Lane Following in Autonomous Car Using CNN

Under Guidance of Prof Dr Bashir Morshed

Abstract:

This project demonstrates on how to use AI algorithms to use in developing ADAS features. We are using a CNN to compute the steering angle by feeding images from front camera of the car. We take the image from front camera and perform preprocess these images and train the CNN model using these preprocessed images and the corresponding steering angles. We were able to achieve a Model that was able to compute the steering angle and drive the car on the road.

Literature:

Initially we started with learning which algorithms are better fit for this application and we explored various AI algorithms. Then we decided to use CNN. Then we focused on developing a prototype using python pygame where we developed a small game where a car will be moving on the track and sensors on all sides of the car will give us the distance from the sensor to edge of the track/obstacle. As the pygame interface directly work on raw pixels it was hard to create environment to train the model. Then we moved onto Quanser simulator which can be used to simulate car and have may components built in. The components include camera, lidar, depth camera and other objects like traffic lights, sign boards, crosswalks. This simulator can be used to simulate real world. We have also used image processing to extract some important features from the image.

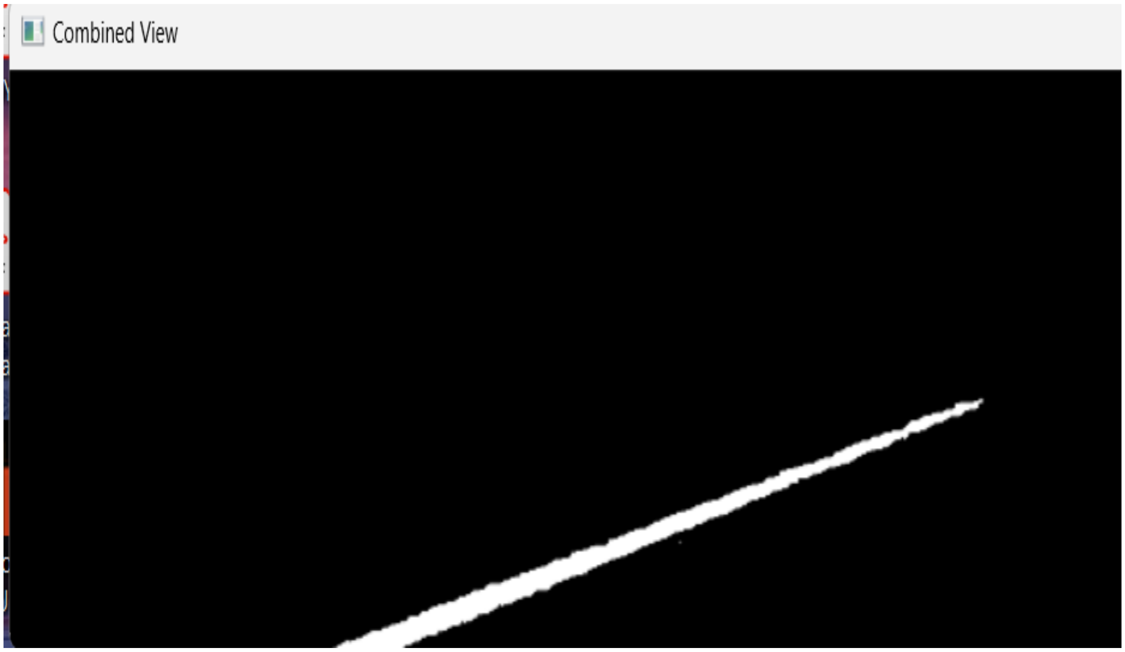
Quanser Simulator is like a digital twin for simulation of autonomous cars. Digital twin is virtual replica of physical vehicle and its environment. Digital twins enable us to simulate critical environments which are not practical to simulate using physical car and real word environment. These are helpful as they incorporate advanced features to integrate AI into the car.

Implementation:

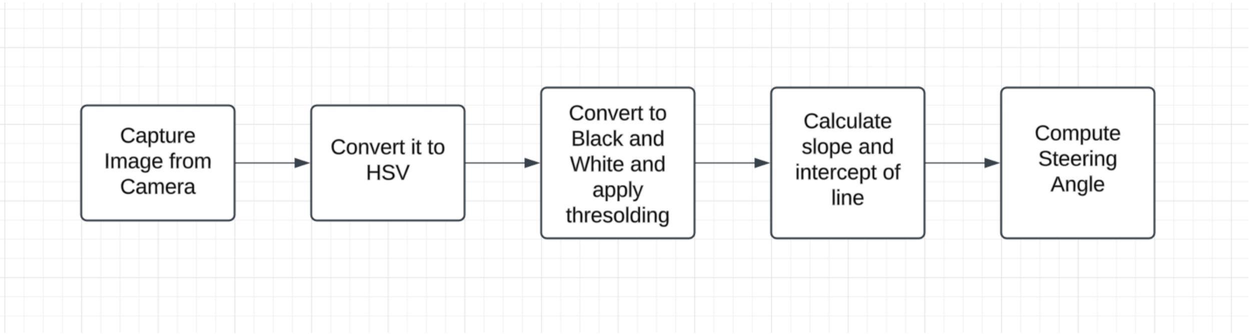
With Quanser simulator we have performed 2 methods to compute steering angles. In the first method we have calculated steering angle using slope. In the second method we have used CNN for predicting the steering angle from the input image.

Method 1: Steering angle from slope

In this method we calculate steering angle using the slope of the yellow line that is on the road. We take the input from the front camera of the image. We then convert the image from RGB to HSV. We then do the binary thresholding for yellow line so that we get yellow line as white and rest all pixel's image as black. We calculate the slope and intercept of this yellow line. We then use this slope and then compute steering angle from this slope and intercept.



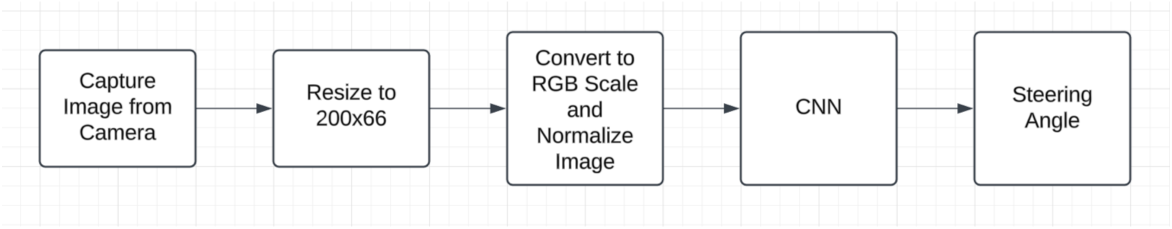
Processed image for slope calculation.


Capture Image from Camera
Convert it to HSV
Convert to Black and White and apply thresolding
Calculate slope and intercept of line
Compute Steering Angle

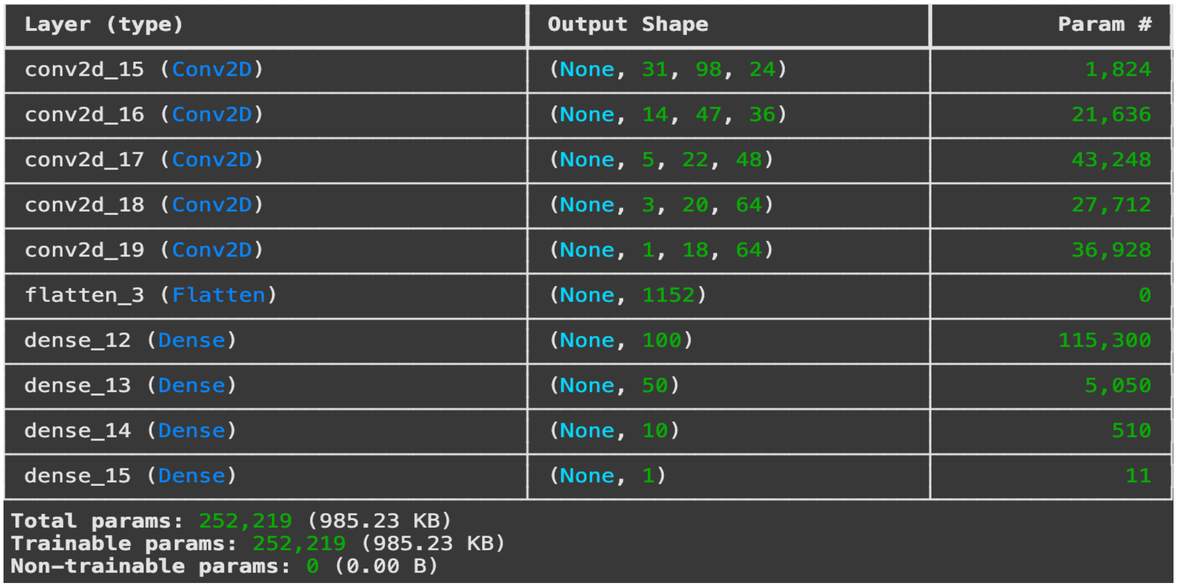
Flow Diagram for steering angle using slope

Method 2: Steering angle from CNN

In this method we calculate the steering angle by giving image as input to the Convolutional Neural Network (CNN). In this method we get the image from the front camera and then resize it to 200x66. We then normalize the image by dividing it by 255. We then train the CNN using the images and steering angles that were collected during data collection step. Which we discuss in later section. We have trained the CNN using 10 different data sets that we generated during data collection step. We combined all the data sets and used it to train the CNN. Once the CNN is trained, we download the model and load it into the file locally. We can then use this model by inputting the image and we get the steering angle. We can directly use this steering angle from CNN to steer the car.


Capture Image from Camera
Resize to 200x66
Convert to RGB Scale and Normalize Image
CNN
Steering Angle

Flow Diagram for steering angle using CNN



```
Layer (type)                  Output Shape                    Param #
conv2d_15 (Conv2D)            (None, 31, 98, 24)                1,824
conv2d_16 (Conv2D)            (None, 14, 47, 36)               21,636
conv2d_17 (Conv2D)            (None, 5, 22, 48)                43,248
conv2d_18 (Conv2D)            (None, 3, 20, 64)                27,712
conv2d_19 (Conv2D)            (None, 1, 18, 64)                36,928
flatten_3 (Flatten)           (None, 1152)                          0
dense_12 (Dense)              (None, 100)                     115,300
dense_13 (Dense)              (None, 50)                        5,050
dense_14 (Dense)              (None, 10)                          510
dense_15 (Dense)              (None, 1)                            11
Total params: 252,219 (985.23 KB)
Trainable params: 252,219 (985.23 KB)
Non-trainable params: 0 (0.00 B)
```

CNN Architecture

Data Collection:

We have modified the code such that we can get control the car in simulator using the direction arrows on the keyboard. We store the image and steering angle at that instant when the function to save image is called. We are saving images and steering angles in a folder locally on the system.

Future scope:

In this project we have kept the speed of the car to be constant and the steering angle to be varying. But in real world both are varying so as enhancement to this project we can make speed and steering able to be varying. So that we can have better control at turnings and when we have obstacles on the way.