TECHNOVITY

Traffic Vehicle and Object Detection

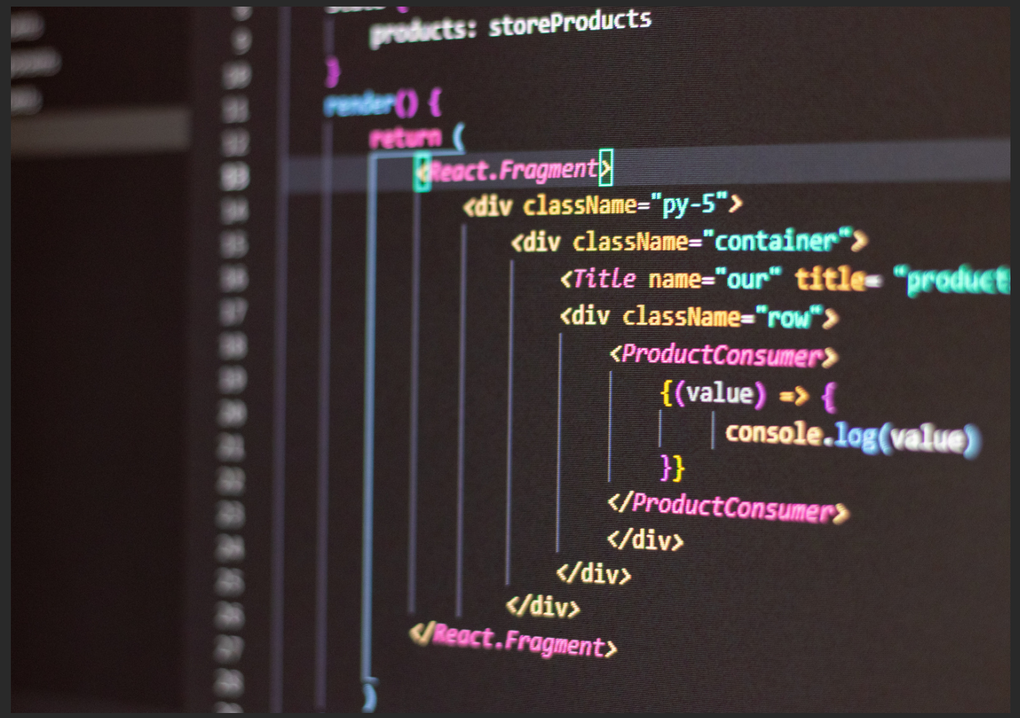
Navina G Amuthan
Ashmita A
Ajay Sundar Rajkumar
Sneha S

What's this all about?

Automation is omni-present! And when it comes to mobility, what's the main obstacle?
Self-driving cars and logistics - We need a coherent, efficient, feasible compendium of , methods to culminate this possibility.

What did we come up with?

Abstracts and debunking its assumptions and the flaws.



Initial Approach:

We aimed to use 3 Datstets: IDD Multi-modal primary for building, Secondary for validating hyper-paramenters and fine-tuning, and Supplement for testing. The massive amount of data was the challenge.

Middle Flaws:

Hence, we upgraded it to IDD Lite. And also used architecture of PSPNET to demonstrate along with ResNet34

Closure of the final approach:

To fuse in the algos was not possible, and we have stand-alone approaches for the front end and backend ML approaches.

Process Overflow and Technology

SEMANTIC SEGMENTATION

1

Algos:
ResNet34+PSPNet Architecture Model

OBJECT DETECTION + CLASSIFICATION

2

Algos:
YOLOv5 or v7 and EfficientDet D1

DUPLICATE + LANE DETECTION

3

Algos:
OpenCV, and SURF and SIFT

DISTANCE CALCULATION

4

Algos:
Manhattan Distance, Euclidean Theorem

TRAFFIC LIGHT COLOR DETECTION

5

Algos:
OpenCV

In [87]:
```
print("Validation mIoU Score : ",sum(mIoU)/len(mIoU))
```

```
Validation mIoU Score :  0.363701048404907
```

Observations

In [10]:
```
from prettytable import PrettyTable

# Representing results in a table
pt = PrettyTable()

pt.field_names = ["Model", "Highest Train Accuracy", "Highest Validation Accuracy", "Validation mIoU"]
pt.add_row(["PSPNet (from Scratch)",'89.76 %','80.80 %',0.43338])
pt.add_row(["PSPNet with a Pretrained Encoder",'69.438 %','66.919 %',0.24952])
pt.add_row(["PSPNet (from official models)",'89.91 %','83.08 %', 0.36370])
print(pt)
```

```
+---------------------------------+-----------------------+-----------------------------+-----------------+
|              Model              | Highest Train Accuracy | Highest Validation Accuracy | Validation mIoU |
+---------------------------------+-----------------------+-----------------------------+-----------------+
|      PSPNet (from Scratch)      |        89.76 %        |           80.80 %           |     0.43338     |
| PSPNet with a Pretrained Encoder |        69.438 %       |           66.919 %          |     0.24952     |
|  PSPNet (from official models)  |        89.91 %        |           83.08 %           |      0.3637     |
+---------------------------------+-----------------------+-----------------------------+-----------------+
```

- The best results are found using PSPNet from Scratch model.
- When we train PSPNet using a pretrained encoder, the results fluctuate a lot and the validation mIoU is very low as well.
- While using the official model, we get a high validation accuracy, but the validation mIoU score is low.

Future enhancements AKA Scalability Options

- KEEPING THE AMOUNT OF BRAKING HINDERANCES MINIMAL.
- DETECT OBSTACLES AHEAD AND SUGGESTING AN OPTIMAL DECELERATION AND ACCELERATION
- MOTION -DETECTION LIVE CONNECTIVITY TO NEARBY INFRASTRUCTURE LIKE TRAFFIC SIGNALS, CONTROL SYSTEMS

References

1. Sensor fusion for dynamic object detection in autonomous driving by Christina Lee Dao Wen
2. https://medium.com/analytics-vidhya/tensorflow-object-detection-api-for-indian-driving-dataset-idd-in-windows-10-f1f99c19b110
3. https://atharvamusale.medium.com/semantic-segmentation-on-indian-driving-dataset-3054cb2e70a7
4. https://www.connectedpapers.com/main/3dc3fa3ae91ea9ec6c3d263a9940f0c20f663cc6/Object-Detection-in-Autonomous-Vehicles%3A-Status-and-Open-Challenges/graph
5. https://rbcdsai.iitm.ac.in/projects/hybrid-intelligent-systems-in-autonomous-vehicles/