**Seminar** **WS 2021**
**Prof. Ph. D. Tomas Skopal** **Universität Passau**
**Fakultät für Informatik und Mathematik**

# Semantic Segmentation And Object Detection In Autonomous Vehicles

**Fariha Hossain Tithy**
**82255**

**March 2021**

**Abstract**

Autonomous vehicles are developing rapidly not only in better computer vision for resolving conventional driving tasks but also working on significantly large datasets that sustain training of such techniques. State-of-the-art techniques for solving the driving tasks are object detection and semantic segmentation.The importance of semantic segmentation and object detection is to avoid obstacles during self-driving. This report explains how semantic segmentation and object detection are implemented in autonomous vehicles.Semantic segmentation was carried out with cityscapes dataset using Unet Model and object detection using darknet on own images captured from the streets of Passau (ZOB) using Pytorch in Google colab and a few images downloaded from google.Object detection implementation was done in YoloV4 using darknet model that contains helper function for uploading a stored image directly from the computer and performs object detection on it. The results of semantic image segmentation showed promising results with an IOU of 82 percent which may be rated as relatively good and the results from the object detection implementation showed good results on the objects detected but did not detect everything such as trees or buildings which are crucial.

## 1 Introduction

### 1.1 Semantic Segmentation

Semantic segmentation defines the procedure of connecting each and every pixel of an image with a class label which is defined as a natural step in the development from coarse to fine inference. This is where the source is situated at classification where the whole input is made from predictions, detection presents classes and also further information for the spatial location of those classes and semantic segmentation achieves fine-grained interpretation by creating condensed predictions indicating labels for every single pixel so that each pixel is observed with the class of its enclosing object. Almost all of the work in semantic segmentation take on encoder-decoder structural design which lessens spatial resolution of the feature maps, and up samples them with learned transposed convolution, fixed bi-linear interpolation, or unpooling and the dilated convolutions or spatial pyramid pooling expand the approachable field and enhance the accuracy.

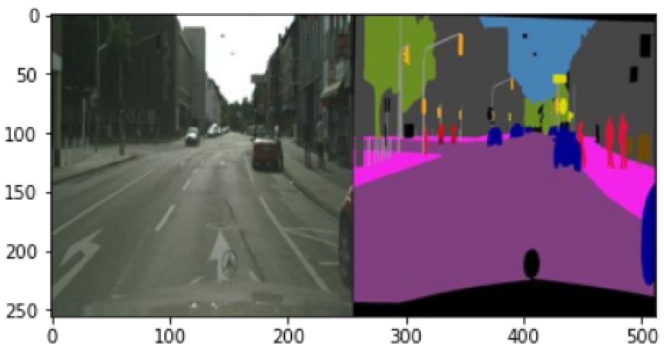

Figure 1: Showing semantic segmentation of one of the images from cityscapes dataset.Image includes the original image from the dataset and its corresponding output pair in the same image before splitting

## 1.2 Object Detection

This is a computer skill that is associated to computer vision and processing of the images and videos that detects instances of semantic objects of a specific class such as pedestrian detection, car detection or detecting anything that is in the way of the car which helps the car to make a decision on where to turn such as either turning left or turning right. This helps to avoid obstacles during self-driving and other road problems.In this report, Object detection implementation is explained where it was carried out using YoloV4 (You Look Only Once version 4).

**YOLO (You Look Only Once)**

YOLO is trendy since it accomplishes high accuracy and running in real-time. The method implies that the image is looked at only once and then it makes predictions as it by-passes through forward propagation. After it only detects the object once, the recognized objects are output with the bounding boxes.[2]

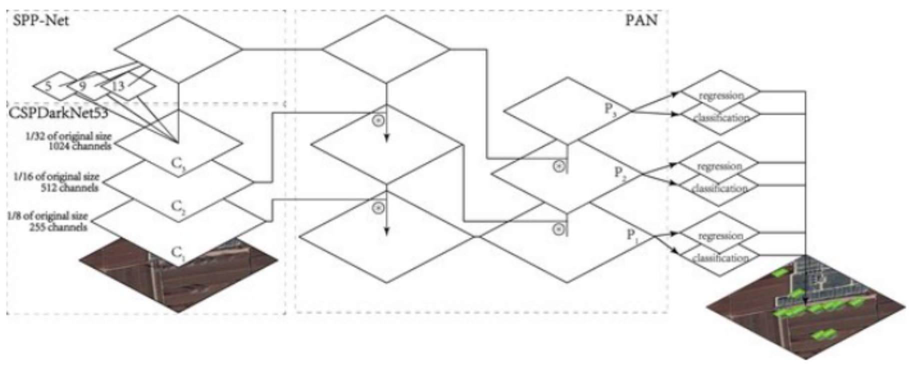

Figure 2: Illustration showing the strucutre of YOLOV4.[6]

# 2 Datasets

## 2.1 Cityscapes for Semantic Segmentation

### 2.1.1 context

This dataset comprises of numerous stereo video series described in street scenes from different cities covering high pixel annotations of frames and a bigger set of faintly annotated frames.the use of the dataset is to assess the performance of vision techniques for understanding urban semantic scenes such as instance level. pixel level and panoptic semantic labelling[3]

### 2.1.2 Content

This specific dataset contains 2975 training image files and 500 validation image files. Each image file is 256x512 pixels, and each file is a composite with the original photo on the half left of the image, alongside the labeled image (output of semantic segmentation) on the half right[4]. This dataset was saved in my google drive.

### 2.1.3 Platform and programming language

Python was used for this project and the platform used was google colab notebook.Google colab was used because it can process its run-time in GPU which is much faster for training data compared to CPU.

## 2.2 Own images for Object detection

I used four images that I captured from the streets of Passau more specifically captured around the Passau main Bus stop (ZOB) and a few images downloaded from google images for this task.

# 3 Use-cases

## 3.1 Semantic segmentation Implementation

### 3.1.1 Model Construction

Important python packages were first imported as the fist step such as os, pandas, numpy, matplotlib and sklearn.The run-time of the google colab notebook for faster results and implementation was set to run on GPU instead of CPU.The cityscapes dataset was downloaded in the google drive and set to be mounted in the google colab notebook for easier access and if stored in the local directory of the notebook, the files have to be downloaded again since all the downloaded

files get deleted after every session.The dataset root path was then set. After determining the dataset root path,data analysis is performed by obtaining a sample image with their corresponding output pair in the same image from the dataset to test if the image retrieval works well.The input image and the its corresponding label are in the same image.
The cityscapes image is then split into two different entities showing the cityscapes image and its corresponding label. The dataset is defined as the class and this functions as an iterator that yields a distinct input and its label image.The image is then normalized. Normalization lessens the effect that any channel (R,G,B) may have on the initiations on the subsequent outputs.
The Model is then defined,Unet model has been used in this implementation.

### UNet

### Background

The UNet structural design was established in the medical field for biomedical image segmentation. This design consists of two key components such as decoder and encoder. The encoder is associated with the covenant levels and pooling processes which are used to obtain elements in the image. The decoder authorizes localization with the use of transposed convolution.[5] In-order to further experiment on the model,it was implemented in scene segmentation of autonomous vehicles.

### UNet structural Design

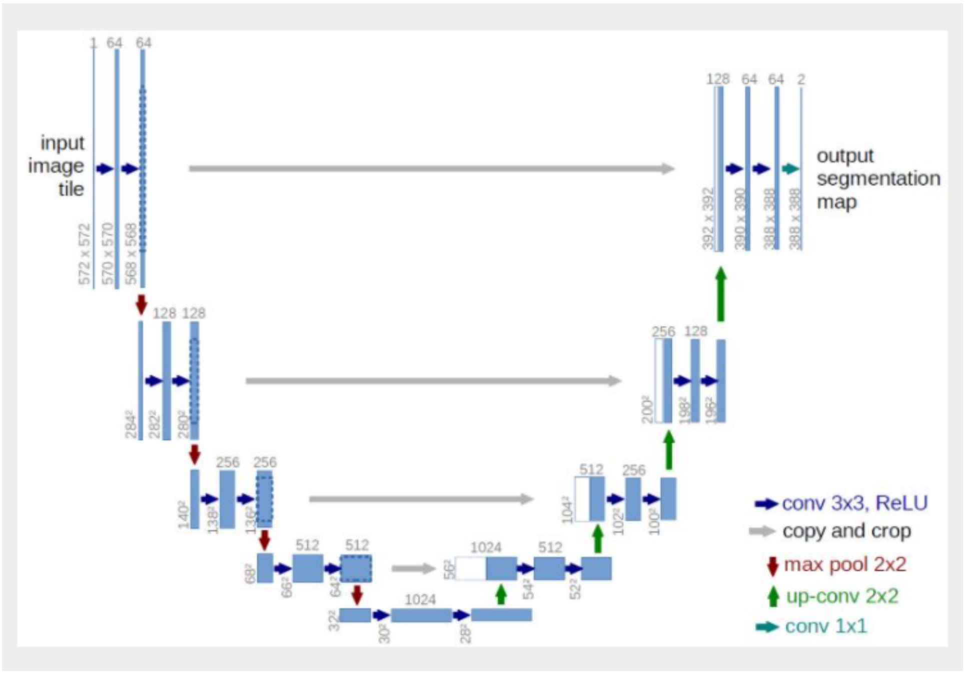

Figure 3: Illustration showing UNet structural design.[5]

### 3.1.2 Model Training

Training the model was done for 5 epochs and reduced the batch size from 16 to 10 to help in faster implementation since it takes too much time for each epoch to be completed and used a learning rate of 0.01. An epoch specifies the sum of runs of the full training dataset the algorithm has achieved.However, the higher the number of epochs are set, the better the results as the error rates are reduced.the model is saved and the training losses are plotted to monitor how close the predictions are to the true labels.

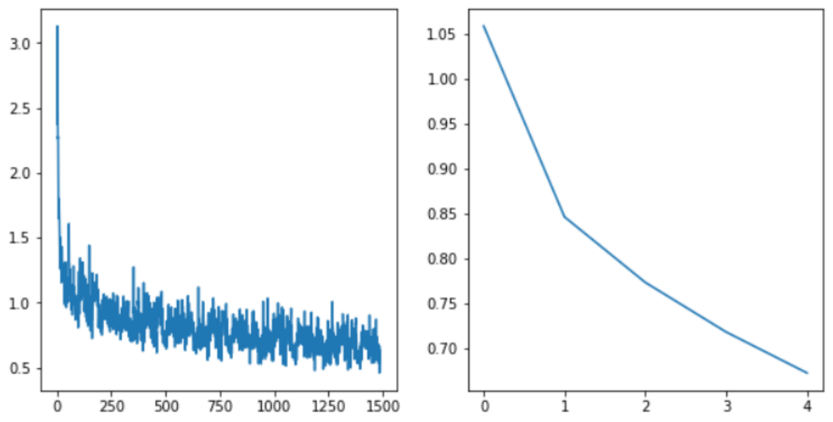

Figure 4: Illustration of the plot showing the step losses and epoch losses in axes 1 and axes 2 respectively on the training data

### 3.1.3 Model Prediction

In the predictions, the model is loaded and saved where the test set is fetched by iterating through the set and predicting the new unobserved images. All the keys should match successfully in both the model and the device to make the predictions.

### 3.1.4 Intersection Over Union (IOU) Score

Intersection over Union is an assessment metric that calculates the accuracy of object detection on a specific dataset.This is used to evaluate the accuracy of the detection after it has been trained.For the application of the Intersection over Union in the dataset, the ground-truth and predicted bounding boxes from the model is required.[7] In the figure 3 illustration, the bounding boxes which highly overlap has a higher IOU score which means that the predictions are closer to the ground-truth

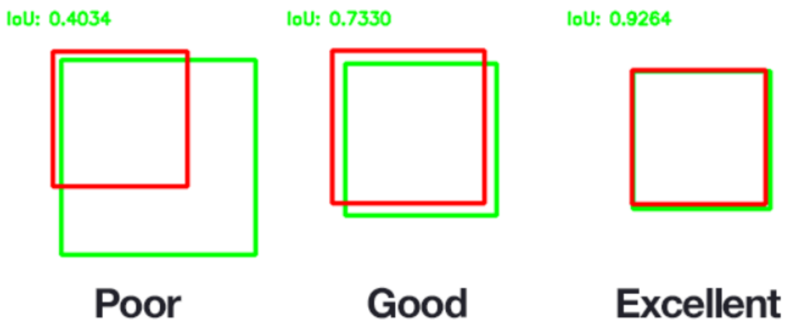

Figure 5: Illustration showing the IOU of various bounding boxes.The predicted bounding box is in red and the ground truth bounding box is in green,[7]

The objective is to use the training data and validation data with the bounding boxes to develop an object detection algorithm and performing its evaluation on the testing data.The IOU score of the implementation was 0.8198. predictions.[7]

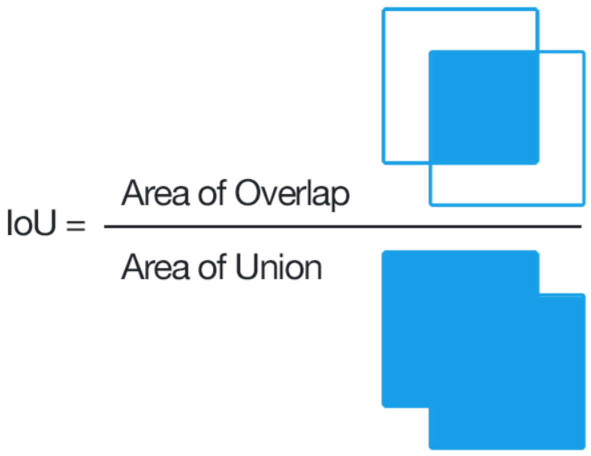

Figure 6: This shows the formula for calculating the IOU[7]

## 3.2 Object Detection Implementation

I cloned the repository that contains all the darknet files.Darknet was installed.Darknet is an open source neural network framework that has been written in C and Cuda(parallel computing platform that speeds up applications with the help of the GPU).Opencv was also used as a dependency for Darknet since it comprises of a larger array of backed images.The Darknet was then built so that the darknet executable file to run or train object detectors could be used.The Google colab was then connected to the google drive to access the darknet since it was stored in the drive.The hardware accelerator was changed to GPU to perform faster (Cuda).Yolov4 is a state-of-the-art real-time object detection system where validation of the information with darknet is done as it loads the configuration file and the weights after which it later classifies the images and then prints the top 10 classes for the image.[1]

Darknet contains a helper function that uploads an image directly from the computer and performs Object detection on that image.The images stored in the google drive can also be assessed by mounting the drive on the notebook by signing into the google drive account.After verification,the images are accessed using the root path.The output of the images are stored and showed in the prediction.jpg with all the classes labelled.

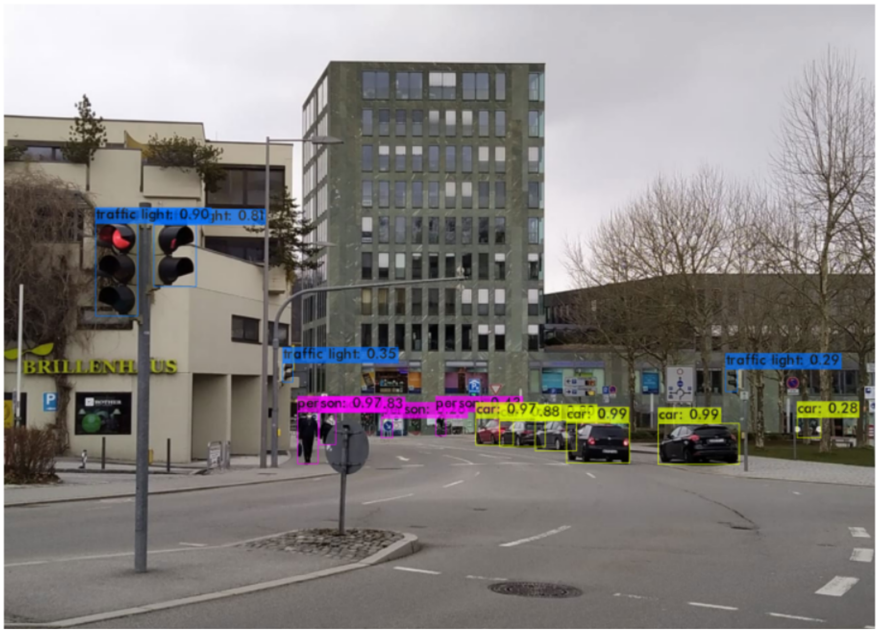

Figure 7: Illustration showing the predicted image after the implementation of Darknet in Yolov4

## 4 Challenges faced during implementation of object detection and semantic segmentation

Since object detection in autonomous vehicles are still in the development phase,more implementations have to be carried out.I used Google colab notebook because it has GPU which makes the implementations faster.This means that using other notebooks such as jupyter notebook will make the implementation very slow in case no GPU is present in some computers and it will have to run on CPU.
Training the images for semantic segmentation was also very slow even in Google colab with a higher batch size and epochs as the GPU run-time memory was running low.I reduced the Batch size and the epochs to suit the memory and also to implement faster but the results were not good enough.

## 5 Conclusion

The implementation carried out with the UNet model in cityscapes dataset gives good results. An IOU score of approximately 82 percent shows relatively good result.This implies that the similarity between the actual label and the predicted output is not very big although better implementation can be carried out to get better results.Implementation of object detection in yolov4 using Darknet labels the top 10 classes which implies that some important classes may be left out such label on buildings or trees which may not be good as it may collide with them and cause accidents.More work should be done on the identifying and defining more classes in the model.

## References

[1] Reviews, features, pricing, comparison, 2017.

[2] Overview of the yolo object detection algorithm, 2018.

[3] 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.

[4] DanB. Cityscapes image pairs, Apr 2018.

[5] Rohit Dwivedi. My experiment with unet - building an image segmentation model, Mar 2021.

[6] Sijia Qiao, Yu Sun, and Haopeng Zhang. Deep learning based electric pylon detection in remote sensing images. *Remote Sensing*, 12:1857, 06 2020.

[7] Rosebrock. Intersection over union (iou) for object detection - pyimagesearch, 2021.