Indian Driving Dataset – A Multi-label Problem

Individual Final Report

Author: Hemangi Kinger

12/13/2022

Overview of Project

Semantic image segmentation is a computer vision task in which we label specific regions of an image according to what's being shown. Mostly used in Autonomous vehicles and Medical Image diagnostics, the goal of semantic image segmentation is to label each pixel of an image with a corresponding class of what is being represented. Because we're predicting for every pixel in the image, this task is commonly referred to as dense prediction. One important thing to note is that we're not separating instances of the same class: we only care about the category of each pixel. In other words, if you have two objects of the same category in your input image, the segmentation map does not inherently distinguish these as separate objects.

IDD is a dataset for road scene understanding in unstructured environments used for semantic segmentation and object detection for autonomous driving.

Roles and Responsibility

Team Member	Area of Work	Shared Responsibility
Varun Shah	Model exploration	Fine-tuning
Hemangi Kinger	Model interpretation	Model exploration
Ishan Kuchroo	Data Preprocessing and Fine-tuning	Model interpretation

---

What is my responsibility?

I have taken the primary responsibility of data interpretation and model exploration for the Indian Driving Dataset. I'll be using the processed data and will explore various models and try to apply some fine tuning.

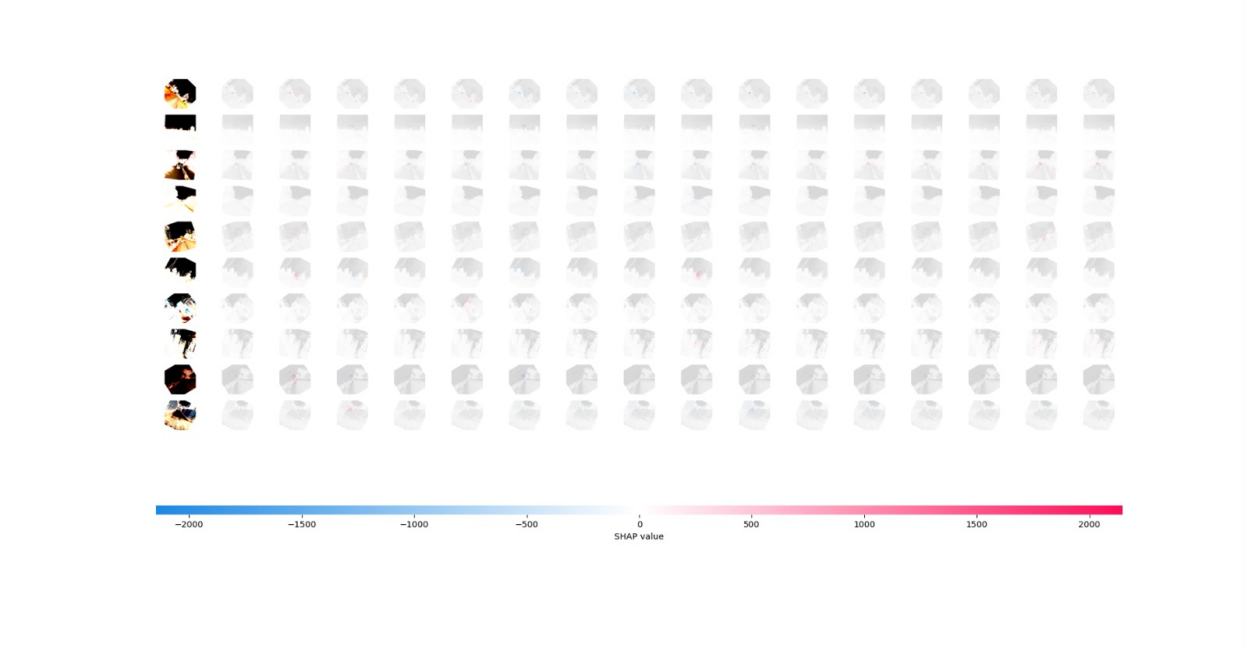
In addition to this:

- I'll be proof-reading and making changes in the summary report created by team
- Consolidating the code of data-preprocessing and modelling and creating a pipeline to ensure the code runs smoothly.

Model Interpretation

I have used SHAP partition explainer to explain the predictions made by our network. I have tried to explain both correct and wrong predictions of our model to see which parts of an image are contributing to predictions. We have generated image plots that visualizes shap values generated by the explainer object.

Below, we have generated the first image plot using shap values generated in previous cells. The chart shows the actual image and parts of it highlighted in shades of red and blue colors. The shades of red color show parts that contributed positively and shades of blue color show parts that contributed negatively to the prediction of that category. It also shows the sixteen categories that the model thinks the image belongs to.



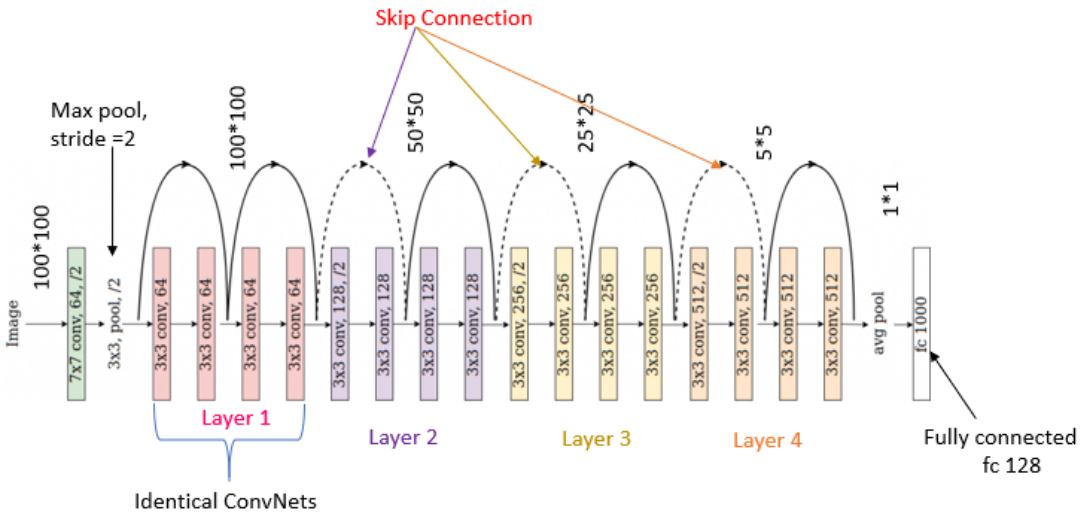
We can see the model does generate some pink and blue spots but it's not that good explanation hence we didn't include it in our main presentation.

Model Training and Fine-Tuning

Multiple models were trained and fine-tuned (including AlexNet, VGG16, NFNet, MobileNet, ShuffleNet, PNASNet etc.) but here we'll talk about the models I have worked on.

1. ResNet-18:

ResNet-18 is a convolutional neural network that is 18 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 224-by-224

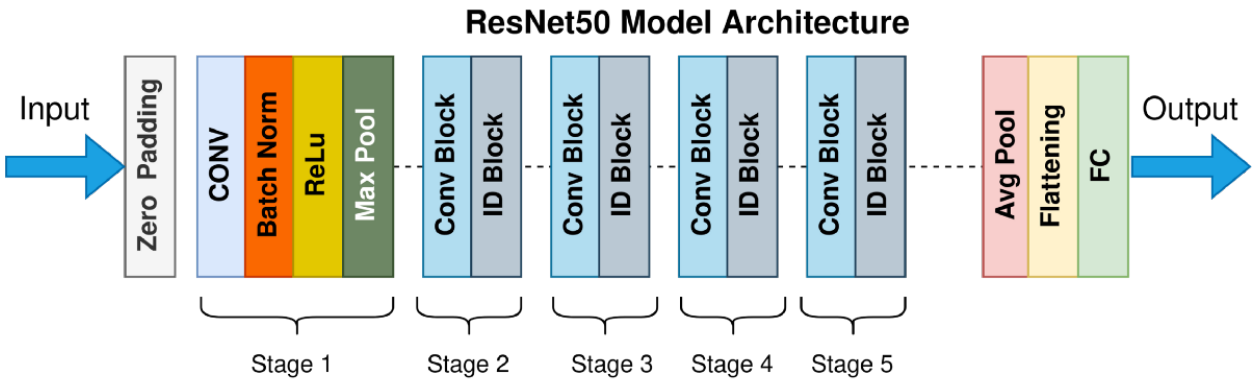


ResNet-18 Architecture

Fruit 360 Input Image size= 100*100 px

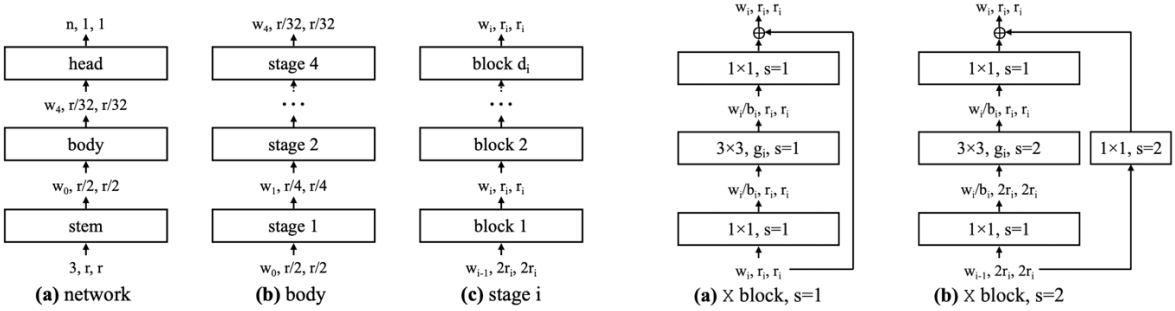
2. Resnet-50

ResNet-50 is a convolutional neural network that is 50 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 224-by-224.



3. RegNet:

Reg-Net is not a single architecture, it is a design space defined as a regulatory module for ResNet. The network is composed of multiple stage consisting of multiple blocks, forming a stem (start), body (main part) and head (end). Inside the body, multiple stages are defined, and each of the stages is composed of multiple blocks. There is only one type of block used in the RegNet which is the standard residual bottleneck block with group convolution.



Results

	1	2	3	4	5
Optimizer	Adam	Adam	Adam	Adam	Adam
Model	**Resnet18**	**Resnet50**	**Resnet50 (with different weights)**	**Regnet_x_800mf**	**Regnet_y_800mf**
Epoch	25	25	25	10	10
Learning Rat	0.0001	0.0001	0.0001	0.00001	0.00001
Batch Size	16	16	16	32	32
Image Size	224	224	224	400	400
Test Acc	0.43096	0.41736	0.45502	0.36402	0.3368
Test hlm	0.07263	0.07204	0.06917	0.08028	0.08362
Test sum	1.29249	1.27739	1.32941	0.4443	0.42044

Conclusion

From my analysis of different neural networks, we can conclude that ResNet module performs the best on our data.

Referenced Code %

(210 – 164)/ 210 + 52 * 100 = **17%**

References

https://shap.readthedocs.io/en/latest/example_notebooks/image_examples/image_classification/PyTorch%20Deep%20Explainer%20MNIST%20example.html

https://machinelearningmastery.com/how-to-implement-major-architecture-innovations-forconvolutional-neural-networks/

https://github.com/rishikksh20/MLP-Mixer-pytorch/blob/master/mlp-mixer.py

https://towardsdatascience.com/self-driving-car-on-indian-roads-4e305cb04198

https://medium.com/mlearning-ai/vision-transformers-from-scratch-pytorch-a-step-by-step-guide96c3313c2e0c

https://github.com/labmlai/annotated_deep_learning_paper_implementations

https://rwightman.github.io/pytorch-image-models/models/vision-transformer/

https://github.com/BrianPulfer/PapersReimplementations/blob/main/vit/vit_torch.py

https://medium.com/the-owl/imbalanced-multilabel-image-classification-using-keras-fbd8c60d7a4b