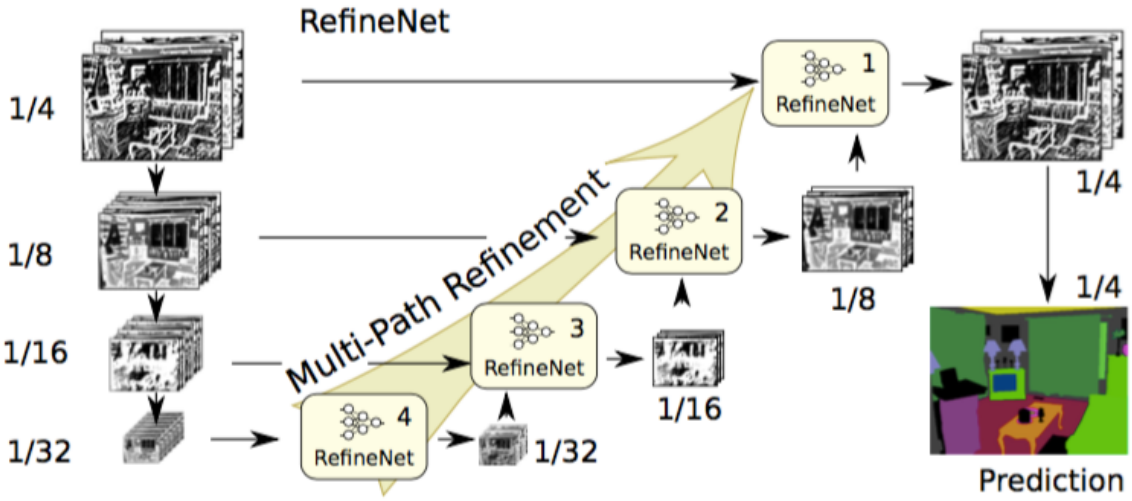
Key Contributions:

- Encoder-Decoder architecture with well thought-out decoder blocks
- All the components follow residual connection design

Explanation:

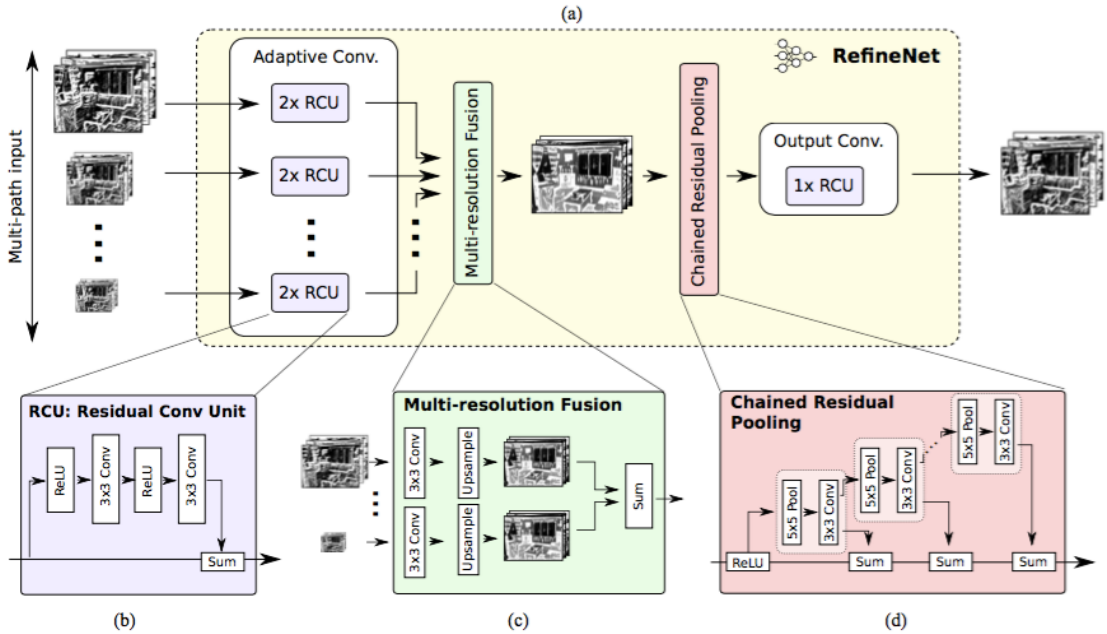
Approach of using dilated/atrous convolutions are not without downsides. Dilated convolutions are computationally expensive and take a lot of memory because they have to be applied on large number of high resolution feature maps. This hampers the computation of high-res predictions. DeepLab's predictions, for example are 1/8th the size of original input.

So, the paper proposes to use encoder-decoder architecture. Encoder part is ResNet-101 blocks. Decoder has RefineNet blocks which concatenate/fuse high resolution features from encoder and low resolution features from previous RefineNet block.



RefineNet Architecture. Source.

Each RefineNet block has a component to fuse the multi resolution features by upsampling the lower resolution features and a component to capture context based on repeated 5 x 5 *stride 1* pool layers. Each of these components employ the residual connection design following the identity map mindset.



RefineNet Block. Source.

Benchmarks (VOC2012):

Score	Comment	Source
84.2	Uses CRF, Multiscale inputs, COCO pretraining	leaderboard

PSPNet

Pyramid Scene Parsing Network
Submitted on 4 Dec 2016
Arxiv Link

Key Contributions:

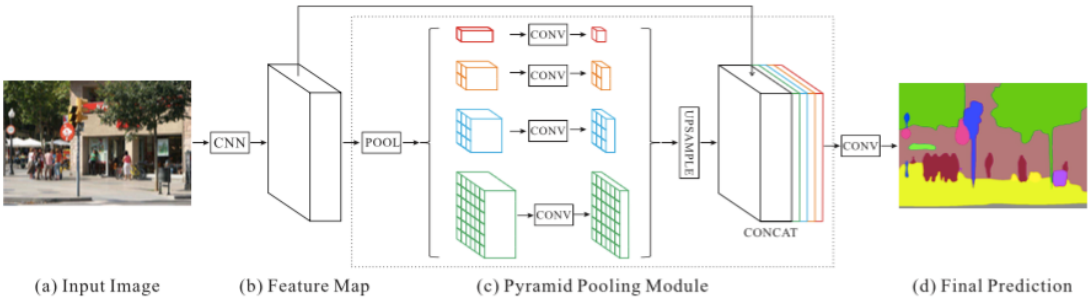
- Propose pyramid pooling module to aggregate the context.
- Use auxiliary loss

Explanation:

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.

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.



PSPNet Architecture. Source.

Benchmarks (VOC2012):

Score	Comment	Source
85.4	MSCOCO pretraining, multi scale input, no CRF	leaderboard
82.6	no MSCOCO pretraining, multi scale input, no CRF	reported in the paper

Large Kernel Matters

> Large Kernel Matters -- Improve Semantic Segmentation by Global Convolutional Network
> Submitted on 8 Mar 2017
> Arxiv Link

Key Contributions:

- Propose a encoder-decoder architecture with very large kernels convolutions

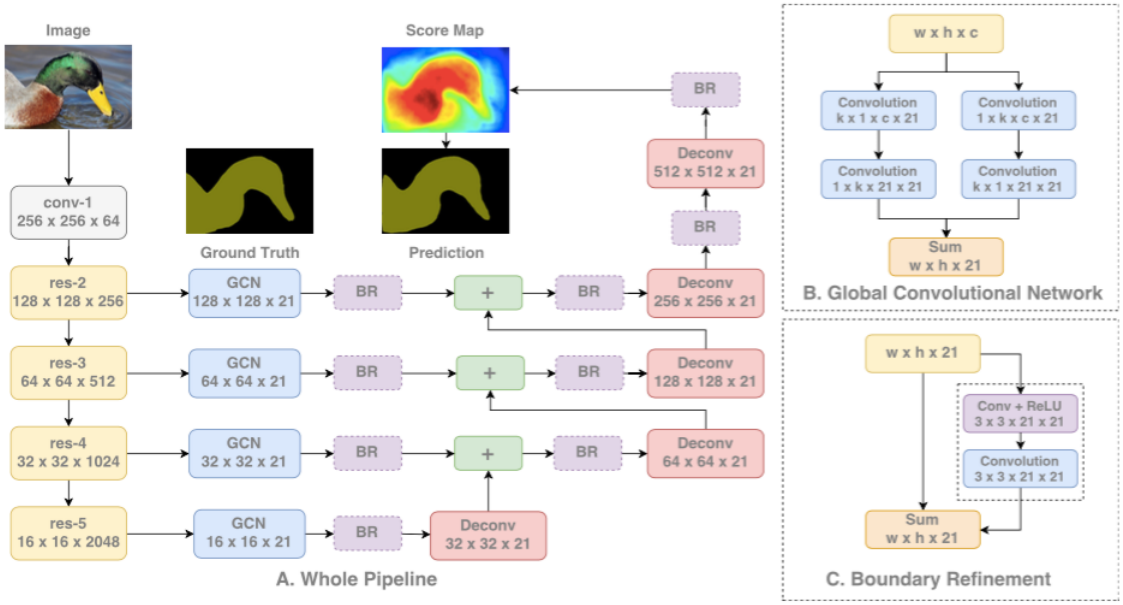
Explanation:

Semantic segmentation requires both segmentation and classification of the segmented objects. Since fully connected layers cannot be present in a segmentation architecture, convolutions with very large kernels are adopted instead.

Another reason to adopt large kernels is that although deeper networks like ResNet have very large receptive field, studies show that the network tends to gather information from a much smaller region (valid receptive filed).

Larger kernels are computationally expensive and have a lot of parameters. Therefore, k x k convolution is approximated with sum of 1 x k + k x 1 and k x 1 and 1 x k convolutions. This module is called as *Global Convolutional Network* (GCN) in the paper.

Coming to architecture, ResNet(without any dilated convolutions) forms encoder part of the architecture while GCNs and deconvolutions form decoder. A simple residual block called *Boundary Refinement* (BR) is also used.



GCN Architecture. Source.

Benchmarks (VOC2012):

Score	Comment	Source
82.2	-	reported in the paper
83.6	Improved training, not described in the paper	leaderboard

DeepLab v3

Rethinking Atrous Convolution for Semantic Image Segmentation
Submitted on 17 Jun 2017
Arxiv Link

Key Contributions:

- Improved atrous spatial pyramid pooling (ASPP)
- Module which employ atrous convolutions in cascade

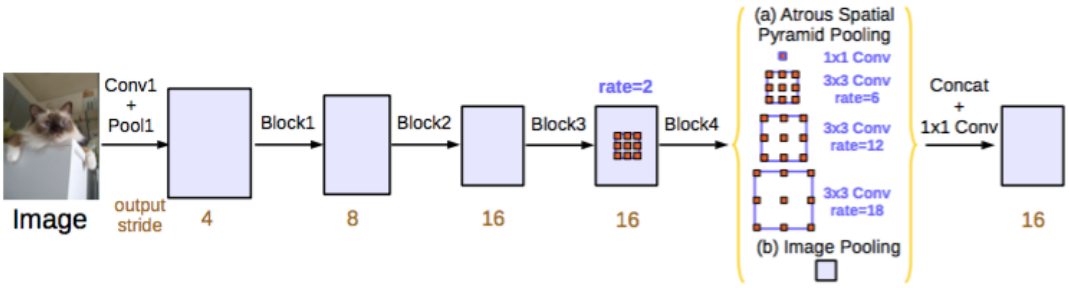
Explanation:

ResNet model is modified to use dilated/atrous convolutions as in DeepLabv2 and dilated convolutions. Improved ASPP involves concatenation of image-level features, a 1x1 convolution and three 3x3 atrous convolutions with different rates. Batch normalization is used after each of the parallel convolutional layers.

Cascaded module is a resnet block except that component convolution layers are made atrous with different rates. This module is similar to context module used in dilated convolutions paper but this is applied directly on intermediate feature maps instead of belief maps (belief maps are final CNN feature maps with channels equal to number of classes).

Both the proposed models are evaluated independently and attempt to combine the both did not improve the performance. Both of them performed very similarly on val set with ASPP performing slightly better. CRF is not used.

Both these models outperform the best model from DeepLabv2. Authors note that the improvement comes from the batch normalization and better way to encode multi scale context.



DeepLabv3 ASPP (used for submission). Source.

Benchmarks (VOC2012):

Score	Comment	Source
85.7	used ASPP (no cascaded modules)	leaderboard

<< Back