Friedrich-Alexander-Universität
Technische Fakultät

FAU

Road Scene Understanding For Visually Impaired

Dhruv Sharma - aw51ylaq
(Group 6)

Introduction

Task

To develop a Sidewalk Environment Detection System for enhancing the mobility capabilities of visually impaired people through the combination of GPS systems and image segmentation techniques refined for sidewalk recognition.

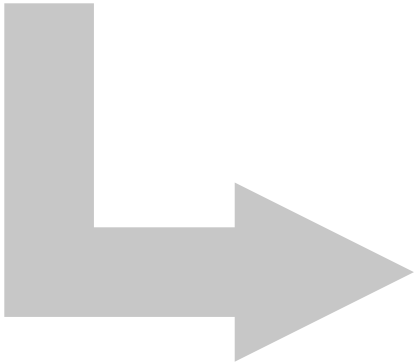
Method

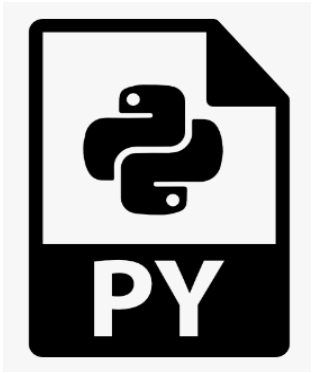
- Using Valhalla routing API and GPS tracker signals to determine direction instructions.
- Training DeepLabv3 ResNet50 image segmentation model on Cityscapes dataset (5000 images, 50 cities) in Pytorch.
- Fine-tuning the model on Mapillary dataset (1000 images) to improve sidewalk detection.
- Combining GPS system and image segmentation model to assist navigation.

Part 1 – GPS System

Processing GPS signals

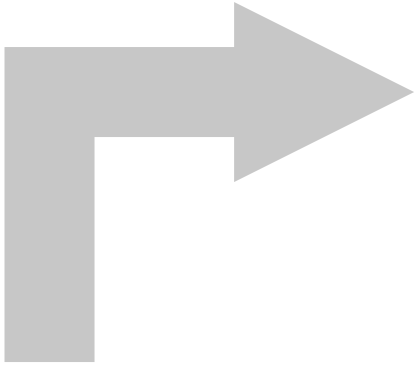
File with GPS coordinates to simulate a walk (Timestep, Lat, Long)





Calculate time, position and speed of the walk (using Haversine distance)

Output for the route



```
{'elapsed_time': 3.0,
 'lat': 49.59628978,
 'lon': 11.00211628,
 'speed': 1.5846891848498625},
{'elapsed_time': 5.0,
 'lat': 49.59627368,
 'lon': 11.00212265,
 'speed': 0.9240861571274116},
{'elapsed_time': 7.0,
 'lat': 49.59627498,
 'lon': 11.00213958,
 'speed': 0.6143663616426172},
{'elapsed_time': 9.0,
 'lat': 49.59627272,
 'lon': 11.00218249,
 'speed': 1.551428202417884},
{'elapsed_time': 11.0,
 'lat': 49.5962736,
 'lon': 11.002212,
 'speed': 1.0645657103600967},
{'elapsed_time': 13.0,
 'lat': 49.59627653,
 'lon': 11.00224469,
 'speed': 1.1892470136246764},
```

Using Valhalla API

https://valhalla.github.io/valhalla/

Processed GPS coordinates

Valhalla Map Matching API

Instructions from Valhalla for the entire walk route

```
Route info from valhalla

[{'duration': 24.0,
  'instruction': 'Walk east on Bahnhofplatz.',
  'location': [49.596276, 11.002119]},
 {'duration': 34.882,
  'instruction': 'Turn right to stay on Bahnhofplatz.',
  'location': [49.596342, 11.002576]},
 {'duration': 52.941,
  'instruction': 'Turn left onto Calvinstraße.',
  'location': [49.595933, 11.002722]},
 {'duration': 31.765,
  'instruction': 'Continue on Hugenottenplatz.',
  'location': [49.596085, 11.00374]},
 {'duration': 32.471,
  'instruction': 'Turn left onto Hauptstraße.',
  'location': [49.596176, 11.004351]},
 {'duration': 28.941,
  'instruction': 'Turn right onto Hugenottenplatz.',
  'location': [49.596582, 11.004208]},
 {'duration': 123.4,
  'instruction': 'Continue on Universitätsstraße.',
  'location': [49.596666, 11.004765]},
 {'duration': 31.765,
  'instruction': 'Turn right onto Schuhstraße.',
  'location': [49.596999, 11.006975]},
 {'duration': 0.0,
  'instruction': 'You have arrived at your destination.',
  'location': [49.596606, 11.007124]}]
```

**Valhalla map matching API requires a list of coordinates i.e. a trace route. Use the Turn-by-Turn route API if only start and end locations are known.

GPS System Output

Using GPS signals and Valhalla Instructions

Sample walk with navigation data

- Speed – Calculated using time elapsed and Haversine distance.
- Instruction – Valhalla instruction for current GPS position.
- Direction command – Go Left, Go Right or Stay Center based on Valhalla instruction.

Part 2 – Image Segmentation

Cityscapes Dataset

https://www.cityscapes-dataset.com/

*all images have been rescaled to 300x600

Image

Ground Truth

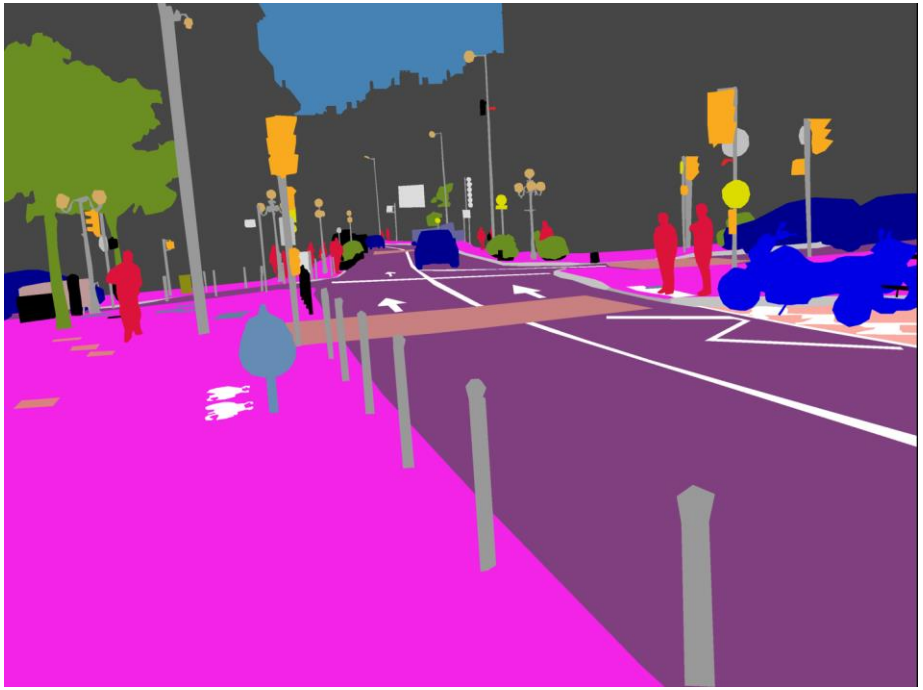
Ground Truth Grayscale

Mapillary Dataset

Provided by the Professor for fine-tuning the model.

Image

Ground Truth



Mapillary Dataset

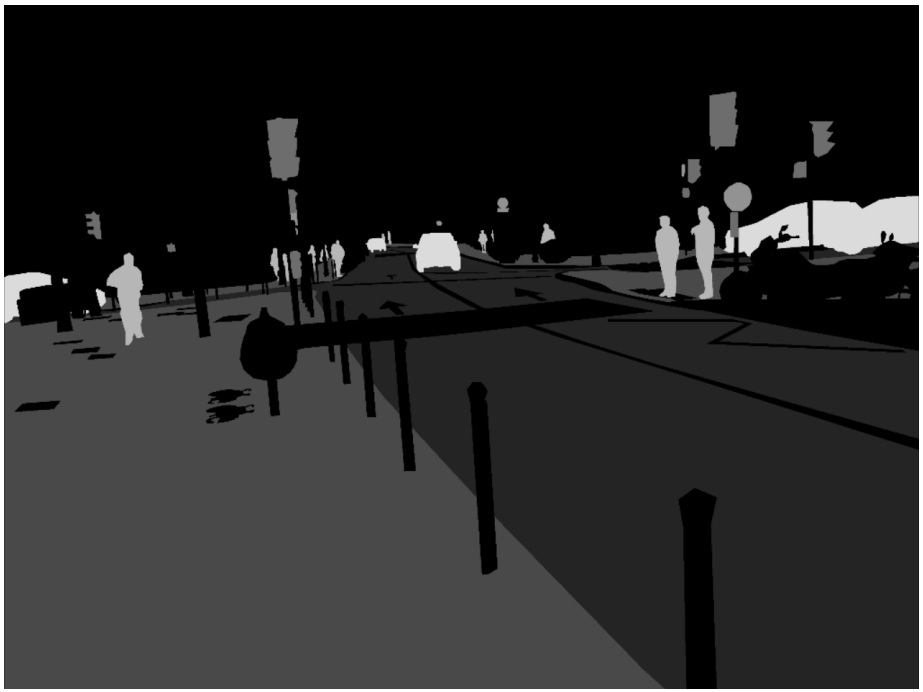
Provided by the Professor for fine-tuning the model.

Image

Ground Truth Grayscale

(processed from

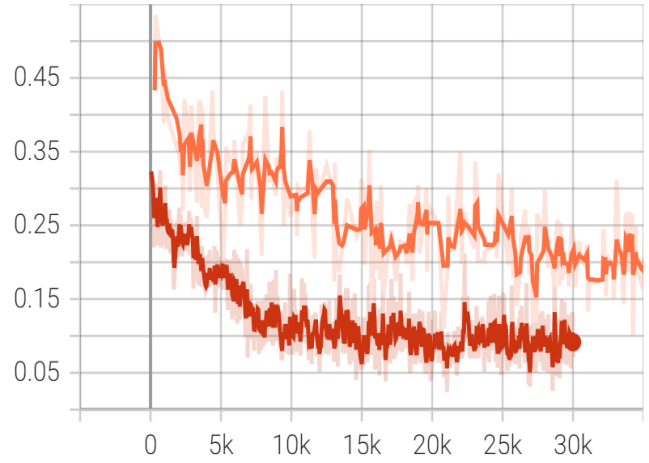
convert_masks_to_grayscale.py)



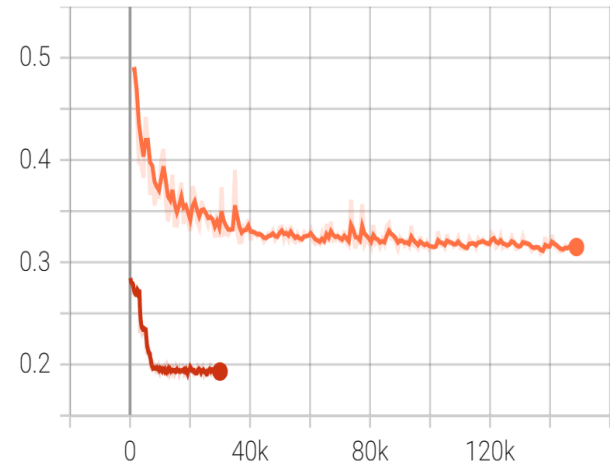
Initial Training on Cityscapes (21 classes)

fine-tuning on mapillary (8 classes)

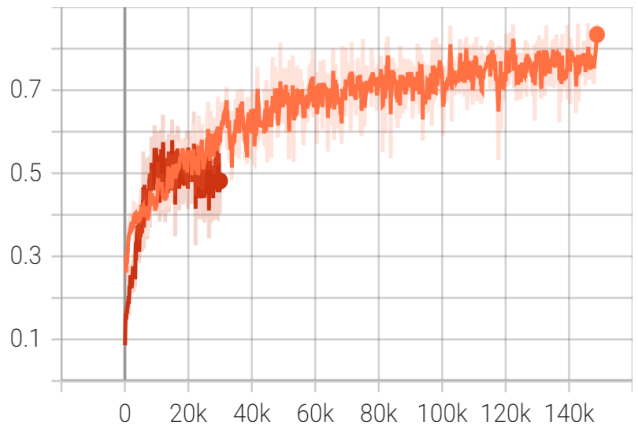
train_loss



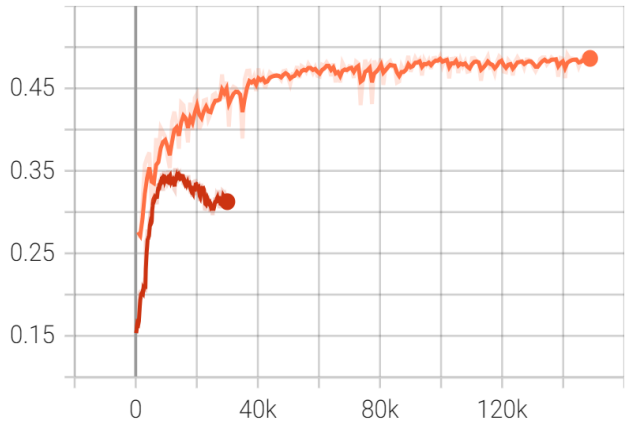
val_loss



train_iou



val_iou



Cityscapes – Orange

Mapillary - Red

num_epochs = 200

batch_size = 4

learning_rate = 0.001

Val_iou for training is around 46% for cityscapes and 32% for mapillary (not good)

Target Mean IOU for Cityscapes Dataset

Mean IOU for COCO val2017 dataset

https://pytorch.org/hub/pytorch_vision_deeplabv3_resnet101/

Model structure	Mean IOU
deeplabv3_resnet50	66.4
deeplabv3_resnet101	67.4
deeplabv3_mobilenet_v3_large	60.3

Mean IOU for Cityscapes dataset

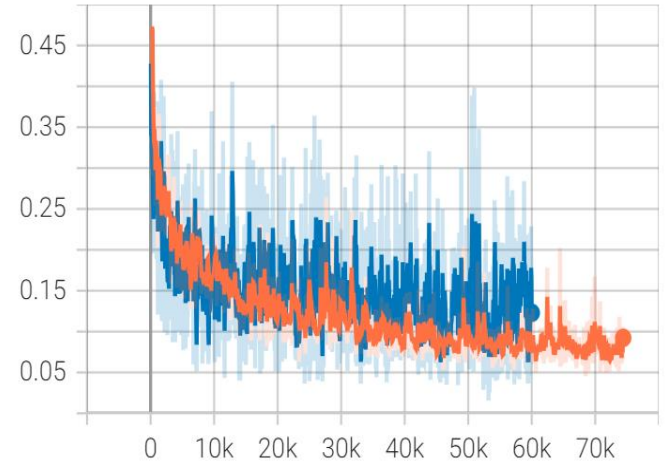
https://paperswithcode.com/sota/semantic-segmentation-on-cityscapes

Model	Mean IoU (class)
DeepLab	63.1%

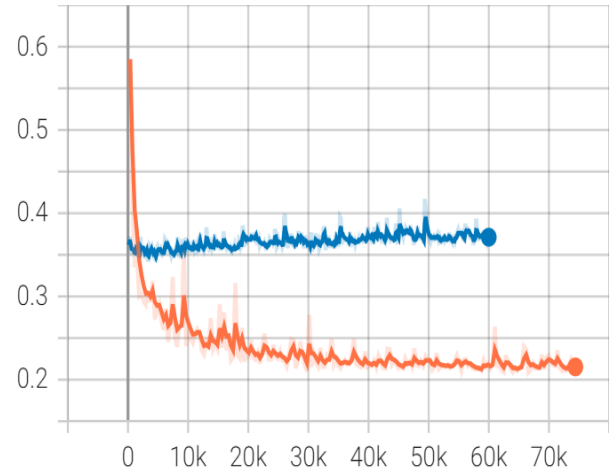
New Training on Cityscapes (8 classes)

fine-tuning on mapillary (8 classes)

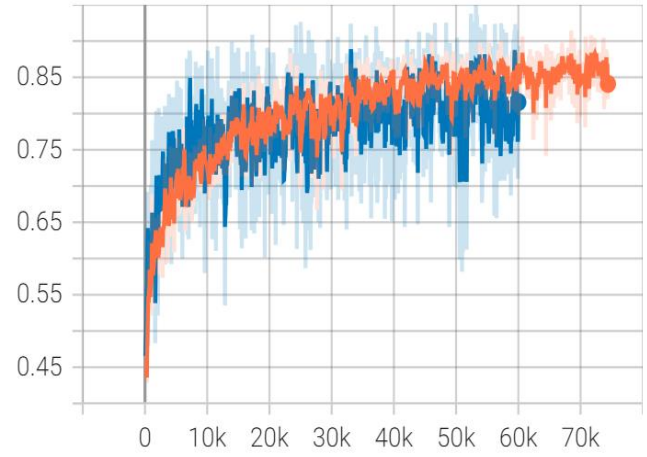
train_loss



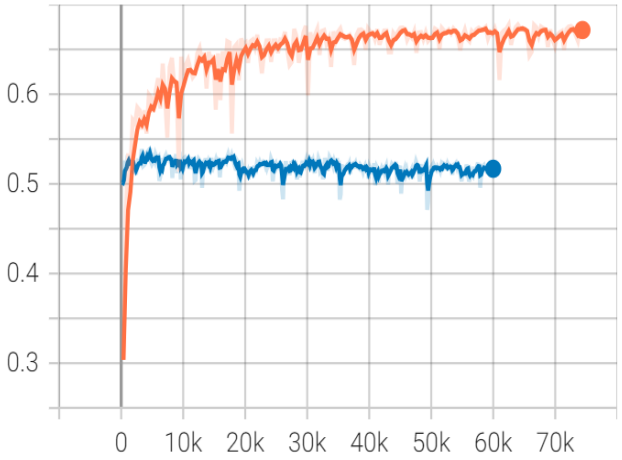
val_loss



train_iou



val_iou



Cityscapes – Orange

Mapillary - Blue

num_epochs = 200

batch_size = 8

learning_rate = 0.001

Val_iou for training is around 66% for cityscapes (close to the expected value as described in the previous slide)

Test scores on mapillary dataset

Test metric	DataLoader 0
test_iou	0.3197413682937622
test_loss	0.5499306917190552

Model only trained on cityscapes

Mean IOU = 31.97%

Test metric	DataLoader 0
test_iou	0.5382474660873413
test_loss	0.3432418704032898

Model fine-tuned on mapillary dataset

Mean IOU = 53.82%

21.85% improvement in Mean IOU

Inference

Model only trained on cityscapes

Model fine-tuned on mapillary

Inference

Model only trained on cityscapes

Model fine-tuned on mapillary

Segmentation Output

Combined with navigation data

Future Tasks

Possible tasks to work on

- Allow for detours/rerouting

Currently, the route information is calculated only once in the beginning of the walk. This method will break if the BVIP decides to take a detour (for e.g. due to construction) because the system does not perform rerouting.

- Ensemble Techniques

Train different networks then build an ensemble to improve segmentation mask prediction.

- Brightness

Apply color augmentations (random HSV) or train on images having different brightness/contrasts.

Thank You