DRIVER CLASSIFICATION

Students: Adithya Arvind, Yingyi Chen, Yao Hwang, Abhishek R Mandapmalvi, Hana McVicker, Tzuhua Peng, Kevin Zhao

Introduction

- Our project focuses on developing a computer program that utilizes inputs from on-board sensors to accurately recognize reckless driving behaviors, including speeding, weaving, and tailgating, exhibited by vehicles in close proximity to our test PACCAR truck.
- The license plate of the reckless drivers are recorded for future reference.

Objective

- By employing Hj reachability concept, autonomous vehicles can identify safe regions and plan collision-free trajectories.
- Understanding obstacle characteristics helps exponentially reduce computational power requirements, optimizing trajectory planning algorithms.
- Our project aims to decide the nature of previously encountered moving vehicles to create an optimized safety set of reachable states.

Requirements

- Define the features of Reckless Driving
- Machine Learning Models for Lane Detection and License Plate detection with accuracy above 60%
- Done using Vertex API, NodeJS, Google Vision API, Firebase
- 3D Object Detection Models and Trackers with both accuracy above 75%

Datasets and Models

DATASETS
- Kaggle Car License Plate Detection
- KITTI
- YOLOv5 Real Time Object Detection Algorithm
- Google Cloud AutoML

MODELS
- YOLOv5
- OpenPCDet
- AB3DMOT
- Google Cloud AutoML

License Plate Detection

YOLOv5 is deployed with simple functionality for Test Time Augmentation (TTA), model assembling, hyperparameter evolution, and export to ONNX. CoreML and TFLite is used to do object detection and OpenCV is used to implement text recognition. We also trained and tested a model on Google Cloud, extracted text using the Google Vision API, and deployed the model on a web application to show the detection working in Real Time.

Lane Detection

For lane detection, we first developed a version that uses computer vision algorithms to detect lane marks. We used the Canny edge detection to generate an edge map and used the probabilistic Hough transformation to detect the lane lines. The accuracy of this method can be further improved by preemptively cropping out non-road area on the image feed if the camera position is fixed.

Approach and Implementation

License Plate Detection Performance: 0.875

License Plate

Video Demo of: LiDar Tracking Functions

Distance estimation

Car License Plate Detection

Speeding and Distance

For Lidar part, we use OpenPCDet and AB3DMOT to detect and track the vehicle's position. OpenPCDet is used for 3D object detection with Lidar data input. The detection result is then sent to the AB3DMOT tracker and produce the tracking result.

Results


SPONSOR: PACCAR Technical Center