Final Project Report

Zhixu Tao
Department of Mathematics
University of Toronto

Xiaoli Yang
Department of Mathematics
University of Toronto

Abstract

In this final project, we will focus on semantic segmentation. We explore two existing models SegNet and DeepLabv3+ which are specifically designed for the task of semantic segmentation. We describe architectures of two models and some notable characteristics. We also show the performance of SegNet is much worse than DeepLabv3+ by running experiments on three small datasets that we manually constructed from Pascal VOC 2012.

1 Introduction

Semantic segmentation, which assigns semantic labels to each pixel and partitions a digital image into several segments, has become one of the most fundamental tasks in computer vision. Before the appearance of deep Convolutional Neural Networks (CNN), semantic pixel-wise segmentation relied on classical techniques such as Gray Level Segmentation or Conditional Random Fields to detect hand-crafted features. These techniques only produced coarse results. In order to get finer results, more accurate localization and detection are desired. Thus, the deep CNN has come into play, and the extraordinary success in deep CNN has fuelled the progress of semantic segmentation tremendously.

One of the earliest deep CNNs used in semantic segmentation is Fully Convolutional Network (FCN) [17] which purely consists of convolution layers. However, one issue in FCN is that the output images are in low resolution, resulting in blurry object boundaries. In order to solve this issue, several more advanced FCN-based models have been proposed. In this work, we consider two state-of-the-art FCN-based models, namely SegNet [1] and DeepLabv3+ [5]. We run these two models on three datasets with different characteristics to analyze their performance. The remainder of the report is organized as follows. In section 2, we review several related work. In section 3, we outline and analyze architectures of SegNet and DeepLabv3+. In section 4, we describe our experiment details and results.

2 Related work

Before the arrival of models designed specifically for semantic segmentation, there were several attempts to apply models used for object categorization to segmentation such as [8, 10, 11]. Later, models designed particularly for semantic segmentation based on FCNs such as SegNet [1], U-Net [19], ParseNet [16], Feature Pyramid Network [15], Mask R-CNN [13], and DeepLab [3] have advanced this research area. The general architecture of these models can be thought as an encoder followed by a decoder. The encoder-decoder networks have shown success in many computer vision tasks [9, 18, 15, 20]. Usually, the encoder is a pre-trained classification network and the decoder is used to project features learnt by encoder in low resolution onto pixel space in high resolution. The research of semantic segmentation has also been motivated by challenging datasets such as The Pascal Visual Object Classes (VOC) [7], Common Objects in Context (COCO) [14], and The Cambridge-driving Labeled Video Database (CamVid) [2].

Preprint. Under review.

3 Architecture

In this section, we briefly discuss the architecture of SegNet and DeepLabv3+. Both models have the encoder-decoder structure. SegNet is motivated by road scene understanding applications, while DeepLabv3+ is designed to deal with the challenge imposed by multi-scale contextual information and sharper object boundaries contained in the image.

3.1 SegNet

SegNet employs an encoder-decoder network structure, followed by a pixel-wise classification layer (Figure 1). Its encoder network is identical to the 13 convolutional layers of the VGG16 network [21]. Each convolutional layer in the encoder network produces a set of feature maps. Following each convolutional layer, batch normalization and ReLU are applied. Then, max-pooling with 2×2 filter and stride 2 is performed and then the output is sub-sampled by a factor of 2. One distinct characteristic about this encoder is that it stores max-pooling indices, the indices of maximum feature value. Each encoder convolutional layer has a corresponding decoder layer. The decoder network upsamples the feature maps by using max-pooling indices from the corresponding encoder layer as shown in Figure 2. The architecture of SegNet is similar to U-Net [19], but U-Net does not re-use the max-pooling indices.

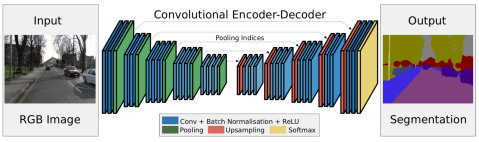

Figure 1: The network architecture for SegNet [1]

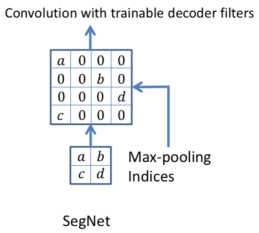

Figure 2: Maxpooling [1]

3.2 Deeplabv3+

The series of DeepLab contains DeepLabv1 [3], DeepLabv2, DeepLabv3 [4] and DeepLabv3+ [5] where DeepLabv3+ has been the most advanced model so far. The first notable improvement of DeepLabv3+ is that it employs Atrous Spatial Pyramid Pooling (ASPP) modules at pooling layer in the encoder network. In ASPP, atrous convolution is combined with Spatial Pyramid Pooling Net (SPPNet) [12]. Atrous convolution creates "holes" in the filters by inserting zeros to enlarge the field-of-view of the filters. Then these filters can capture more contextual information without training more parameters. The main advantage of atrous convolution is that by inserting zeros to enlarge the size of the filter, we can avoid significant reduction in spatial resolution caused by down-sampling in traditional deep CNN. On the other hand, SPPNet has multiple pooling layers with different scales to capture multi-scale contextual information. Therefore, using ASPP as pooling layer can handle reduced feature resolution and existence of multi-scale objects in the image at the same time. Figure 3 shows an example of ASPP consisting of four atrous convolutional layers.

The second notable improvement of DeepLabv3+ is the use of a simple but efficient decoder. Figure 4 shows the complete encoder-decoder structure of DeepLabv3+. Features from the encoder are first upsampled by a factor of 4. Low-level features from the network backbone DCNN are convoluted with a 1×1 filter to reduce the number of channels. Then the decoder concatenates them together and a 3×3 convolution is applied, followed by an upsampling by a factor of 4. This decoder successfully recovers sharper object boundaries, resulting in more accurate segmentation.

The last promising improvement is the use of modified aligned Xception model as the network backbone. However, due to the limit of computational ability, we did not use Xception in our training. We still adopted ResNet-101 as the network backbone. We refer readers to [5], [6] for more details.

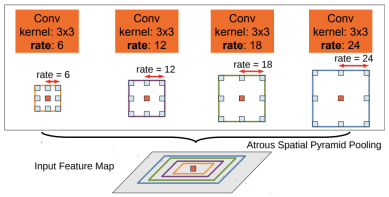

Figure 3: Atrous Spatial Pyramid Pooling [3]

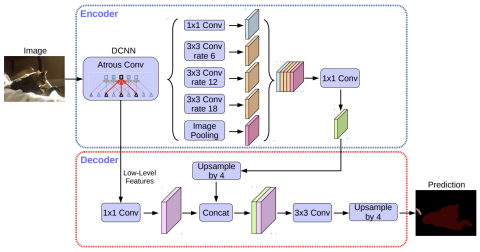

Figure 4: The Architecture of DeepLabv3+ [5]

4 Experiments and Results

In this section, we introduce the experiments that we did to analyze the performance of SegNet and DeepLabv3+. We describe three datasets that we used during the experiments, metrics that were employed to evaluate the performance, and the final experiment results. We will see that DeepLabv3+ performs much better than SegNet.

4.1 Datasets

Three datasets that we used were extracted from Pascal Visual Object Classes 2012 (VOC2012) [7]. The Pascal VOC2012 dataset contains 17,125 labelled images which are divided into 20 classes, covering from person, animal, vehicle to indoor objects. Due to the limit of computational ability, we did not use the full dataset to train. Instead, we constructed three small datasets from Pascal VOC2012, and three datasets serve different purposes. The first dataset Aeroplane contains purely images of aeroplane. This dataset intends to test the ability of single-class semantic segmentation of these two models. The second dataset Vehicle consists of images from four categories: bus, train, car and motorbike. This dataset intends to test the multi-class semantic segmentation ability. The third dataset Person consists of images in category person. However, these images are carefully selected so that people in the image present at multiple scales. This dataset favours the ability of DeepLabv3+ to capture information at multiple scales. See Appendix A for the size of three datasets.

4.2 Evaluation Metrics

The easiest and most straightforward metric to evaluate a model is pixel accuracy which measures the percentage of pixels that are correctly segmented. Suppose we are considering pixel accuracy for a specific class. If True Positive (TP) represents pixels that are correctly classified as belonging to this class, True Negative (TN) represents pixels that are correctly classified as not belonging to this class, False Positive (FP) and False Negative (FN) represent false cases correspondingly, then the pixel accuracy for this class is given by

$$\text{pixel accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}.$$

However, this metric may give misleading results if the object belonging to this class is small in the image. Therefore, we also consider another metric Intersection over Union (IoU). IoU measures the percentage of overlap between the target and the prediction image, i.e.,

$$\text{IoU} = \frac{\text{target} \cap \text{prediction}}{\text{target} \cup \text{prediction}}.$$

Class IoU measures the percentage of overlap for one particular class, and mean IoU (mIoU) averages class IoU over all classes.

4.3 Training Process

For both SegNet and DeepLabv3+, we use pretrained ResNet-101 as the network backbone. We use Stochastic Gradient Descent with weight decay in all training processes. The initial

learning rate for SegNet is 0.001 while the initial learning rate for DeepLabv3+ is 0.007. All training processes ran for 50 epochs due to the limit of computational ability.

4.4 Results

Table 1 and Table 2 summarize the validation and test class IoU of SegNet and DeepLabv3+ on three datasets. The results produced by SegNet are significantly worse than what we expect. In both validation and test, SegNet cannot successfully segment any classes. There are three main reasons. First, the size of the dataset is too small. Compared to the full Pascal VOC 2012 dataset which contains more than $17,000$ images, our dataset contains no more than 200 images. Second, we only trained the model for 50 epochs, while in [1], the model was trained on a dataset for more than 100 epochs. Third, the most important reason is that SegNet is motivated by road scene understanding. Road scene understanding is different from segmenting Pascal VOC 2012. In segmenting Pascal VOC 2012, most images have one or two distinct foreground classes surrounded by a highly varied background [1], so the segmentation task is more or less similar to the classification task. However, in the road scene understanding applications, the segmentation is more smooth since typically, there is no distinct foreground object, and most pixels may belong to large classes such as roads or building. See Appendix C for a comparison.

The results of DeepLabv3+ on three data sets are much better, though still far from being optimal. For the dataset Vehicle, the highest validation class IoU is 0.77 from category Background, while the lowest is 0.42 for categoty Motor. Similar for Vehicle's test results, the only two positive class IoUs are 0.72 and 0.70 for categories Car and Background, repsetively. As mentioned above, the small dataset size might be one reason for the decreased performance. Another reason is that since there are four classes in dataset Vehicle, imbalance between the number of examples provided for each class might have led to an imbalance in class IoUs. Finally, due to the limit of computational ability, we were not able to train DeepLabv3+ using Xception as our backbone model. This might be responsible for the overall decrease in our model's performance. On the other hand, for datasets Person and Aeroplane, even though the validation results are not optimal, our DeepLabv3+ model achieved class IoUs of more than 0.70 on all non-background categories. This is a good indicator that even with presented limitations, DeepLabv3+ is still a very promising and generalizable model in segmenting single-category images with multiple scales.

Table 1: Validation Class IoU

	Vehicle					Aeroplane		Person	
	Background	Train	Bus	Motor	Car	Background	Aeroplane	Background	Person
SegNet	0.56	0	0	0	0	0.81	0	0.46	0
DeepLabv3+	0.77	0.52	0.62	0.42	0.60	0.90	0.60	0.71	0.64

Table 2: Test class IoU

	Vehicle					Aeroplane		Person	
	Background	Train	Bus	Motor	Car	Background	Aeroplane	Background	Person
SegNet	0.37	0	0	0	0	0.88	0	0.48	0
DeepLabv3+	0.70	0	0	0	0.72	0.97	0.80	0.61	0.71

We also provide validation pixel accuracy, validation mIoU, and test mIoU in Appendix B to show that the performance of SegNet is worse. Moreover, in Appendix D, we provide a successful segmentation result and a failed segmentation result produced by DeepLabv3+.

5 Conclusion

In conclusion, the performance of SegNet is much worse than DeepLabv3+. This suggests that besides the fact that SegNet is not sutible for handling Pascal VOC 2012, more importantly, DeepLabv3+ is the more state-of-the-art model.

6 Acknowledgment

We would like to thank the GitHub user **yassouali** for providing his code of implementation at `https://github.com/yassouali/pytorch_segmentation/tree/8b8e3ee20a3aa733cb19fc158ad5d7773ed6da7f`. Our code is edited based on his implementation. For this final project, Zhixu Tao and Xiaoli Yang contribute equally to implementing, modifying, and debugging two models as well as writing and proofreading each section of the report.

References

[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[2] Gabriel J Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30(2): 88–97, 2009.

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[4] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[6] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[7] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015.

[8] Clément Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Scene parsing with multiscale feature learning, purity trees, and optimal covers. *arXiv preprint arXiv:1202.2160*, 2012.

[9] Cheng-Yang Fu, Wei Liu, Ananth Ranga, Ambrish Tyagi, and Alexander C Berg. Dssd: Deconvolutional single shot detector. *arXiv preprint arXiv:1701.06659*, 2017.

[10] Carlo Gatta, Adriana Romero, and Joost van de Veijer. Unrolling loopy top-down semantic feedback in convolutional deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 498–505, 2014.

[11] David Grangier, Léon Bottou, and Ronan Collobert. Deep convolutional networks for scene parsing. In *ICML 2009 Deep Learning Workshop*, volume 3, page 109. Citeseer, 2009.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916, 2015.

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[15] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[16] Wei Liu, Andrew Rabinovich, and Alexander C Berg. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015.

[17] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[18] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016.

[19] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[20] Abhinav Shrivastava, Rahul Sukthankar, Jitendra Malik, and Abhinav Gupta. Beyond skip connections: Top-down modulation for object detection. *arXiv preprint arXiv:1612.06851*, 2016.

[21] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

A Appendix: Size of Dataset

Table 3: Dataset Summary

	Vehicle	Aeroplane	Person
Train	100	100	100
Validation	40	25	23
Test	40	25	23

B Appendix: Pixel Accuracy and mIoU

Table 4: Validation Pixel Accuracy and mIoU

	Vehicle		Aerolane		Person	
	Acc	mIoU	Acc	mIoU	Acc	mIoU
SegNet	0.56	0.03	0.81	0.04	0.46	0.03
DeepLabv3+	0.80	0.14	0.90	0.07	0.73	0.14

Table 5: Test mIoU

Model\mIoU	Vehicle	Aeroplane	Person
SegNet	0.018	0.042	0.023
DeepLabv3+	0.064	0.084	0.063

C Appendix: Comparison between CamVid Road Scene and Pascal VOC 2012

Figure 5: Original road scene image from CamVid [2]

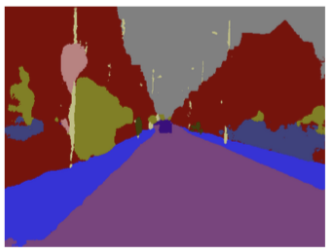

Figure 6: Segmentation result

Figure 7: Original image from Pascal VOC 2012

Figure 8: Segmentation result

D Appendix D: Segmentation Examples

Figure 9: Failure example
(a) original picture (b) ground truth (c) model segmentation

Figure 10: Success Example
(a) original picture (b) ground truth (c) model segmentation