Analysis of Segmentic Segmentation

Nemekhbayar Nomin[1],
[1]Computer Science Engineering Dep. of PNU.

1 Introduction

In this study, we implement and analyze a Fully Convolutional Network (FCN) model, specifically the FCN-8s, for semantic segmentation tasks using VGG16 as the base feature extractor. FCN models have gained popularity for pixel-level predictions due to their fully convolutional structure, allowing for precise segmentation with efficient parameter use.

2 Dataset

The dataset consists of 17127 images along with the corresponding ground truth images taken from PASCAL VOC 2012 dataset.

3 Network Architecture

The FCN-8s model is based on the VGG16 architecture and modified to output pixel-level classifications by adding transposed convolution layers for upsampling. The network consists of the following key components which can also be seen in Figure 1:

- **VGG16 Features**: The model uses VGG16 pretrained on ImageNet to extract deep feature maps.
- **Fully Connected Layers**: Two fully connected layers (fc6 and fc7) are used to process features extracted by VGG16.
- **Prediction Layers**: Convolutional layers are used to produce pixel-wise predictions, with three separate predictions made from different layers: one from the fully connected layers (fc6 and fc7), and the other two from the VGG16 feature maps (pool3 and pool4).
- **Deconvolution (Upsampling)**: A series of transposed convolutions (deconvolutions) are used to upscale the feature maps to the original image size for pixel-wise segmentation.

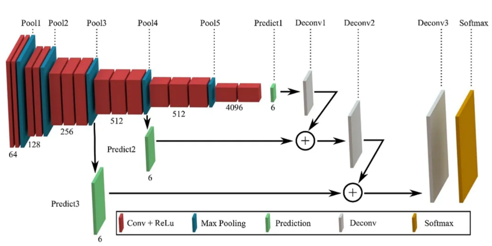

Figure 1: Network Architecture [1]

4 Model Configuration

The training configuration is set up as follows:

- **Optimizer**: Adam optimizer with a learning rate of 0.001.
- **Loss Function**: Pixel-wise cross-entropy loss for semantic segmentation.
- **Augmentation**: Each image undergoes random scaling and horizontal flipping to enhance model robustness.
- **Iterations**: The model is trained over 20,000 iterations rather than epochs due to efficient streaming of images with batch size 1.

5 Training Analysis

The training process was executed for 20,000 iterations, with tracking of the average loss per 100 iterations. Data augmentation techniques including random scaling and flipping were applied at each iteration to reduce overfitting. The use of skip connections in the FCN-8s architecture allowed for better localization, as feature maps from different layers were combined to capture both low- and high-level information.

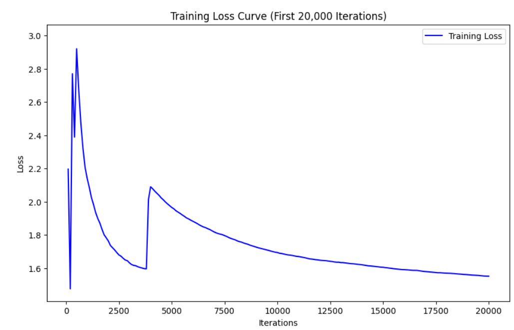

Figure 2: Training loss curve over 20000 iterations.

6 Evaluation Metrics and Results Analysis

The evaluation focused on pixel-wise accuracy and mean Intersection over Union (mIoU) as metrics for segmentation performance. FCN-8s achieved satisfactory segmentation quality across classes, especially in regions with distinct boundaries, owing to skip connections that retain spatial details from earlier layers.

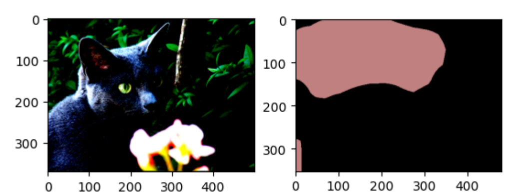

Figure 3: Segmentation Visual Example.

7 Discussion

While the FCN-8s model with VGG16 as the base network shows promising segmentation results, further improvements could involve using deeper backbones, such as ResNet, for enhanced feature extraction, or refining the upsampling process to reduce spatial artifacts. Moreover, increasing the batch size or incorporating multi-scale inputs may improve model stability and performance.

8 Conclusion

The FCN-8s model proves to be effective for semantic segmentation tasks, balancing computational efficiency with pixel-level accuracy. With additional optimizations, FCN models can be powerful tools for a range of image segmentation applications.

[1] Wilbur Des. Semantic segmentation using fully convolutional neural networks. https://medium.com/@wilburdes/semantic-segmentation-using-fully-convolutional-neural-networks-86e45336f99b, Oct 5 2018. Available at: https://medium.com/@wilburdes/semantic-segmentation-using-fully-convolutional-neural-networks-86e45336f99b.