Semantic Segmentation

Assignment 3 Report

Syed Kumayl Raza Moosavi

Reg. No.: 364013

MS AI 2K21

May 23, 2022

CS 893 - Advanced Computer Vision

Instructor - Dr. Muhammad Moazam Faraz

NETWORK DETAILS

Two networks have been employed for this assignment as shown in Figure 1. Network A uses U-net baseline architecture and Resnet34 is used as a backbone for feature extraction. Network B is basically a rendition of the DeepLab V3 network in which xception is used as the backbone, atrous convolution is used in the last few blocks of the backbone. On top of extracted features from the backbone, an ASPP network is added to classify each pixel corresponding to their classes and the output from the ASPP network is passed through a 1 x 1 convolution to get the actual size of the image which will be the final segmented mask for the image. For training, callback API is used in which LR is reduced on Plateau and model checkpoints are saved according to the validation accuracy. For performing a fair comparison between the two networks, the same parameter settings are used i.e. softmax activation, Adam Optimizer, Categorical Cross Entropy Loss and a learning rate of 0.001.

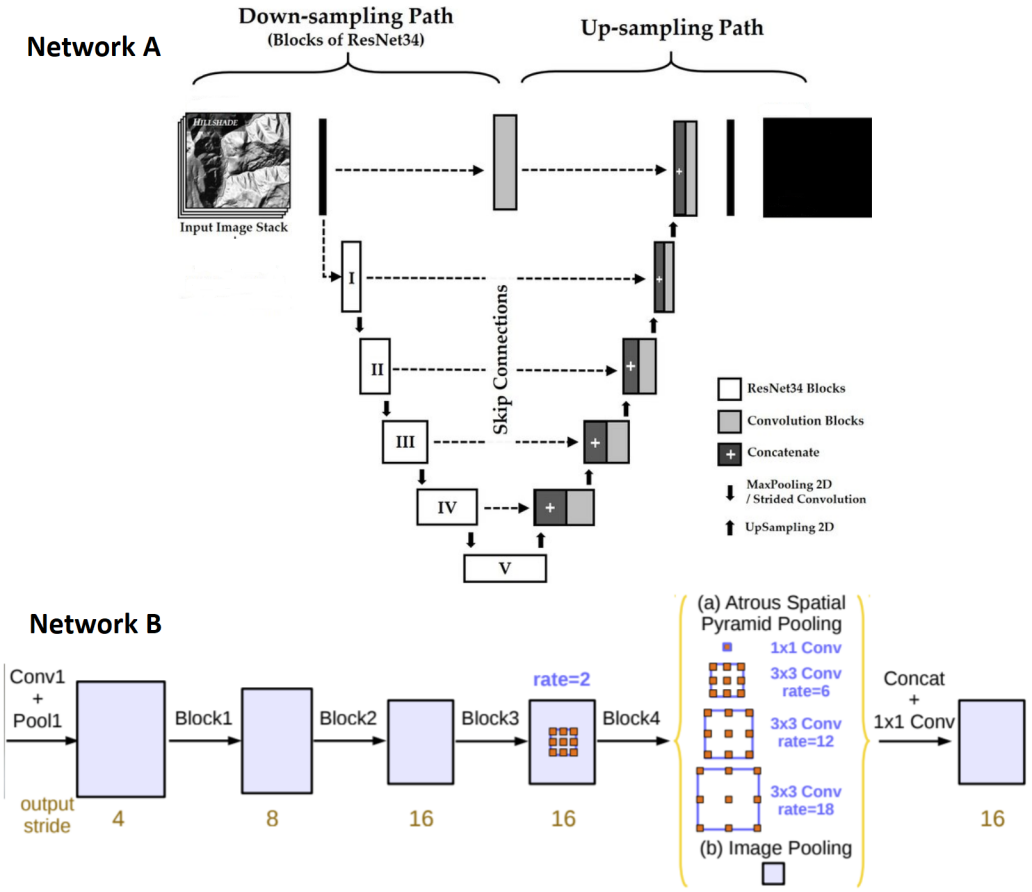

Figure 1: Network Architectures

DATASET

The dataset downloaded from the Github link contains 367 training images. In conjunction with these images, relevant segmentation files are present for 11 classes. So classification models will be trained on 12 classes.

To increase the number of files for better model training, each image is divided into patches of 9 images and during training Data Augmentation is performed from the ImageDataGenerator class such as rotation, width/height shift, horizontal flip, etc.

PERFORMANCE MEASURES

The following performance metrics were used for the classification task:

1) Accuracy: Measure of correctly classified images over all classified images
2) F1 Score : To cater for the imbalance of the dataset classification test the F1 Score is used using the precision and recall of the dataset
3) Dice Coefficient: It is a statistical tool which measures the similarity between two sets of data
4) Specificity: The ability of a test to correctly identify people without the desired class
5) Sensitivity: Refers to the probability of a positive test, conditioned on truly being positive

QUANTITATIVE ANALYSIS

This section shows the quantitative results for both networks trained at 20 epochs and a batch size of 16 images per epoch. After image division into patches and resizing, the input training size is (5872, 128, 128, 3) and segmentation masks size is (5872, 128, 128, 12). Train_validation split of the input dataset is 80% by 20%. For testing purposes the test dataset provided in the dataset is preprocessed and evaluated on testing size (1616, 128, 128, 3) and testing segmentation dataset of size (1616, 128, 128, 12). Table I shows the model metrics of training and testing results.

Table I: Training, Validation and Testing Metrics

Train Metric	**Network A**	**Network B**
Loss	0.5270	0.3654
Accuracy	0.8452	0.8939
F1 Score	0.8531	0.8979
Dice Coefficient	0.7798	0.8413
Specificity	0.9929	0.9948
Sensitivity	0.8025	0.8619

Validation Metric	**Network A**	**Network B**
Loss	0.6748	0.6833
Accuracy	0.8088	0.8213
F1 Score	0.8183	0.8279
Dice Coefficient	0.7437	0.7735
Specificity	0.9902	0.9895
Sensitivity	0.7688	0.7893

Test Metric	**Network A**	**Network B**
Loss	0.6042	0.5598
Accuracy	0.8331	0.8342
F1 Score	0.8376	0.8403
Dice Coefficient	0.7455	0.7754
Specificity	0.9920	0.9911
Sensitivity	0.7843	0.7957

Time is also an evaluation metric for model comparison and evaluation. Table II shows the train and evaluation time of the model to achieve the requisite metrics as shown in Table I

Table II: Time Analysis

Time (sec)	**Network A**	**Network B**
Training Time	13282.2772	27201.8993
Evaluation Time	42.9580	114.1300
Prediction Time	37.0324	97.6910

Model training graph trend is also shown in Figure 2 for both architectures over the epochs for loss, dice coefficient and accuracy. The processor used for computation was an Intel Core i7-9th Gen CPU @ 2.60GHz

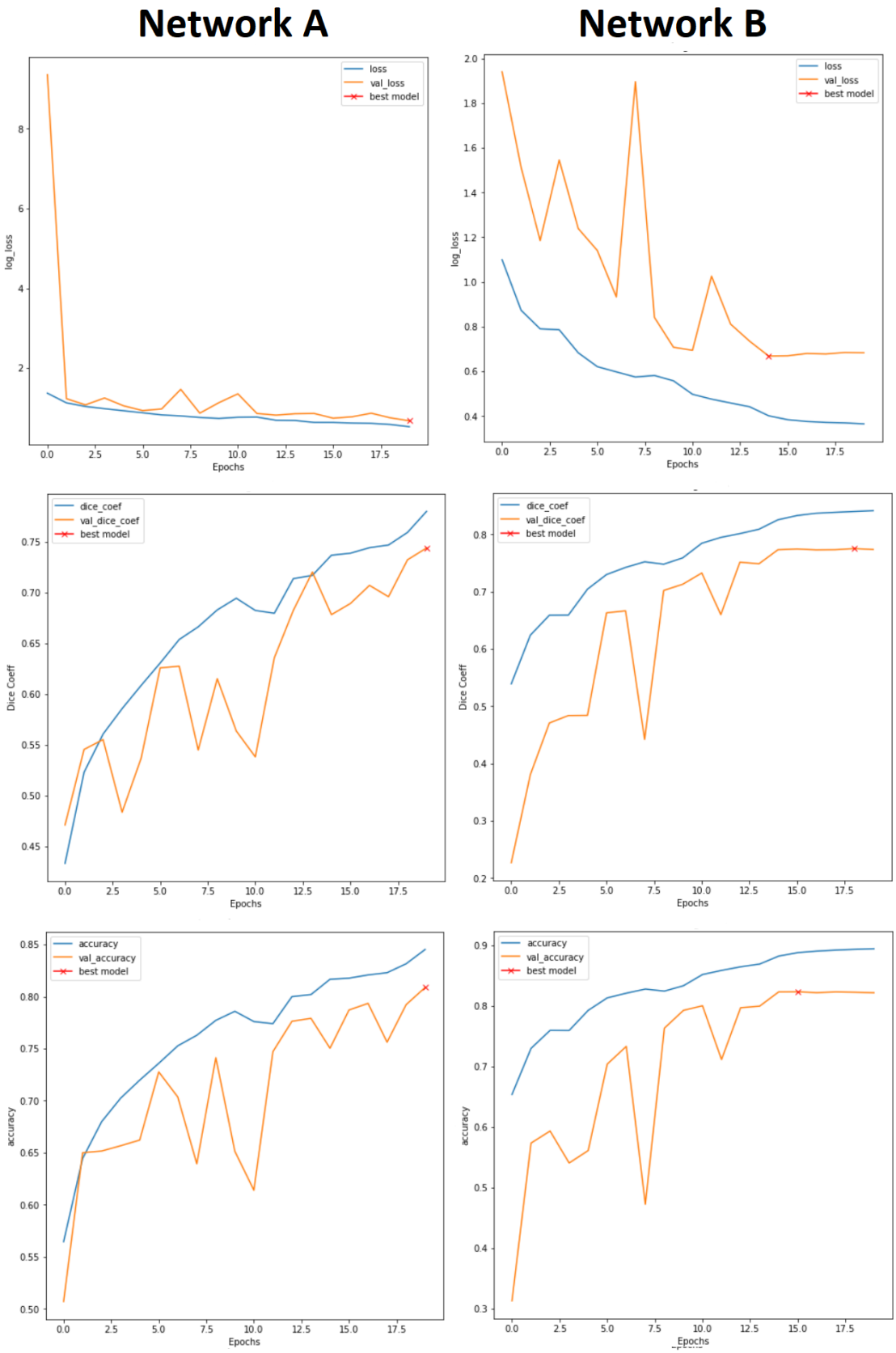

Figure 2: Network learning curve over number of epochs

QUALITATIVE ANALYSIS

Figure 3: Original Image Samples

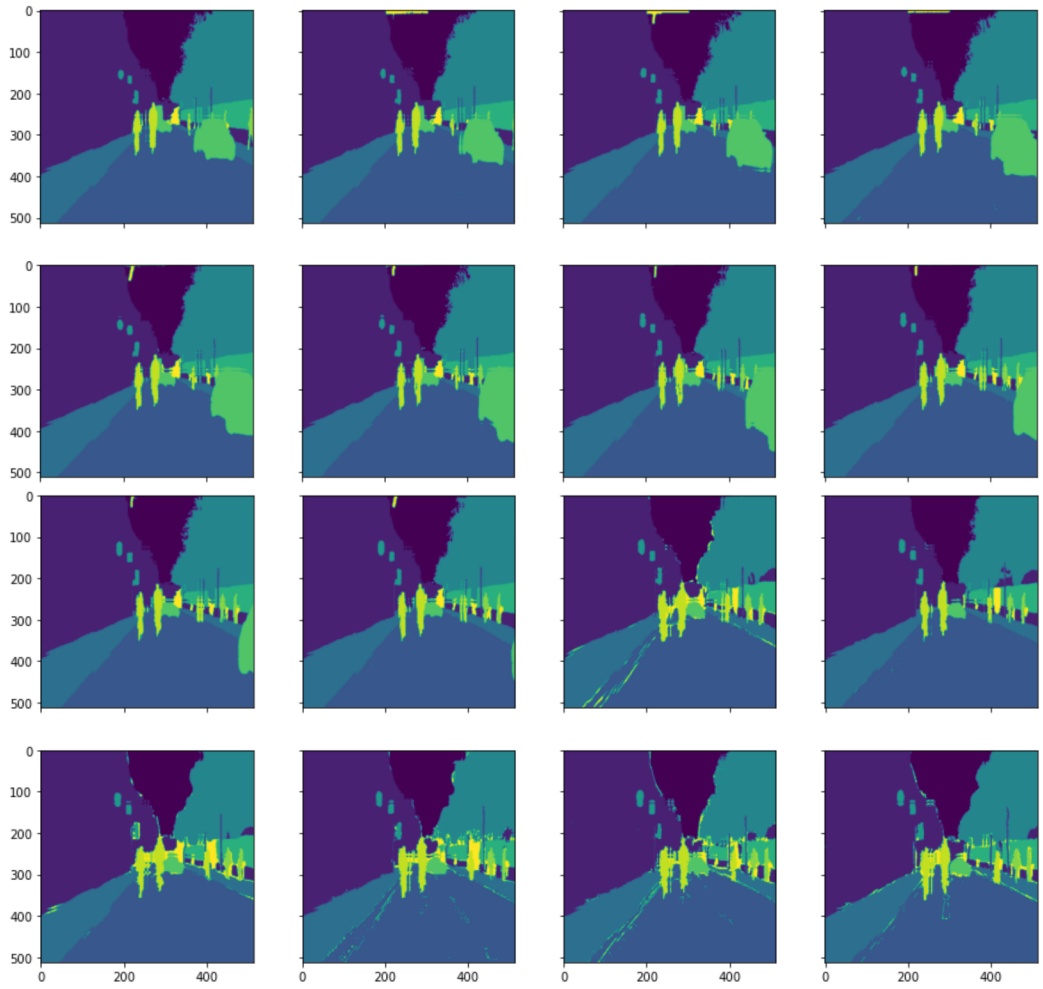

Figure 4: Ground Truth Image Samples

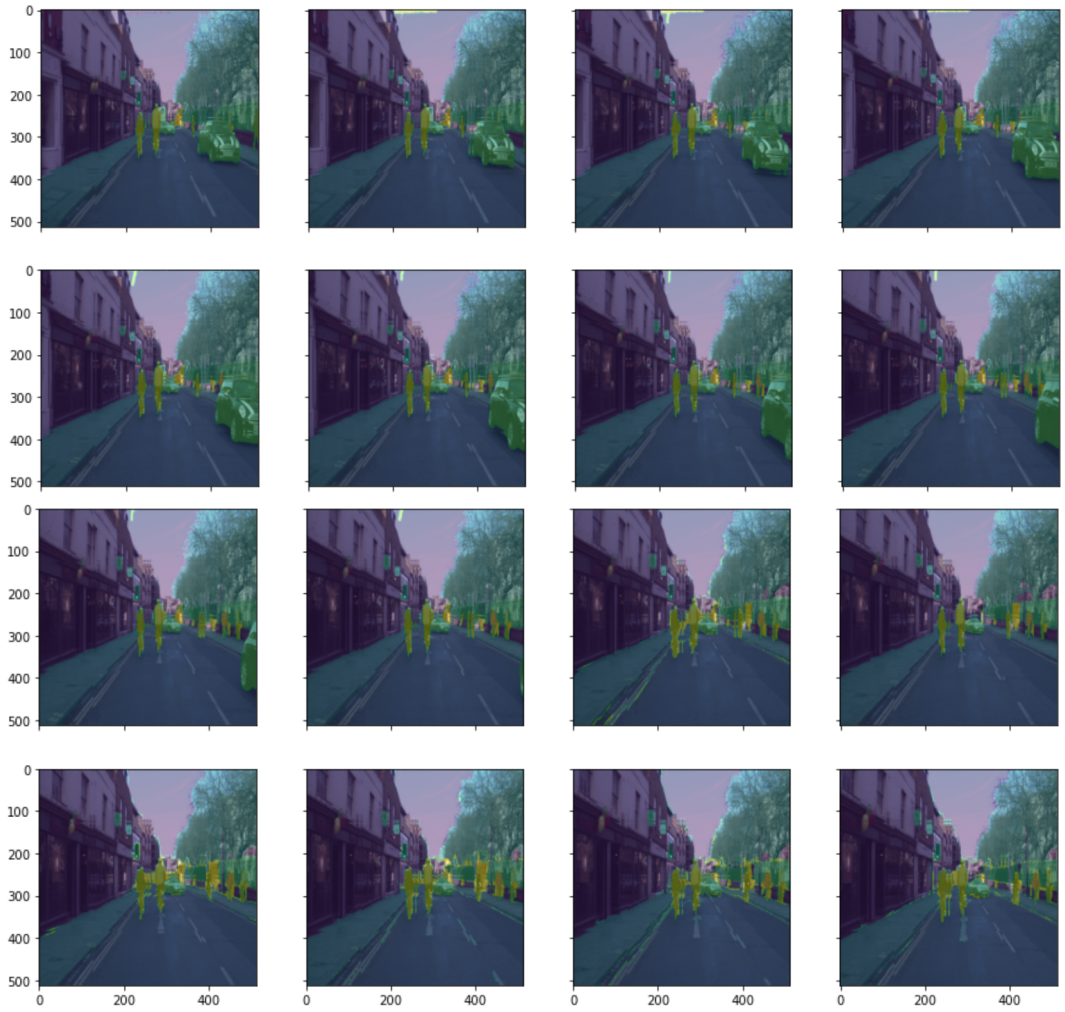

Figure 5: Original Image with Ground Truth Superimposed

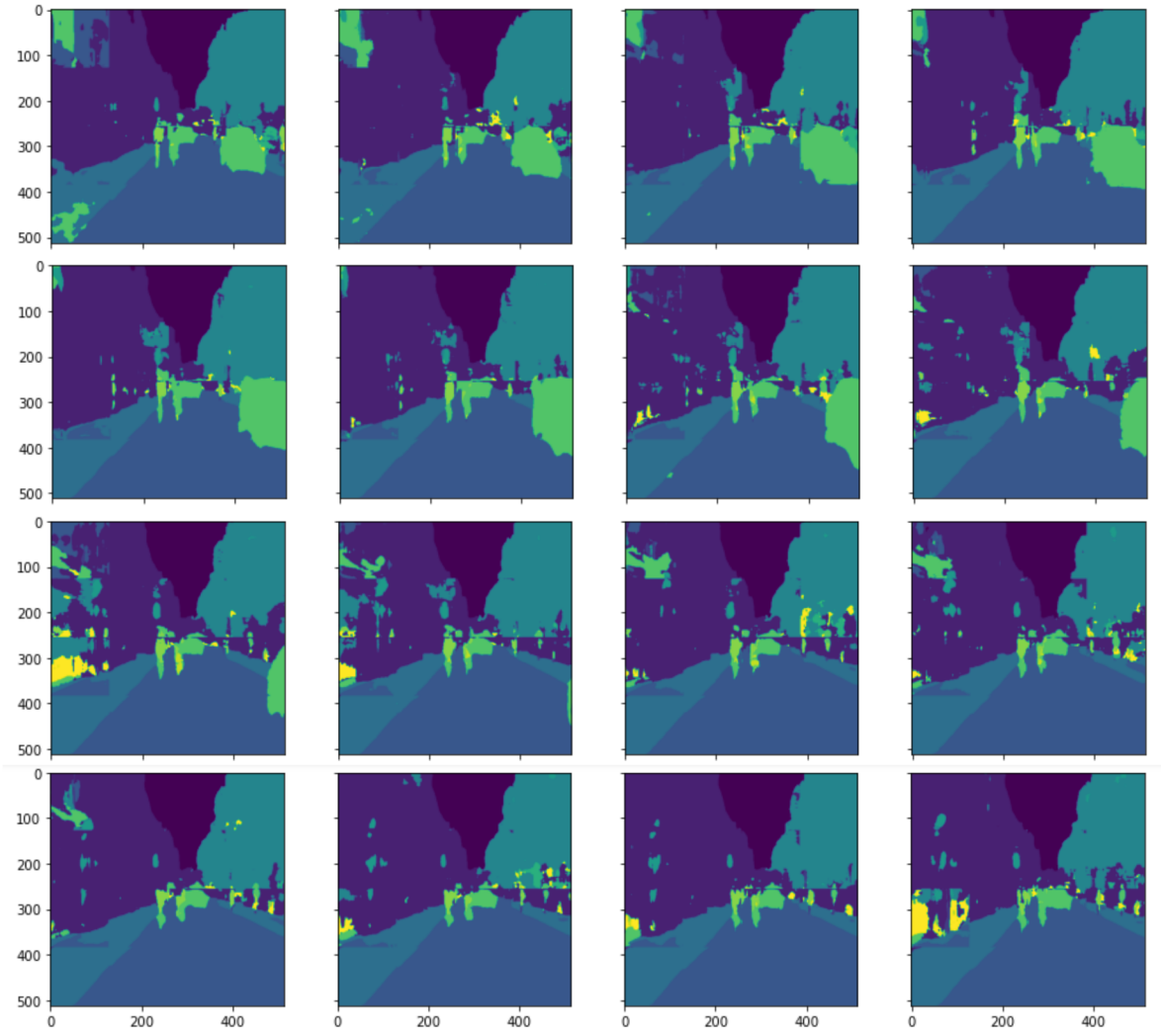

Figure 6: Predicted Segmentation Mask Network A

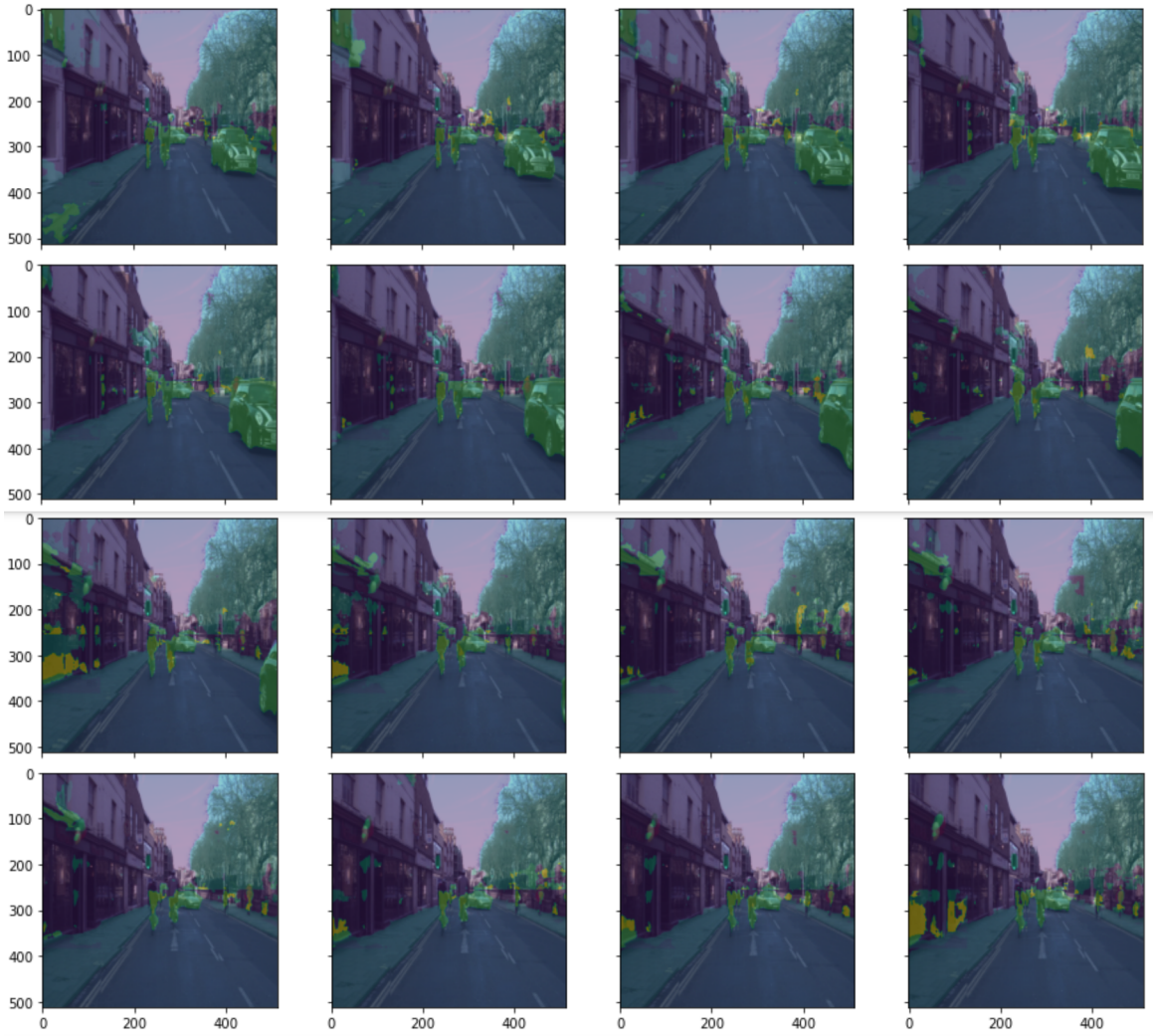

Figure 7: Original Image with Predicted Segmentation Mask Superimposed Network A

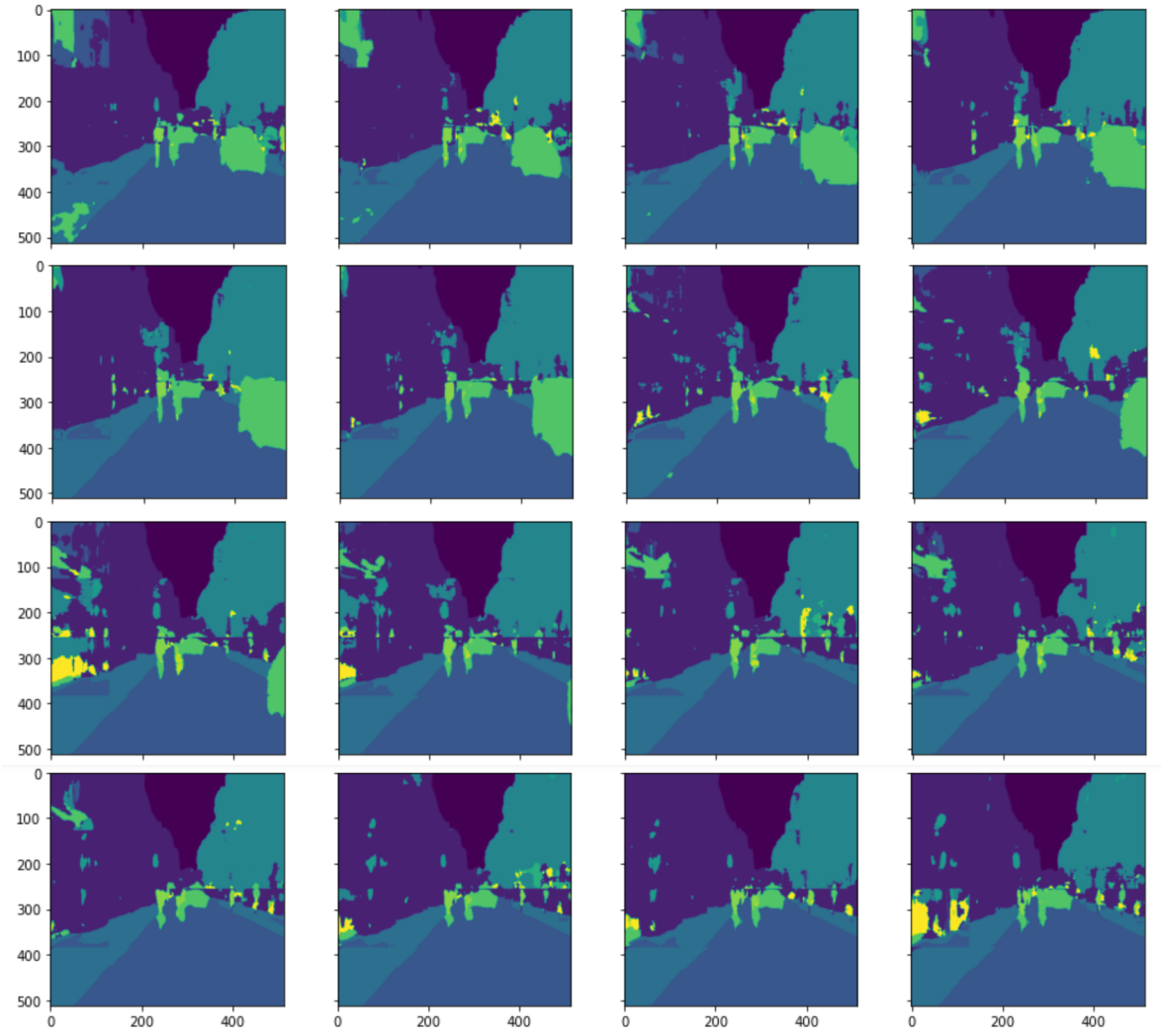

Figure 8: Predicted Segmentation Mask Network B

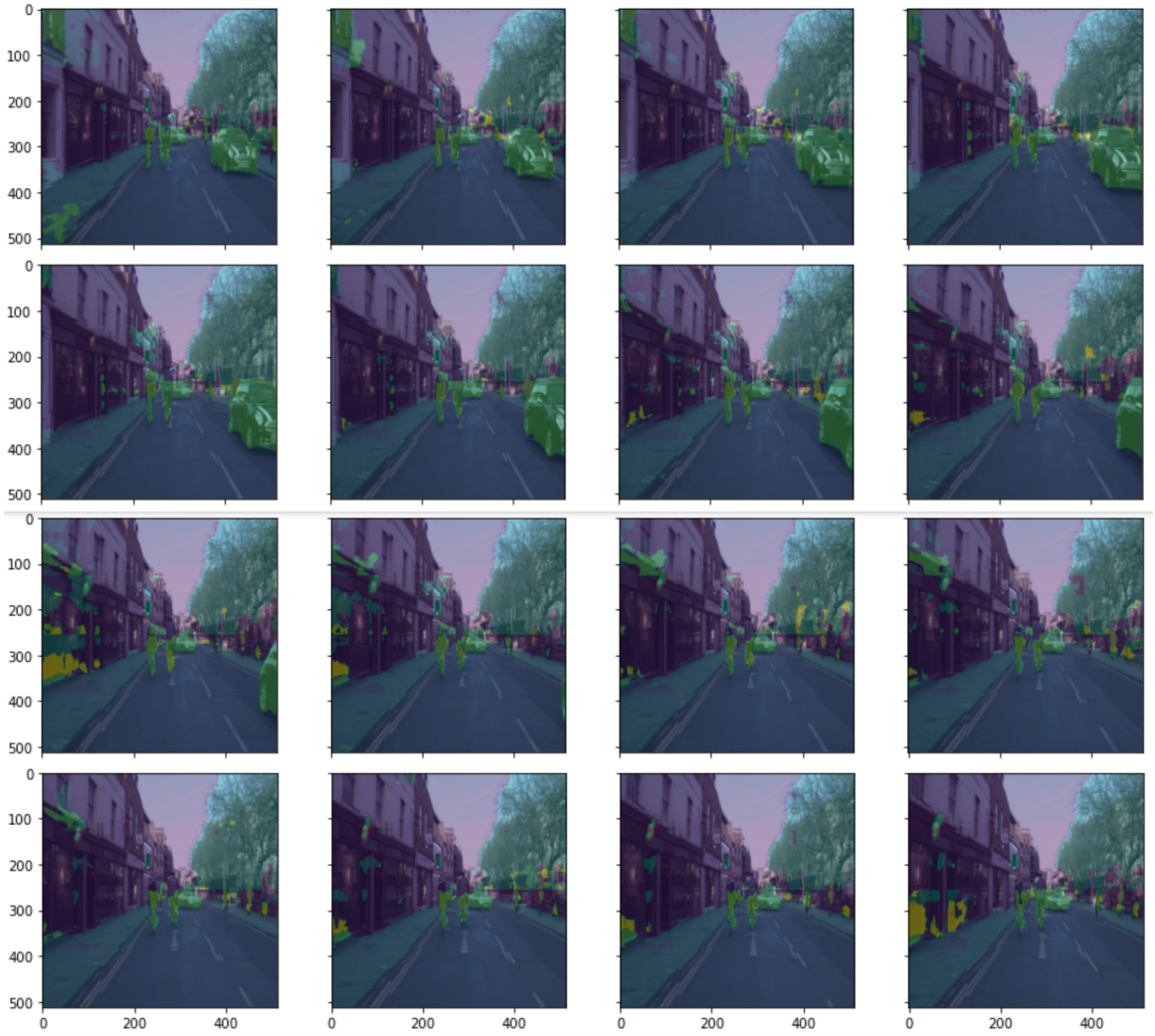

Figure 9: Original Image with Predicted Segmentation Mask Superimposed Network B

HOW TO RUN THE CODE

Download the zip file containing the python notebook from the Github repository from the link provided as under:

`Github`: Python Notebook File

Download dataset zip file from the following link:

`Google Drive` : A3_Dataset

The Github link contains the python notebook. Unzip dataset folder. Folder 'A3_Dataset' and python notebook file should be in the same folder.

Run the jupyter notebook file