

CityScapes

Semantic Segmentation

M.I.A. ML Project 2024/25

Introduction

- *Who is this team?*
- *What is the Aim of the Project?*

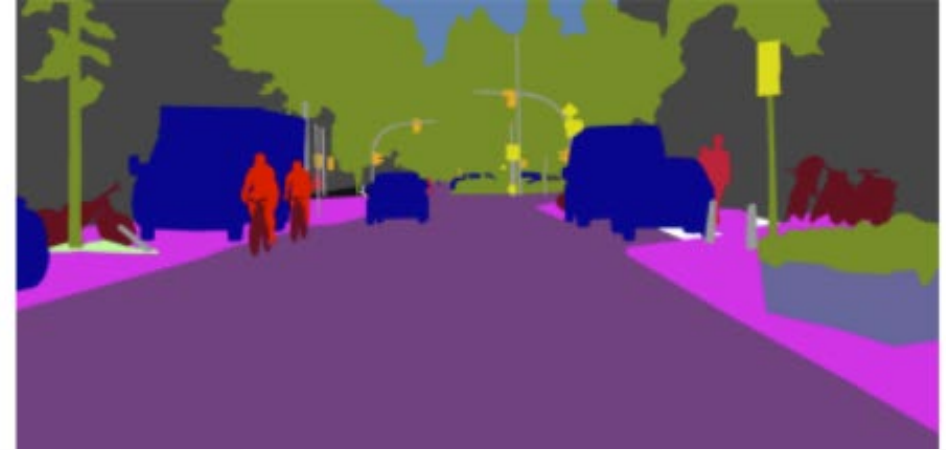
Group 4



- *Names*
- *Roles*
- *Day 1 to*



The Project

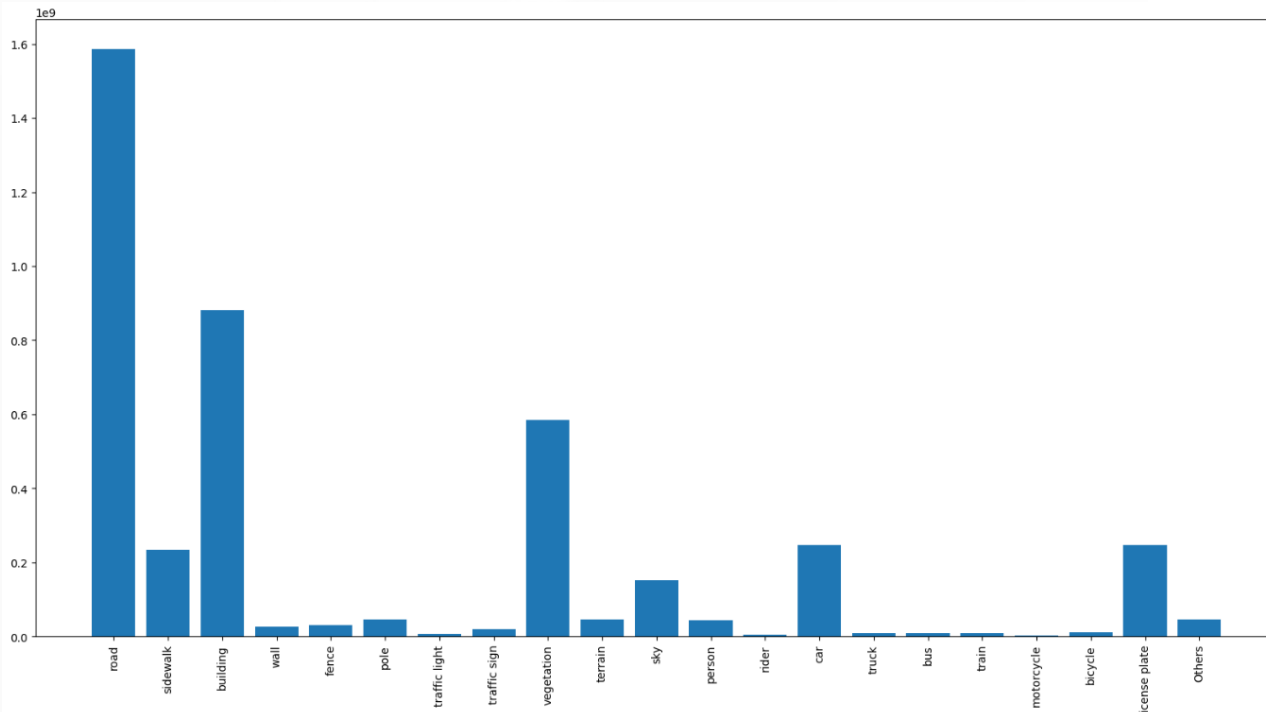
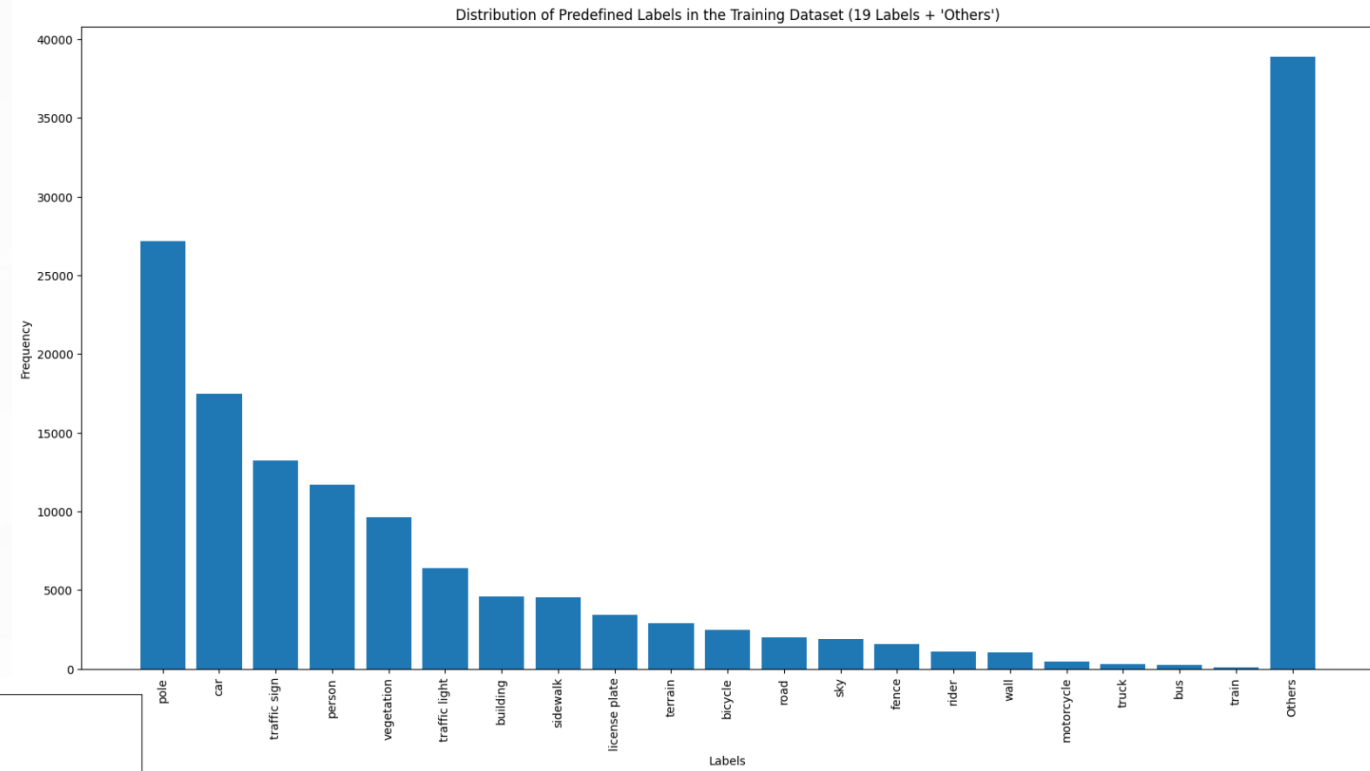


- *CityScapes Dataset*
 - *Released in 2016.*
 - *Contains 50,000 images.*
 - *Captured in 50 different cities in Germany.*
- *Semantic Segmentation*
 - *Types Of Segmentation.*
 - *How to approach?*

Data Pipeline

- *Data Analysis*
- *Data Augmentations & Preprocessing*
- *Data Challenges*

Data Analysis



Data Augmentation & Preprocessing

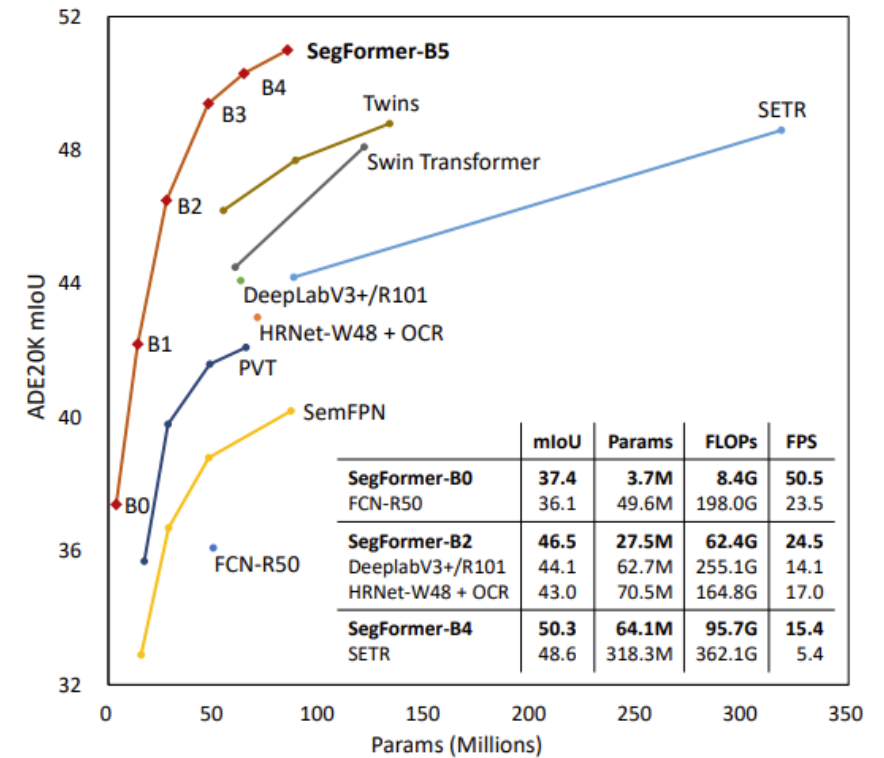
- *Resize*
 - *2048x1024 is too big for our training.*
 - *Used both 1024x512 and 512x256 for several training loops.*
- *Rotation*
 - *Random from -12° to 12°.*
- *Gaussian Blur*
 - *Removing the noise from the images.*
- *Horizontal Flips*
 - *Random Horizontal Flips.*
- *Mask Encoding*
 - *Unlabeled in mask from 255 -> 20.*
 - *One Hot Encoding the mask to suit our model.*

The Model

- *Model Selection*
- *U-net*
- *SiLU vs ReLU*

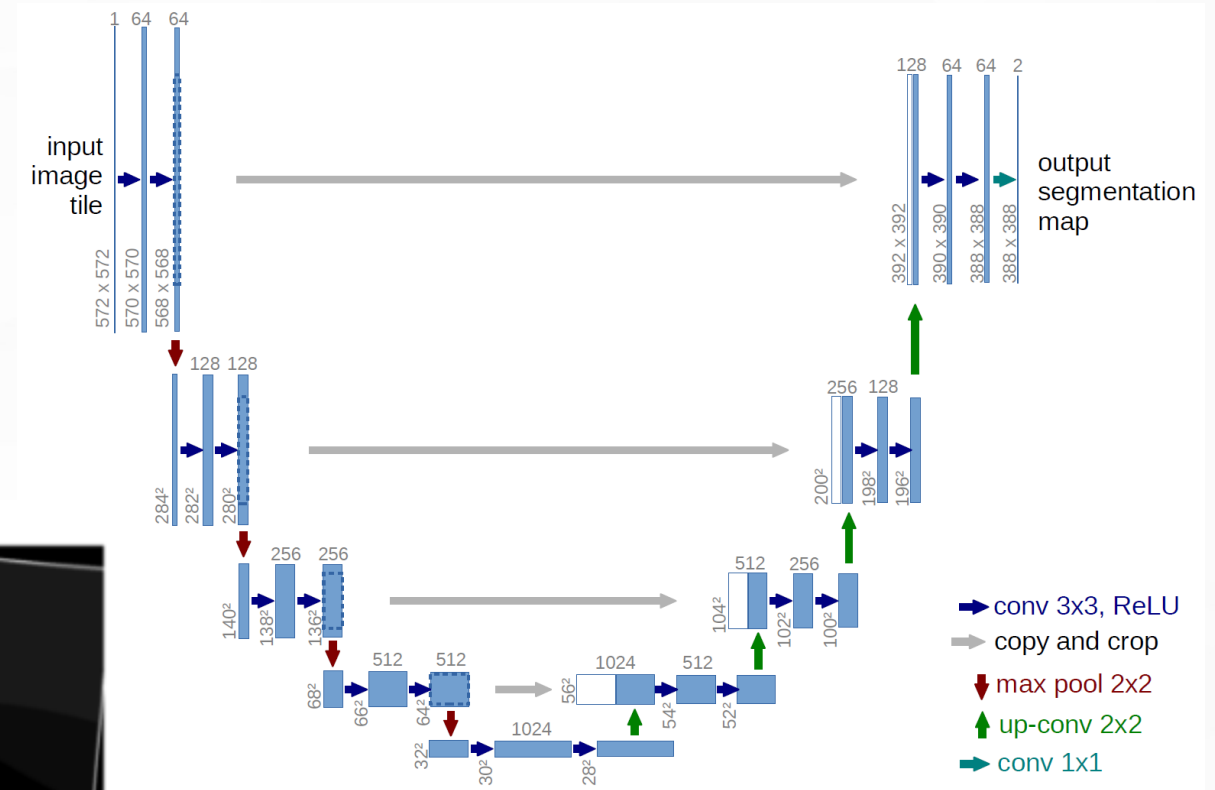
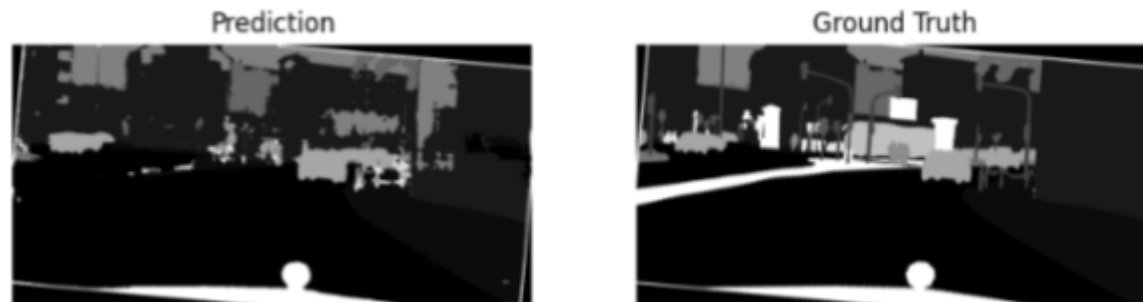
Models Considered

- **FCN**
Relatively weak performance.
- **DeepLabv3+**
Gigantic number of parameters.
- **Segformer**
Needed Nvidia API key.
- **U-Net**
Relatively slow to train.



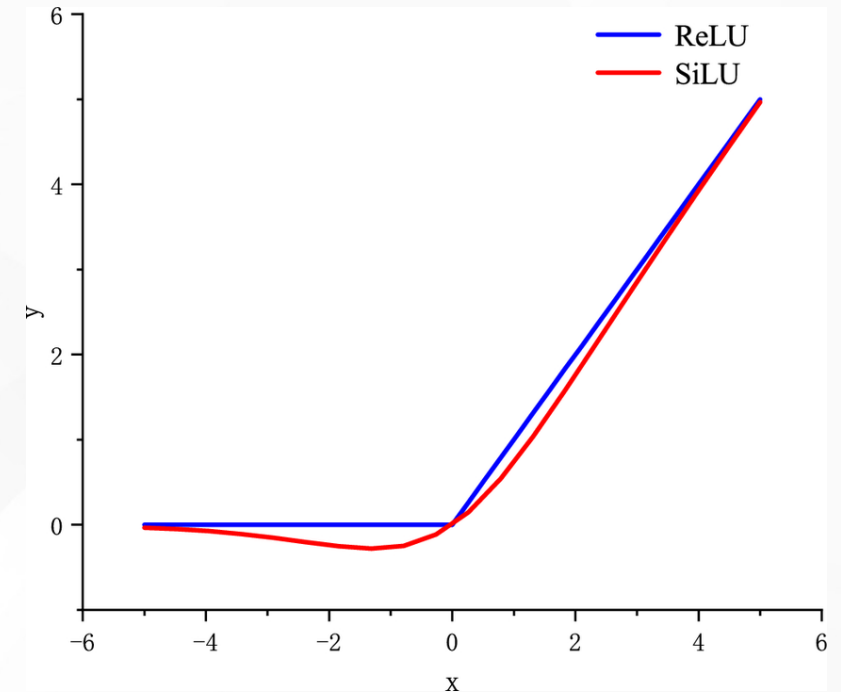
U-Net

- *Performance from 1st epoch!*
- *CNN – enc/dec Architecture.*
- *Input and Output.*
- *Coding Process.*



SiLU vs ReLU

- *ReLU*
 - $F(x) = \max(0, x)$
 - *The one U-Net originally used.*
- *SiLU*
 - $F(x) = x * \text{sigmoid}(x)$
 - *Introduced by Google in 2017.*
 - *Found to be the best practice in our task.*
- *Comparison*
 - *ReLU can cause a vanishing gradient problem.*
 - *SiLU accommodates negative values.*
 - *SiLU is differentiable at Zero.*



Training

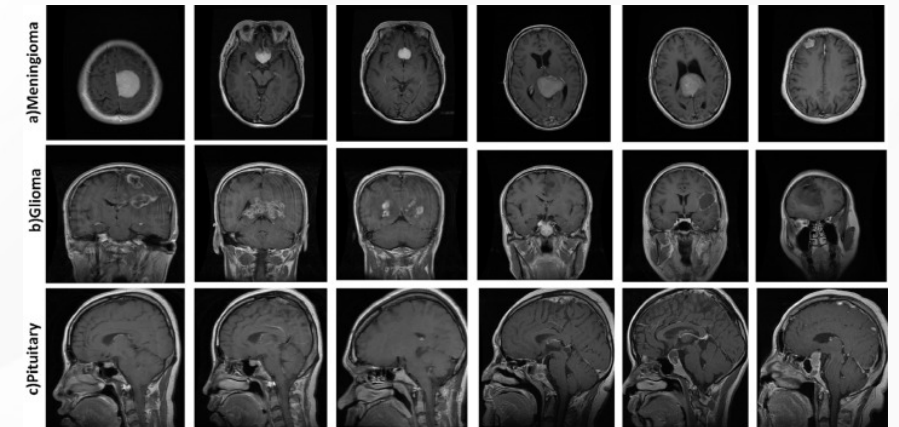
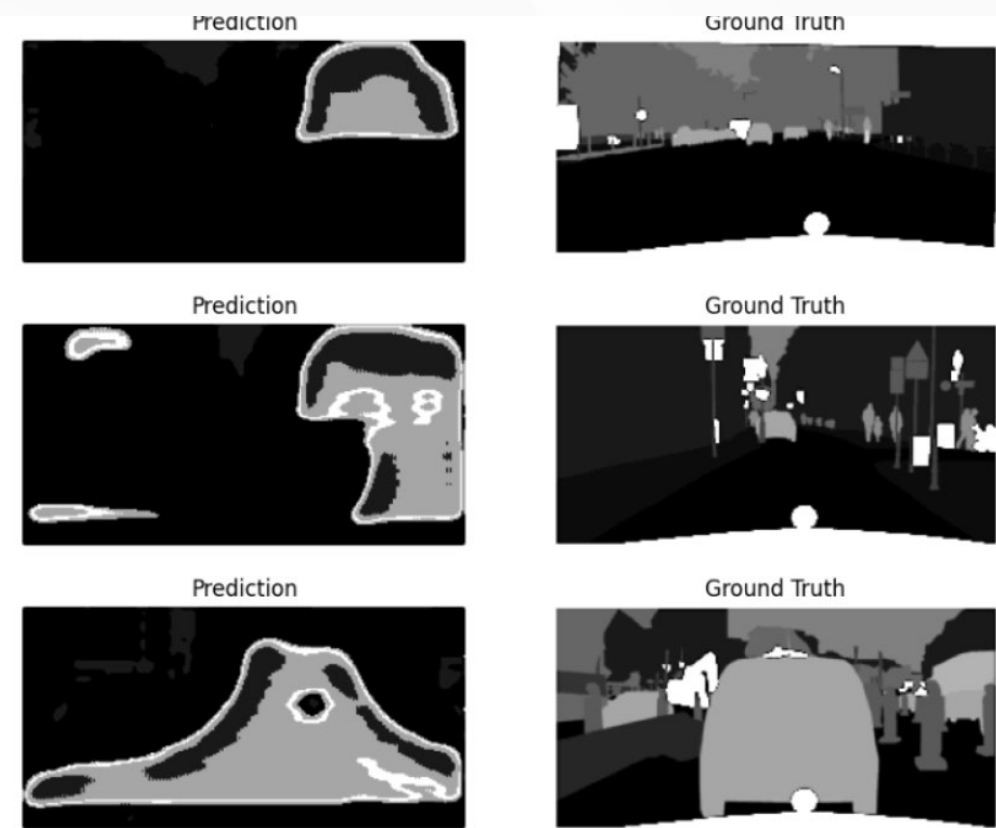
- *Training Setup*
- *Loss function journey*
- *How long? { 4 Days }*

Training Setup

- *Early Stopping.*
 - *Using early Stopping to control overfitting, while Saving the best model.*
- *Optimizer.*
 - *Usage of AdamW to achieve faster convergence.*
- *Training loop.*
 - *100 epochs, patience = 5, batch-size = 4.*
 - *Plotting Image, Prediction, and Ground Truth each 5 epochs.*
 - *Monitoring train_loss, val_loss, dice_score, and IoU_score for each epoch.*
- *Loss function.*
 - *Next Slide, But we settled on nn.CrossEntropyLoss.*

Loss function Journey

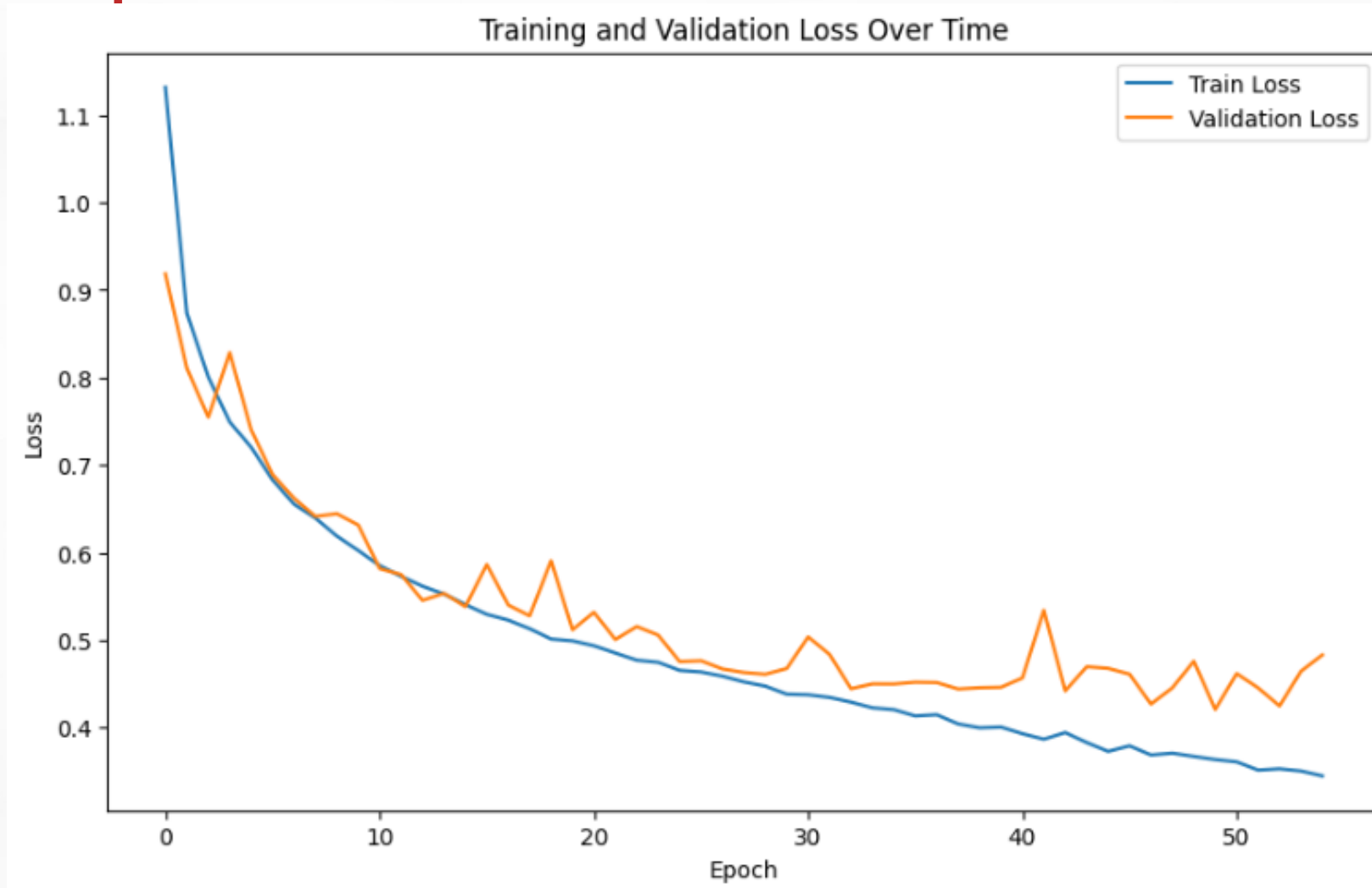
- *Dice Loss*
 - *Failure.*
- *Focal Loss*
 - *Failure.*
- *Combined Loss*
 - *Another Failure.*
- *Cross Entropy Loss*
 - *What we Began with and ended up with.*



Results & Conclusion

- *Final Output performance*
- *Challenges and difficulties*
- *Conclusion*

Final Output

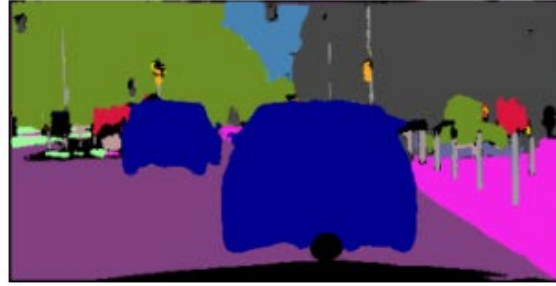


Final Output

Image



Prediction Mask (Colored)



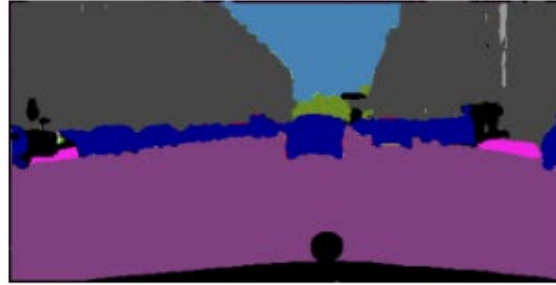
Overlay



Image



Prediction Mask (Colored)



Overlay



Image



Prediction Mask (Colored)



Overlay



Conclusion

- *Learning needs trying, Like our model.*
- *Our First Segmentation Project!*
- *It's finally over, right?*



Q&A



♥ Thank you ♥