Learning based Methods for Semantic Segmentation

Navami Kairanda
2577665
s8nakair@stud.uni-saarland.de

Priyanka Mohanta
2577684
s8prmoha@stud.uni-saarland.de

Abstract

Extracting semantically meaningful information from images is a fundamental problem in computer vision. In the past decade, a number of learning based methods are proposed to make dense semantic label predictions and such methods are often trained on large annotated dataset of real images. In this paper, we explore several deep convolutional network architectures including Fully Convolutional Network, U-net, Recurrent Residual U-net and DeepLabV3 that provide state-of-the-art results in semantic segmentation. We train these networks on PASCAL VOC and Cityscapes datasets and achieve best test accuracy of 79% and 90.3%, respectively. Additionally, we compare and review the strengths and weakness of these neural architectures supported by their performance on the aforementioned datasets.

1 Introduction

Image segmentation is an indispensable task of computer vision process, that involves segregating an image on pixel level into multiple regions based on their respective properties. The goal of the task is to use the spatial information of the pixel to generate a semantically meaningful interpretation that can be analyzed easily. Image segmentation can be seen in two types, namely instance segmentation and semantic segmentation. Combination of the two types creates another type called panoptic segmentation.

In this paper, we review several state-of-the-art methods [14] for learning segmentation tasks from large annotated real datasets. Our specific contributions include the following

- Explores elaborately several CNN based state-of-the-art segmentation models including Fully Convolutional Networks, U-Net, Recurrent U-Net, Residual U-Net, Recurrent Residual U-Net and DeepLabv3
- Qualitatively and quantitatively compares the performance of these state-of-the-art models on PASCAL VOC and Cityscapes datasets

2 Related Work/ Technical Background

Semantic segmentation is a pixel level classification of the input image , that is each pixel is associated with class label of the object or region it is enclosed in[16].The approach for this technique involves learning the class label of the pixel as well as its localization in the input to give context about the image. Irem Ulku et al.[15] have presented a survey on the various architectures used for semantic segmentation over the years. They discuss how the success of CNN in creating semantic has boosted interest in semantic segmentation. And how this has resulted in a number of published survey studies.

Instance segmentation, unlike semantic segmentation segregates multiple objects belonging to the same class. That is, separate instances of the same object class are assigned different labels. It is

Preprint. Under review.

a solution for both object recognition and semantic segmentation. Hafiz et al.[6] provide a survey, discussing the various techniques , evolution and state of art for instance segmentation.

Panoptic segmentation implements both instance and semantic segmentation. The output consists of two channels, one stating the object class of the pixel and the second channel states the instance of the object class. Sultana et al.[14] Discuss briefly about the state of art methods and the datasets used for panoptic segmentation in their survey.

Convolutional Networks for Segmentation In general, semantic segmentation focuses on deriving fine inference from coarse feature maps. As mentioned in section 2.1, the success of Convolutional networks(CNN) to extract features while maintaining spatial integrity led to boost in interest for image segmentation. CNN filters learn features that are shift invariant. Pinheiro et al[12] demonstrate in their paper how recurrent CNN is used for scene parsing.

Each stage produces a two dimensional output as a result of applying the convolutional filter to the entire image. These outcomes are called feature maps and are followed by a nonlinear squashing function.[10] Therefore each convnet filter has convolution, pooling and activation function as basic components. Since, the prediction takes place at pixel level, a dense network is required with large number of parameters. This continuous pooling and striding used in convnet to extract features leads to reduced feature resolution and unable to detect small objects. As a solution to these drawbacks, modifications to the CNN model were introduced. Which led to Fully Conventional Networks(FCN)[8]. FCN model removes the fully connected convolution layers to The FCN model is discussed in detail in section 3.1.

Encoder Decoder networks are currently common architectures used among semantic segmentation algorithms.[10][17][2] As the name suggests the architecture consists of an encoder, which is usually any classic image segmentation algorithm. (For example VGG, Resnet, etc) Encoders are usually a pretrained model to extract features from the input. Feature extraction performed by encoders are at lower resolution. Decoders are used to project these features to higher resolution. A decoder combines layers to perform deconvolution and upsampling to arrive at the input image dimension.

Atrous convolution addresses the drawback of traditional CNN of low feature resolution. As a solution this method adds holes in between convolution layer kernels. It increases the field of view of kernels keeping the computational cost same. [3] Atrous convolution formula for a single dimension can be given as :

$$y[i] = \sum x[i + r.k]w[k]$$

Here, 'x' represents the feature map, 'w[k]' is the filter with k as the length of the filter. 'r' is atrous rate and signifies 'r-1' zeros(holes) between consecutive filters.

3 Methodology

This section summarizes the techniques or methods adopted in different tasks of the project. Task 1 follows the architecture presented by Long et al.[10] in their paper. It is based on advantages of using fully convolutional network for semantic segmentation. Task 2 implements the Recurrent Residual U Network discussed in [2] by Pinheiro et al. For task 3, to improve the results obtained in task 2 DeepLab3 architecture is used.

3.1 Fully convolutional Network (FCN)

A fully convolutional network trained at pixel level for semantic segmentation exceeds performance of state-of-the-art. [10]The convnet takes data inputs with three dimensions (h,w and d). 'h' and 'w' are the dimension of the input and 'd' is for color channels (for rgb, 'd' equals 3). As mentioned in section 2.2, the outcome 'yij' obtained after applying the convnet filter at location 'i','j' for the image 'x' is given as:

$$y_{ij} = f_{ks}(\{x_{si+\delta i, sj+\delta j}\}_{0<=\delta i, \delta j<=k})$$

'k' stands for kernel size and 's' is the stride. Nets composed of layers as in equation, produce outputs for inputs of particular dimensions. However FCN includes only the convolution and pooling layer that accepts inputs of arbitrary sizes and produces corresponding output. FCN adopts the current semantic segmentation architecture of encoders and decoders. With respect to the model used in the project, the encoder is a pretrained vgg16 model which is converted from fully connected to

FCN by appending an 1x1 convolutional layer at the end. The decoder is layers of deconvolution and upsampling performed at different max pool levels of vgg16 model. Thus the decoder ensures combination of lower and higher resolution feature maps. These are known as skip connections.

3.2 U Network

As from the name, it is a 'U' shaped network for semantic segmentation. It consists of a contracting encoder and an expansive decoder. Encoder of unet consists of blocks made of basic convolution, ReLU nonlinearity and max pool for downsampling. After every block the feature informations expands but spatial information contracts. The decoder executes upsampling on the feature map, therefore it consists of deconvolution operations. The U-net architecture preserves the contextual information of the image.

3.3 Recurrent Residual U Network(R2U-Net)

R2U-Net is a variant of U-Net, which combines the strength of both U-net, recurrent convolutional layer(RCL) and residual connectivity. These nets are differentiated on the number of RCL layers that follow a convolution layer. Encoder- decoder architecture of R2U-Net is similar to the U-Net, and the only difference is the replacement of continuous convolutional layers with RCLs containing residual units. These replaced unit support feature accumulation.

3.4 DeepLab

DeepLab architecture uses atrous convolution and fully connected conditional random field to capture efficient dense predictions without losing finer details. Progressively, DeepLab2 replaces atrous convolution with Atrous spatial pyramid pooling (ASPP). Where atrous convolution is applied to the feature map with different rates and combined. In DeepLab3 the connected, conditional random field is removed and Deeplab2 is restructured. The efficient performance of this architecture has made it state of art model.

4 Experimental Setup

4.1 Datasets

PASCAL VOC We evaluate FCN and Unet methods on the PASCAL VOC 2012 semantic segmentation benchmark [5] which contains 20 foreground object classes and one background class. The original dataset contains 1464, 1449, and 1456 pixel-level labeled images for training, validation, and testing, respectively.

Cityscapes Cityscapes[4] is a large-scale dataset containing high quality pixel-level annotations of 5000 images (2975, 500, and 1525 for the training, validation, and test sets respectively) and about 20000 coarsely annotated images. Following the evaluation methods used by previous methods, 19 semantic labels are used for evaluation without considering the void label.

4.2 Training details

We train our networks using PyTorch [11] with cross-entropy loss, and use the Adam[9] optimizer with minibatches of size 16. Training epochs are set to 300 with a learning rate of 0.001, and we exponentially decay this value by gamma=0.96 every epoch. We initialize the VGG[13] backbone of Fully Convolution Network[10] and DeepLabV3 pretrained with ImageNet dataset. Our DeepLabV3 with resnet[7] as the base network is initialized with weights pretrained on COCO dataset. Most network we exploread takes about 4 6 hours to train on 2 NVIDIA RTX8000 GPUs. [1]

In our implementation, the train splits of PASCAL VOC and Cityscapes are used for training various models and validation set is used for evaluation, as labels for test set are not publicly available for either datasets. While both datasets are augmented with additional data/coarse annotations in subsequent works after the original publication, we use only the original and fine labels for training and inference.

[1]Link to our git page: https://github.com/navamikairanda/R2U-Net.git

Method	Accuracy	mIoU	F1-score	Sensitivity	AUC-ROC
FCN-VGG16 (train)	95.4	78.5	68.5	86.9	0.998
U-net (train)	**98.7**	**93.9**	**81.8**	**96.8**	**1.00**
FCN-VGG16 (test)	77.9	**20.5**	**24.7**	**28.8**	**0.908**
U-net(test)	**79.0**	19.6	23.9	24.8	0.868

Table 1: Experimental results of Fully Convolutional Network and U-Net for semantic segmentation on *PASCAL* training and test set

4.3 Evaluation Metrics

Performance of all models are measured using several quantitative metrics including pixel accuracy, recall, F1-score/Dice coefficient, and Intersection-over-Union (also known as Jaccard similarity). We calculate these metrics using the following equations,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Recall = \frac{TP}{TN + FP} \tag{2}$$

$$F1 - Score = \frac{2|GT \cap SR|}{|GT| + |SR|} \tag{3}$$

$$mIoU = \frac{|GT \cap SR|}{|GT \cup SR|} \tag{4}$$

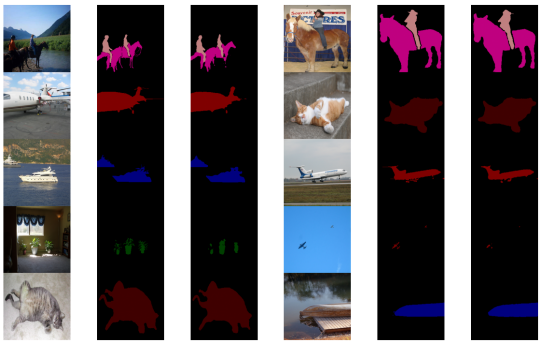
where the variables True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) have their usual meaning. Except in the case of pixel accuracy, the metrics are compputed per class and averaged across all the classes present in the dataset. Additionally, we consider the metric AUC-ROC defined by the area under the Receiver Operating Characteristic curve.

5 Results and Discussion

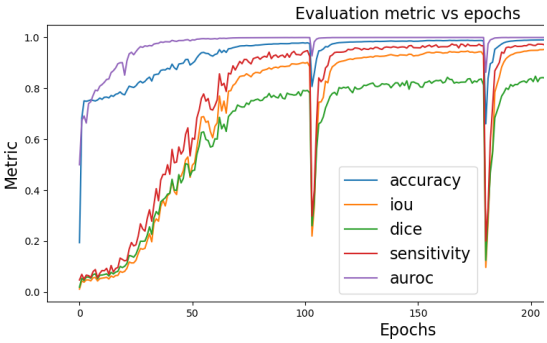
5.1 FCN: Fully Convolutional Network

FCN is arguably the first successful network architecture for the task of semantic segmentation. They define a skip architecture that combines deep, coarse, semantic information and shallow, fine, appearance information. In Fig. 1a we show qualitative results of this model on PASCAL VOC dataset. The network efficiently learns to make dense predictions as supported by the numbers in Table 1 and plot Fig. 1b

5.2 Recurrent Residual Convolutional U-Networks

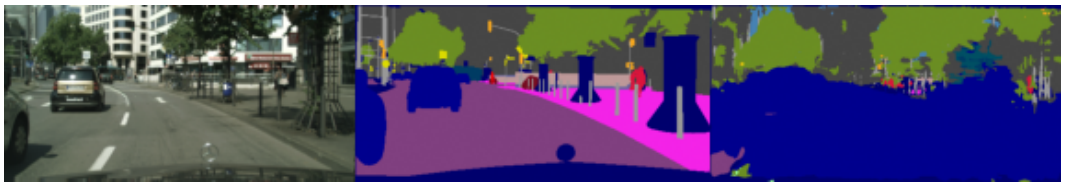


(a) Images along the with their target and predicted segmentation mask for randomly sampled dataset instances

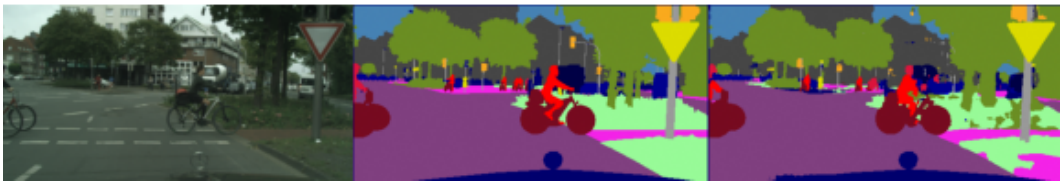


(b) Evaluation metrics values on training dataset computed after each epoch

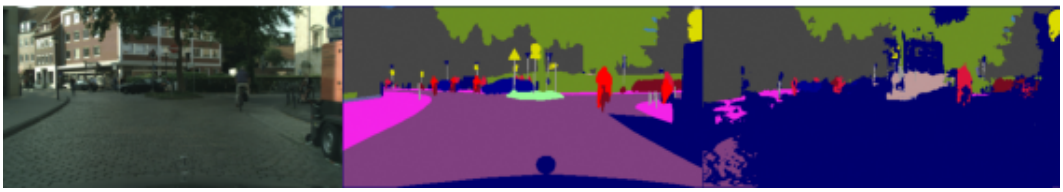
(a) U-Net



(b) Recurrent U-Net



(c) Residual U-Net



(d) Recurrent Residual U-Net

Figure 3: **U-net and derived architectures for Cityscapes segmentation** Here, left column shows input image, target mask is shown at the centre and the predicted segmentation at the right.

U-Net with it's convolutional encoding and decoding units is a popular approach for semantic segmentation. This model allows for the use of global location and context at the same time and shows best performance among all architectures in our experiments as supported by Fig 3a and Table 2

Next, we explore residual version of U-Net. Often, training very deep models is challenging due to the vanishing gradient problem. Toward this end, a residual unit helps when training deep architecture

Method	Accuracy	mIoU	F1-score	Sensitivity	AUC-ROC
U-net	**90.3**	**57.5**	**63.8**	**68.0**	0.985
Recurrent Unet	71.4	20.4	26.8	26.8	0.914
Residual Unet	89.9	55.9	63.2	66.6	**0.987**
R2Unet (t=3)	67.7	20.7	27.7	29.0	0.906
R2Unet (t=3)	73.4	32.0	39.1	42.1	0.964

Table 2: Experimental results of U-Net and networks structures based on U-Net on emphcityscapes test set

Method	Accuracy	mIoU	F1-score	Sensitivity	AUC-ROC
FCN-VGG16	87.3	46.1	53.2	55.7	0.980
DeepLabv3-VGG16	85.4	43.6	50.0	53.2	0.976
DeepLabv3-resnet	**88.0**	**50.5**	**56.2**	**60.5**	**0.983**

Table 3: Performance of further network architectures explored as part of challenge task on emphcityscapes test set

by providing an identity mapping to facilitate the gradient back propagation. This helped us to train faster and achieves comparable visual and quantitative performance to U-Net Fig. 3c and Table 2

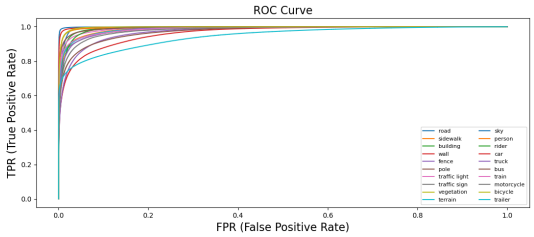
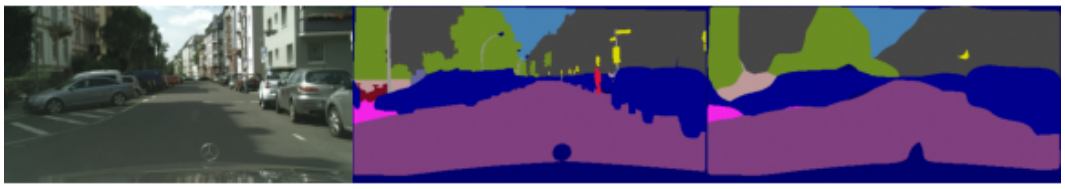


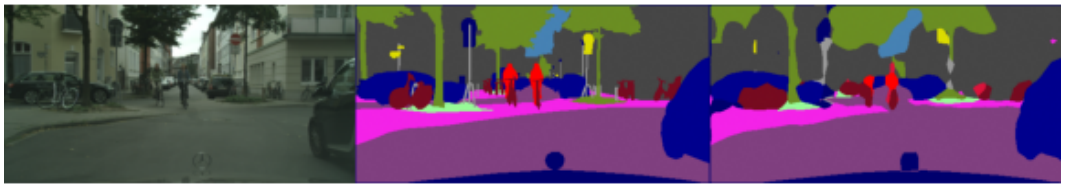
Figure 2: ROC curve for U-Net model on Cityscapes test set plotted separately for each class

Recurrent Convolutional Networks are successful for object recognition task [1] where feature accumulation through recurrent layers can extract high level semantics (class label) of the input image. However, in our experiments we observe that recurrent convolutions hurts the performance in the case of multi-class segmentation, as shown in Fig. 3b, Fig. 3d and Table 2. We believe that that the fine details learned with convolutional filters are lost with recurrent connections.

5.3 DeepLab: Atrous Convolution Networks



(a) VGG backbone



(b) Resnet backbone network

Figure 4: **DeepLabV3 models for Cityscapes segmentation** Here, left column shows input image, target mask is shown at the centre and the predicted segmentation at the right. DeepLabv3 with VGG as base network fails to learn fine structures whereas restnet shows improved local details

In our effort towards further improving the object segmentation challenge, we explore two more architectures, namely FCN and Atrous Convolution based DeepLabV3. DeepLabv3 is an interesting approach that adjusts convultional filter's field of view, thereby learning to segment objects at various scales. In our experiments, we tried two different variations of this model with different backbone networks. In the first case, where we use ImageNet pretrained VGG with DeepLabV3, the segmentations predicted are too coarse as demonstrated by 4a. In order to overcome this, we opt for a resnet backbone network that allows local appearance information to propagate through residual connections. This model shows superior performance in comparison as shown by Fig. 4b and Table 3

6 Conclusion

Semantic segmentation of images is an important and well explored problem in computer vision. With the explosion and success of neural networks in the recent past, a number of learning based methods are proposed to make dense semantic label predictions and such methods are often trained on large annotated dataset of real images. Here, we presented a review of several such methods while providing detailed training and testing scripts to reproduce the results. Additionally, we empirically compare performance of popular class of convolutional networks used for image segmentation, namely FCN, U-Net and DeepLabv3. We observe that best visual as well as meanIoU/pixel accuracy results are achieved by U-Net.

References

[1] Md Zahangir Alom, Mahmudul Hasan, Chris Yakopcic, and Tarek M. Taha. Inception recurrent convolutional neural network for object recognition, 2017.

[2] Md Zahangir Alom, Mahmudul Hasan, Chris Yakopcic, Tarek M Taha, and Vijayan K Asari. Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation. *arXiv preprint arXiv:1802.06955*, 2018.

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

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[5] 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.

[6] Abdul Mueed Hafiz and Ghulam Mohiuddin Bhat. A survey on instance segmentation: state of the art. *International Journal of Multimedia Information Retrieval*, pages 1–19, 2020.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[8] Youngeun Kim, Seunghyeon Kim, Taekyung Kim, and Changick Kim. Cnn-based semantic segmentation using level set loss. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1752–1760. IEEE, 2019.

[9] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[10] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. *arXiv e-prints*, page arXiv:1411.4038, November 2014.

[11] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.

[12] Pedro Pinheiro and Ronan Collobert. Recurrent convolutional neural networks for scene labeling. In *International conference on machine learning*, pages 82–90. PMLR, 2014.

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

[14] Farhana Sultana, Abu Sufian, and Paramartha Dutta. Evolution of image segmentation using deep convolutional neural network: a survey. *Knowledge-Based Systems*, 201:106062, 2020.

[15] Irem Ulku and Erdem Akagunduz. A survey on deep learning-based architectures for semantic segmentation on 2d images. *arXiv preprint arXiv:1912.10230*, 2019.

[16] Yunchao Wei, Xiaodan Liang, Yunpeng Chen, Zequn Jie, Yanhui Xiao, Yao Zhao, and Shuicheng Yan. Learning to segment with image-level annotations. *Pattern Recognition*, 59:234–244, 2016. Compositional Models and Structured Learning for Visual Recognition.

[17] Yongfeng Xing, Luo Zhong, and Xian Zhong. An encoder-decoder network based fcn architecture for semantic segmentation. *Wireless Communications and Mobile Computing*, 2020, 2020.