AUTONOMOUS SELF DRIVING CAR

Bangalore institute of technology

Computer science Dept.

-Abdullah Abdul Sattar Shaikh

1BI18CS003

BACKGROUND

The autonomous car or the driverless car can be referred to as a robotic car in simple language. This car is capable of sensing the environment, navigating and fulfilling the human transportation capabilities without any human input. It is a big step in the advancing future technology. Autonomous cars sense their surroundings with cameras, and navigational paths.

With the exponential development of image processing techniques, a car can navigate to their destination with just a single camera module. To demonstrate this ability, we have built a working prototype of car which travels around a guided path using only images s input and inferencing street signs and act accordingly.

Technologies Used

- Python- For basic input and scripting functions.
- Deep learning – Used convolutional neural networks (CNN) by implementing TensorFlow 2.0.0 framework.
- Open-cv – For image pre-processing, augmentation and HAAR cascade object detection.
- Raspberry-Pi – Used for computation of the model in real-time.
- L298- for precise motor control of the car.

Procedure

1.Data collection:

- To train the CNN model the first requisite was data, which was obtained by taking ~3000 images of a road track having black lines as its guidance, these images were taken in the perspective of the car for better performance.
- This data was then processed to save computational time and increase accuracy by performing augmentation techniques.

2.Training the model and testing for optimal performance:

- The data was trained by pushing it though a CNN consisting of 4 convolutional layers followed by 2 dense layers for 50 epochs.
- This model will be able to classify between going straight, left or right based on its input.
- At the end of the epoch a model was generated having optimal accuracy on Testing.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_4 (Conv2D)            (None, 126, 126, 64)      1792
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 63, 63, 64)        0
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 61, 61, 64)        36928
_________________________________________________________________
max_pooling2d_5 (MaxPooling2 (None, 30, 30, 64)        0
_________________________________________________________________
conv2d_6 (Conv2D)            (None, 28, 28, 128)       73856
_________________________________________________________________
max_pooling2d_6 (MaxPooling2 (None, 14, 14, 128)       0
_________________________________________________________________
conv2d_7 (Conv2D)            (None, 12, 12, 128)       147584
_________________________________________________________________
max_pooling2d_7 (MaxPooling2 (None, 6, 6, 128)         0
_________________________________________________________________
flatten_1 (Flatten)          (None, 4608)              0
_________________________________________________________________
dropout_1 (Dropout)          (None, 4608)              0
_________________________________________________________________
dense_2 (Dense)              (None, 512)               2359808
_________________________________________________________________
dense_3 (Dense)              (None, 3)                 1539
=================================================================
Total params: 2,621,507
Trainable params: 2,621,507
Non-trainable params: 0
_________________________________________________________________
```

3.Building the car & Inference:

- The car was built by placing a Rasbery-Pi on top off a chassis and connected to 4 motors using an L289 for motor speed control. A camera module was also connected to the raspberry-pi for real time input.
- The model was loaded into the Rasbery pi and scripts were hardcoded for motor control on each result of the model.

- A STOP sign HAAR cascade was also implemented for sign detection and the car would stop all movement on detection of a STOP sign.
- The car was then placed on the guided track, the camera took input from the track and pushed it through the model. The model then classified the input images for best predictions upon which the motors of the car were able to determine the direction to move.

4.Conclusion:

- Autonomous cars can efficiently move around a closed path with only its image data and also detect road sign for reduced risk of accidents.
- This prototype proves the possibility of Autonomous of more advance robots which can traverse and have better judgement on its decisions by the implementation of GPS, proximity sensors and other enhancements.