CSE 465: Pattern Recognition and Neural Network

Object Detection in Self-driving Car

Group Members

Md Saiyem Raiyan
2012468042

Zobaer Ahammod Zamil
2021796042

Sheikh Mohammed Wali Ullah
2021186042

Project

Current situation

Developing and implementing technology and algorithms that enable self-driving cars to detect and recognize objects in their surroundings, such as pedestrians, other vehicles, and obstacles, as a crucial aspect of ensuring safe and efficient autonomous driving.

Problems

1 Traffic Congestion

- How car object detection can be used to predict and prevent traffic congestion in urban areas.

2 Easy Handling

- High accuracy
- Robustness
- Real-time processing
- Seamless integration with other components to ensure safe
- Efficient autonomous driving

3 Vehicles Parking

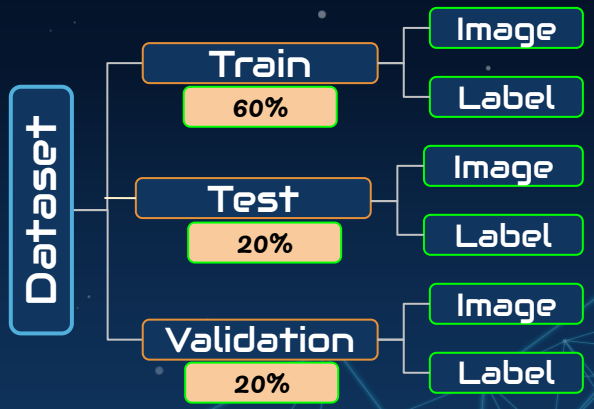
- Precise object detection
- Navigation to enable safe
- Efficient parking maneuvers.
- Identify parking spaces
- Guide the vehicle into them with minimal human intervention

Dataset Statistics

Process

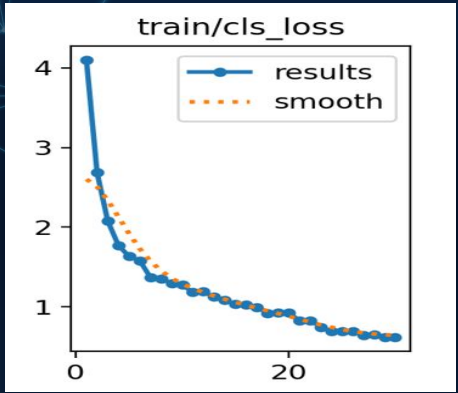
- **Dataset Collection** — From bing images and raw images of objects manually
- **Method** — Raw method, Download from bing by using codes
- **How to manage** — Annotated from ***MakeSense.ai*** and convert images to ***.xml*** and ***.txt*** files
- **Strategy of breakdown** — Split → 60, 20, 20 Convert image size according to each model

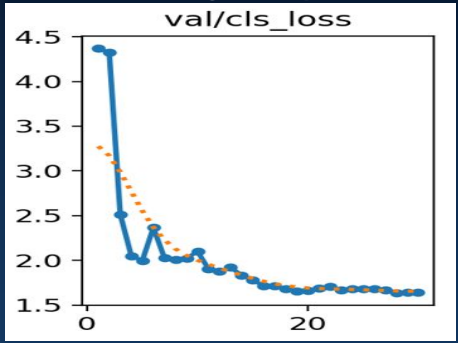
387	Size
29	Number of Classes
32	Number of samples



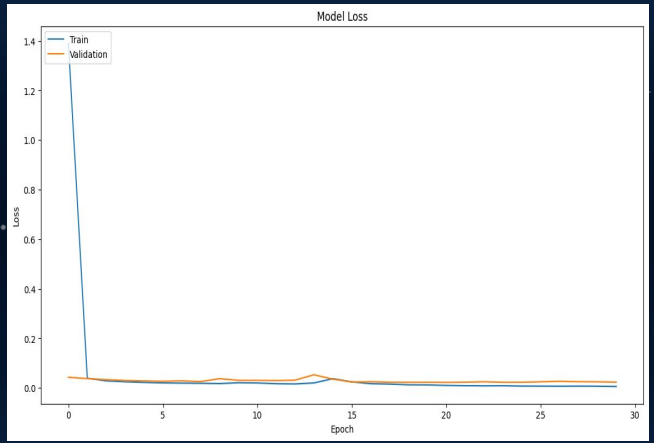
The Training Curve (learning and validation)

YoloV8s



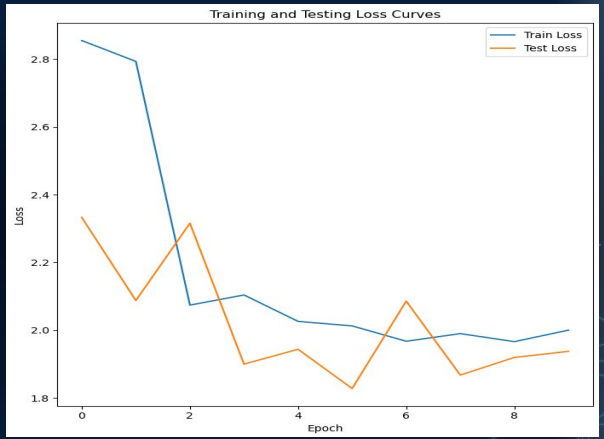


VGG16



Train epoch ⬆ → Training Loss ⬇

Vision Transformer



Training Loss ⬇ → Validation Loss ⬇

Training Loss ⬇ → Validation Loss ⬆ (Overfitting) → Complex model

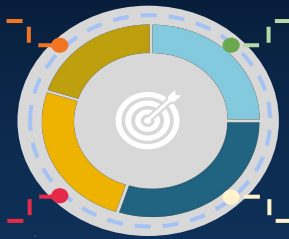
Training Loss ⬆ → Validation Loss ⬇⬆ (Underfitting) → Simple Model

Stats

	YoloV8s	VGG-16	Vision Transformer
Training Time	8 minutes	16.5 min	22 min
No of epoch	30	30	10
Avg. time per epoch	0.26 min	0.55 min	2.2 min

1 **No. of epoch Moderate**
Not enough training time for
Overfitting

2 **Epoch Size**
Boosts Precision up



3 **Changes Weight**
No. of epoch

4 **Epoch Affect**
No. of epoch ⬇
No learning (Underfitting)

The regularization and optimization used

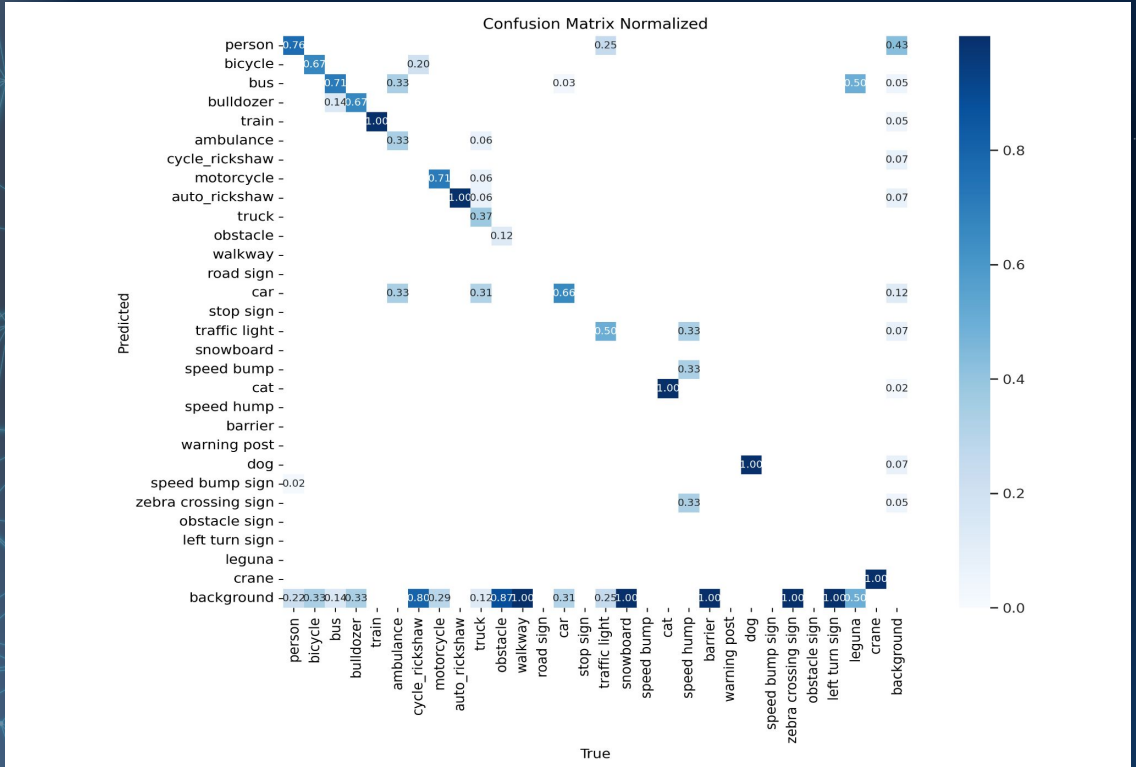
Regularization	A technique to prevent overfitting by adding a penalty term to the loss
Optimization	An algorithm that updates model weights during training
Loss Function	Measures the error between predicted and actual values

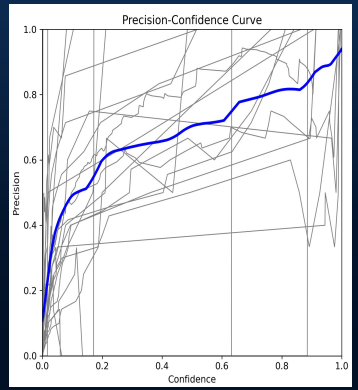
Used in Models:

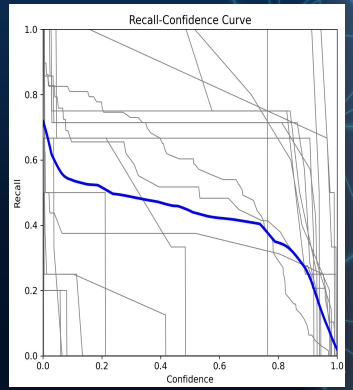
	YoloV8s	VGG-16	Vision Transformer
Optimizer	Adam	Adam	Adam
Regularization	Dropout = 0.0	L2 (Weight) Regularization	Dropout = 0.1
Loss Function	smooth_l1_loss	smooth_l1_loss	CrossEntropyLoss
Learning Rate	0.000303	0.0001	0.0003

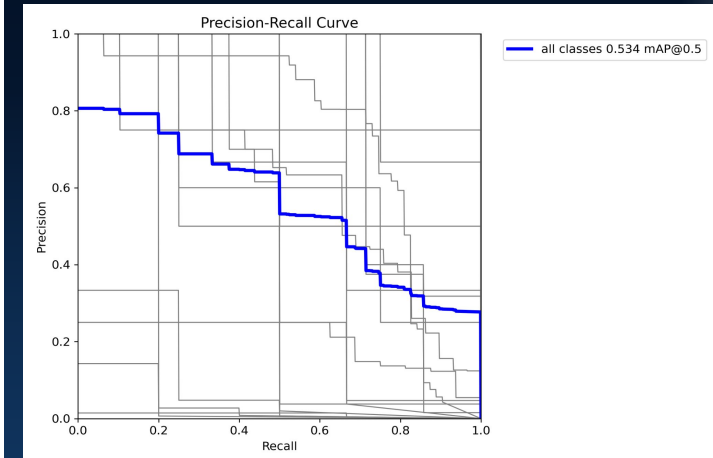
Confusion matrix, Precision, Recall Curve

YoloV8s





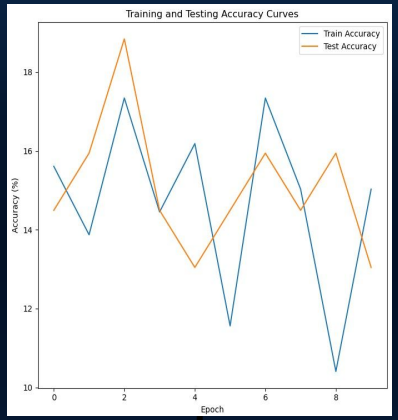




Accuracy, Precision, Recall Values

Vision Transformer

```
Epoch [1/10] - Train Loss: 2.8550
Train Accuracy: 15.61%
Epoch [1/10] - Test Loss: 2.3327
Test Accuracy: 14.49%
Epoch [2/10] - Train Loss: 2.7934
Train Accuracy: 13.87%
Epoch [2/10] - Test Loss: 2.0874
Test Accuracy: 15.94%
Epoch [3/10] - Train Loss: 2.0739
Train Accuracy: 17.34%
Epoch [3/10] - Test Loss: 2.3157
Test Accuracy: 18.84%
Epoch [4/10] - Train Loss: 2.1038
Train Accuracy: 14.45%
Epoch [4/10] - Test Loss: 1.8997
Test Accuracy: 14.49%
Epoch [5/10] - Train Loss: 2.0259
Train Accuracy: 16.18%
Epoch [5/10] - Test Loss: 1.9435
Test Accuracy: 13.04%
Epoch [6/10] - Train Loss: 2.0125
Train Accuracy: 11.56%
Epoch [6/10] - Test Loss: 1.8277
Test Accuracy: 14.49%
Epoch [7/10] - Train Loss: 1.9672
Train Accuracy: 17.34%
Epoch [7/10] - Test Loss: 2.0858
Test Accuracy: 15.94%
Epoch [8/10] - Train Loss: 1.9897
Train Accuracy: 15.03%
Epoch [8/10] - Test Loss: 1.8673
Test Accuracy: 14.49%
Epoch [9/10] - Train Loss: 1.9661
Train Accuracy: 10.40%
Epoch [9/10] - Test Loss: 1.9194
Test Accuracy: 15.94%
Epoch [10/10] - Train Loss: 1.9999
Train Accuracy: 15.03%
Epoch [10/10] - Test Loss: 1.9374
Test Accuracy: 13.04%
```



VGG16

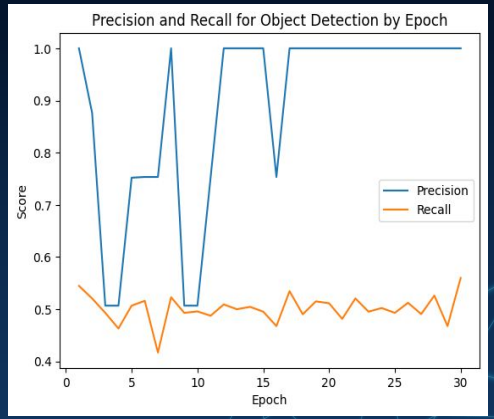
```
Confusion Matrix:
[array([[[  4,   0],
         [152,   0]],

        [[  4,   0],
         [148,   4]],

        [[  0,   0],
         [  0, 156]],

        [[  0,   0],
         [  0, 156]]])]

Test Loss: 0.0234
Test Accuracy: 0.7282
Test Precision for Object Detection: 0.7532
Test Recall for Object Detection: 0.5130
```



YoloV8s

epoch	metrics/precision(B)	metrics/recall(B)
1	0.66581	0.1193
2	0.51834	0.28132
3	0.6672	0.30479
4	0.69202	0.32388
5	0.49508	0.38924
6	0.69791	0.37793
7	0.69406	0.38289
8	0.6564	0.38319
9	0.76893	0.36548
10	0.74578	0.28762
11	0.65618	0.38817
12	0.64686	0.42286
13	0.53434	0.49697
14	0.58018	0.47945
15	0.74493	0.40879
16	0.79161	0.43938
17	0.6662	0.4655
18	0.68716	0.45197
19	0.69081	0.40992
20	0.6102	0.47651
21	0.57738	0.4772
22	0.58232	0.49736
23	0.66278	0.47364
24	0.6529	0.47479
25	0.61884	0.47245
26	0.62482	0.46509
27	0.58017	0.52188
28	0.5792	0.527
29	0.57865	0.52417
30	0.58384	0.5175

Thank You!