

CS 6476 Project 6

Shuvo Newaz

shuvo.newaz@gatech.edu

snewaz3

903614132

Parts 4 & 5: mIoU of different models

Add each of the following (keeping the changes as you move to the next row):

	Training mIoU	Validation mIoU
Simple Segmentation Net (no pretrained weights)	0.3862	0.3526
+ ImageNet-Pretrained backbone	0.5464	0.5204
+ Data augmentation	0.4592	0.4917
ImageNet-Pretrained PSPNet w/ Data Aug. without PPM	0.6078	0.6207
+ PSPNet with PPM	0.5933	0.6130
+ PSPNet with auxiliary loss	0.5976	0.6338

Parts 4 & 5: Per class IoUs

Report your model's IoU for the 11 Camvid classes (you can find the order they are listed in at [dataset_lists/camvid-11/camvid-11_names.txt](#)):

Class Index	Class name	Simple Segmentation Net Class IoU	PSPNet Class IoU
0	Building	0.8638	0.8959
1	Tree	0.8647	0.8924
2	Sky	0.8662	0.9253
3	Car	0.6284	0.8331
4	SignSymbol	0.0000	0.0000
5	Road	0.9045	0.9411
6	Pedestrian	0.1615	0.3884
7	Fence	0.6593	0.6406
8	Column_Pole	0.0000	0.0000
9	Sidewalk	0.7362	0.8221
10	Bicyclist	0.2385	0.6333

Parts 4 & 5: Most difficult classes

[Which classes have the lowest mIoU? Why might they be the most difficult?
Provide an example RGB image from Camvid that illustrates your point]

SignSymbol and *Column_Pole* have the lowest mIoU. Poles and Signs are thin and small (not be confused with short). They contribute less to the segmentation loss than larger objects. An image from CamVid is shown in the next page. The boxed object is small compared to the car or the road.

Parts 4 & 5: Most difficult classes



Part 4: Simple segmentation net qualitative results

[Paste a figure of the generated semantic segmentation from Colab. It should be a 2x3 grid, with ground truth on the top row, and your predictions on the bottom row.]



Part 5: PSPNet qualitative results

[Paste a figure of the generated semantic segmentation from Colab. It should be a 2x3 grid, with ground truth on the top row, and your predictions on the bottom row.]



Part 6: Transfer Learning

Report your model's IoU for the Kitti Dataset.

	mIoU	mAcc/	allAcc
Train result	0.9279	0.9677	0.9763
Val result	0.9210	0.9623	0.9751

Class Index	Class name	iou	accuracy
0	Road	0.8720	0.9422
1	Not_Road	0.9700	0.9823

Part 6: Transfer Learning

Compare the training loss generated when training on Kitti dataset and Camvid dataset. Which decreases at a faster rate? If Camvid or Kitti training loss decreases at a faster rate than the other, why do you think this happened? Or, if the loss decreases at a similar rate, why do you think that is so?

In percentage reduction of loss, KITTI training loss decreases faster. We are distinguishing between two cases with KITTI as opposed to 11 for CamVid, which makes it easier to reduce the loss for KITTI.