Review: Fully Convolutional Networks for Semantic Segmentation

Jiho Bak
School of Electrical and Electronic Engineering, Yonsei University

Abstract

The conventional convolutional classifiers (e.g. VGGnet, GoogLeNet, AlexNet), gives up the positional information due to the fully-connected layer. The paper proposes the application of fully convolutional network for spatially dense prediction, especially for semantic segmentation. Proposed FCN(Fully Convolutional Network) enables pixel-wise classification and arbitrary-size input. The final FCN architecture upsamples the output to remain the size. To make the output finer, the skip layer fusion(with the layer before pooling) is applied. These FCN models can be trained end-to-end and were fine-tunned from the pre-trained models. As expected, FCN shows the state-of-the-art performance on semantic segmentation tasks.

1 Fully Convolutional Network

1.1 1×1 Convolution: Classifier to Dense Prediction

The key to remain spatial information while the classification, is converting the fully-connected layer into 1×1 convolution layer. Then, by combining the channel features of each pixel, pixel-wise classification is done, and it completes the fully convolutional architecture. This also enables arbitrary input size. Unlike the previous models which used many other machineries(e.g. multi-scale pyramid processing, ensembles) for the dense prediction tasks, this FCN's architecture are capable of end-to-end training.

1.2 Patchwise training is Loss Sampling

The FCN architecture is also more efficient in training and inference. Previous convnets used 'patches' for input. However FCN gets the whole input, which prevents the redundant computation due to the intersections of the patches. Since the every patches are included in the input, in perspective of the loss, the patchwise training is just using a part of the total loss. This efficiency of this concept is also tested. Training on whole image was just efficient as sampling the patches for each iterations. However, the convergence was more faster while using less patches or none(Fig 1).

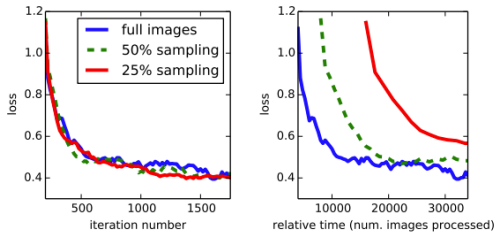

Figure 1: The comparison of convergence, between using whole image or patches.

1.3 Upsampling: Backward Strided Convolution

The spatial size are decreased after several poolings. There exists trade-off in pooling layers' downsampling. Pooling increases spatial invariance and computational advantages with large receptive field. To obtain the fine segmentation result, the output must be upsampled. Previously, the 'shift-and-stitch' techniques [1] are applied, however FCN uses the backward strided convolution, called deconvolution or transposed convolution in case. Obviously, learning the upsampling from encoded features, performs finer segmentation. As a result, the FCN architecture can be divided into convolutional encoder part and deconvolutional decoder part.

1.4 The Skip Layer Fusion

After 5 pooling with stride of 2, spatial information decreases in 32 times. To maintain the advantage of the pooling, upsampling is applied. However, the deconvolution didn't performed enough. The author proposes skip connection from finer resolution, before the pooling. FCN-32s, FCN-16s, FCN-8s, three models are designed(Fig 2). FCN-32s is the basic, using 32x upsampling layer. FCN-16s is upsampling 'stride 16 prediction'. So before the 16x upsampling, 2x upsampled output is summed with the output of 'pool4'(Using two upsampling layers). Recusrively, FCN-8s uses the summation result of FCN-16s', upsampling 'stride 8 prediction'(Using three upsampling layer, two skip connections).

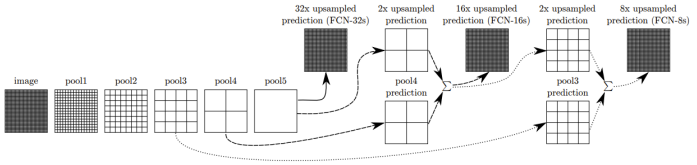

Figure 2: The skip layer fusion architecture.

2 Experiment

AlexNet, VGG-16 and GoogLeNet were tested. The stochastic gradient descent with momentum, weight decay and learning rate scheduling was used for optimizer. The upsampling layers were initialized as bilinear interpolation. Then, all layers are fine-tunned. While fine-tunning, FCN-32s' parameters were used for FCN-16s, and FCN-16s' were used for FCN-8s. Pixel (mean)accuracy, mean IoU(Intersection of Unioin), and frequency weighted IoU were used for metrics. The models are trained and tested mainly on PASCAL VOC 2011 segmentation challenge dataset.
Comparing the FCN-32s, FCN-16s, FCN-8s, obviously, the model with more skip connection from finer resolution performed better(Fig 3).
First of all, VGGNet performed best among classifiers. In PASCAL VOC, FCN-8s model outperformed with 20% relative improvement from previous sota model, SDS and R-CNN. Also in other datasets like NYUDv2, SIFT Flow, FCN showed the state-of-the-art performance.

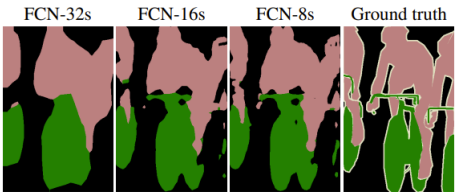

Figure 3: The segmentation results of FCN-32s, FCN-16s, FCN-8s.

Reference

[1] X. Zhang M. Mathieu R. Fergus P. Sermanet, D. Eigen and Y. LeCun. Deep residual learning for image recognition. *ICLR*.