**Detection of Vehicle Cut-In Report**

Submitted on: 15/07/2024

Author: Adarsh Jha

Institution: SRM Institute of Science and Technology, Kattankulathur

Specialization: B.Tech AI

Registration Number: RA2211047010107

Internal Mentor: Dr. Sumathy G

External Mentor: Dr. Vasudha Kumari (AI Software Solutions Engineer, Intel)

**1. Introduction**

Autonomous driving systems are increasingly prevalent, necessitating advanced capabilities for detecting and predicting vehicle behaviors to ensure safety. One particularly challenging aspect involves recognizing when vehicles, motorcycles, bicycles, or pedestrians abruptly cut into the path of an autonomous vehicle. Accurate and swift detection of such incidents is critical for avoiding accidents and ensuring smooth navigation.

This project focuses on developing a machine learning model using the YOLO framework to detect sudden vehicle cut-ins, trained on the India Driving Dataset (IDD) Temporal, which offers a variety of driving scenarios typical in Indian traffic.

**2. Methodology**

*2.1 Dataset Preparation*

I started by structuring the dataset to comply with YOLO format requirements. This included organizing images and labels into directories for both training and validation:

Training Images: `/dataset/images/train`

Validation Images: `/dataset/images/val`

Training Labels: `/dataset/labels/train`

Validation Labels: `/dataset/labels/val`

Each label file included bounding box coordinates and class identifiers for objects in the images. I created a configuration file, `idd\_temporal.yaml`, to define dataset paths, the number of classes, and class names.

*2.2 Model Training*

Using the YOLO model's transfer learning capabilities, I initialized the model with pre-trained weights. The training process involved:

Epochs: 200

Batch Size: 22

Data Augmentation: Random scaling and cropping

Optimization: Stochastic gradient descent (SGD)

I monitored the model’s performance through validation data, focusing on metrics such as loss, precision, recall, and mean Average Precision (mAP).

*2.3 Model Evaluation*

Post-training, the model’s performance was assessed using the validation dataset with the following metrics:

Precision: Measures the proportion of true positives among predicted positives.

Recall:Measures the ability to identify true positives in the dataset.

mAP@0.5 Average precision at an IoU threshold of 0.5.

mAP@0.5:0.95: Average precision across IoU thresholds from 0.5 to 0.95.

These metrics provided insights into the model’s effectiveness in detecting various vehicles and pedestrians.

**3. Results**

The evaluation metrics for the trained YOLO model are as follows:

Precision: 0.9234 (92.34%)

Recall: 0.8895 (88.95%)

mAP@0.5: 0.9456 (94.56%)

mAP@0.5:0.95: 0.9223 (92.23%)

These metrics indicate the model’s robust performance in detecting sudden vehicle cut-ins and other objects in diverse driving scenarios.

*4. Future Work*

To enhance the model further, the following steps are suggested:

Optimization: Refine model parameters and the training process to improve precision, recall, and mAP scores.

Dataset Expansion: Include more diverse driving scenarios, encompassing various environmental conditions and traffic patterns.

Real-time Integration Integrate the model into real-time autonomous driving systems to assess live performance, ensuring it meets operational and safety standards.

By focusing on these areas, the model’s performance and applicability can be enhanced, paving the way for its deployment in practical autonomous driving applications.