Autonomous Car Simulation using Soft-Computing

Aman Bhanse[1], Mandar Gondane[2], Akshay Sargar[3], Kapil Kadadas[4], Prof. Dr. Shrishailappa Tatyasaheb Patil[5]

[1]*Department of Computer Engineering & College & Vishwakarma Institute of Technology*
(Affiliated to Pune University)
Pune, India.
[2]*Department of Computer Engineering & College & Vishwakarma Institute of Technology*
(Affiliated to Pune University)
Pune, India.
[3]*Department of Computer Engineering & College & Vishwakarma Institute of Technology*
(Affiliated to Pune University)
Pune, India.
[4]*Department of Computer Engineering & College & Vishwakarma Institute of Technology*
(Affiliated to Pune University)
Pune, India.
[5]*Department of Computer Engineering & College & Vishwakarma Institute of Technology*
(Affiliated to Pune University)
Pune, India.

---***---

Abstract - In the modern era, vehicles are focused to be automated to give human driver relaxed driving. In the field of automobile various aspects have been considered which makes a vehicle automated. Nvidia, the leading gaming industry has started working on the self-driving cars since 2016 and still making and adding some new feature according to the environment. In this paper we have mainly focused on Lane detection, Speed control and steering angle. Our system automatically learns internal representations of the necessary processing steps such as detecting useful road with only the human steering angle as the training signal with help of keyboard (WASD). Here we are collecting/creating our data using the Udacity Car Simulator and then using it to train the model with the help of TensorFlow. For now, we plan to create a virtual environment where the the car will be running autonomously with all the objects in environment taken into consideration like pedestrians and traffic lights, etc.

Key Words: Simulation, Autonomous Car, Udacity Car Simulator, TensorFlow

1. INTRODUCTION

Automated vehicles are technological development in the field of automobiles. Although the automated are for ease of human kind, yet they are very expensive vehicles. In this paper we are taking into the consideration the various features of the environment and the cost, and making a autonomous car simulation, which has been designed to stay on the track.

We have mainly focused on a application of an self-driving car here and designed a simulated model of a car. During the procedure of building such an autonomous system, many concerns arise. Does each of its algorithms work correctly? Is it robust enough? Can the system adapt in different real traffic scenarios and perform well? Simulation testing is an effective way to test, validate and evaluate the performance, the adaptability to the environment, and the stability of an unmanned car system and algorithms besides field testing. We are going to use Udacity Car simulator and Anaconda Python environment to train the model.

2. LITERATURE

After the development of the autopilot airplanes [1], self-driven sailboats and ships; the deceptively modest dream that has rarely ventured beyond the pages of science fiction is the self-driving car [2]. The development of the digital computer made possible to dream of self-driven vehicles outside the fiction. By the 1960s the self-driven cars have been dreamed to navigate on ordinary streets on their own. German pioneer Ernst Dickman's, in the 1980s, got a Mercedes van to drive hundreds of miles autonomously on highways, a tremendous feat especially with the computing power of the time [2]. In the mid-2000s, the Defense Advanced Research Projects Agency (DARPA) sorted out the Grand Challenges where groups assembled to contend with self-driving vehicles. In 2009, Google began the self-driving car venture, including colleagues who had effectively devoted years to the innovation. By 2012 the Google car hits the road for testing. By the passing years, the car is developed and equipped with multiple sensors, radars, lasers, Global Positioning System (GPS), it uses heavily detailed maps, and many other things to safely drive and navigate itself with no human interaction. The car can not only drive itself but it can be parked on its own, it can go on freeways, Cameras are used to find and detect objects that are then processed by the computer within the car [3]. In May 2014, Google presented a new concept for their

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 11 | Nov - 2023 SJIF 2023: 8.176 ISSN: 2582-3930

driverless car that had neither a steering wheel nor pedals and unveiled a fully functioning prototype in December of that year that they planned to test in 2015[4]. In summer 2015, Google launched and tested some different features where each prototypes speed is capped at a neighborhood-friendly 25mph, and during this phase safety drivers aboard with a removable steering wheel, accelerator pedal, and brake pedal that allow them to take over driving if needed [5]. After many successful roads testing of Google car has made to believe in some year's roads will be safely occupied with self-driven cars.

3. OBJECTIVE

- To make sure that no accidents take place, even while the driver is not actively focused on driving.
- Person with disability with also be able to travel safely in his/her car with minimum risk.
- To maintain the flow of traffic in a good way, and hence even might reduce the traffic in daily scenarios
- More free time and better health because traffic jam has inverse effect on the blood pressure, and body.

4. FLOWCHART

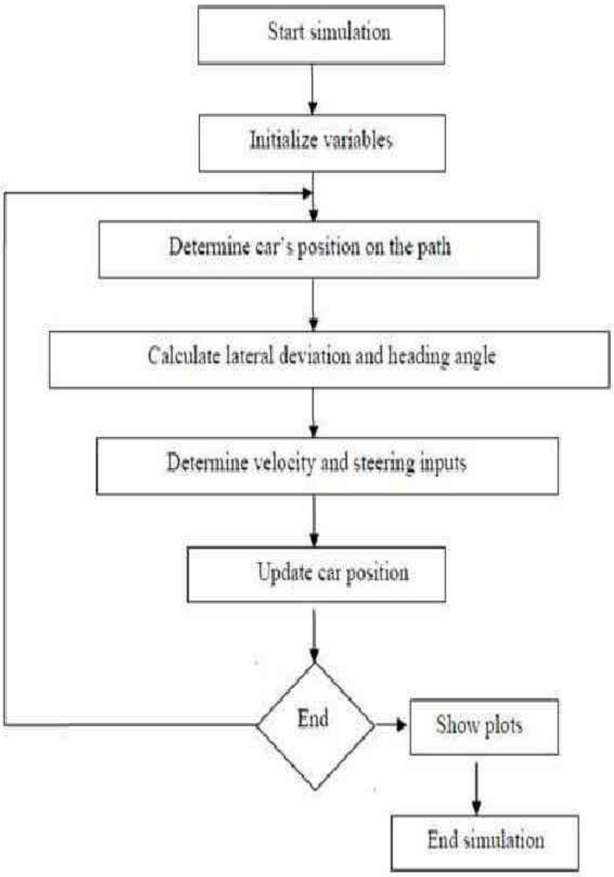

Fig -1: Flowchart

4. ARCHITECTURE

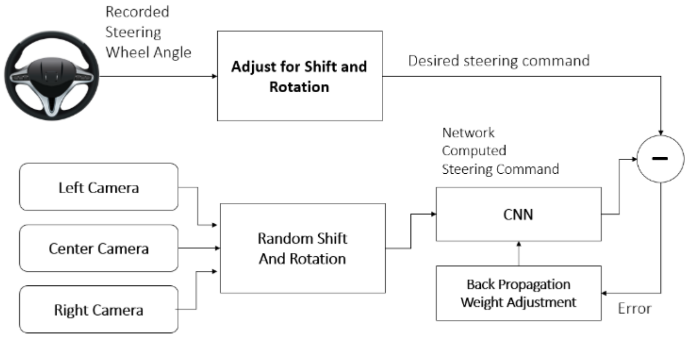

Fig -2: Architecture of the designed model

5. IMPLEMENTATION METHODOLOGIES

Using the Udacity Car simulator, we have collected the data of 10 laps around the track, our data basically consists of photos from right camera, center camera, left camera, Steering angle, Speed Control (acceleration and retardation). We have used Conda Python to create a virtual environment and with the help of TensorFlow and our dedicated GPU we trained our data set. The dependencies which we installed into python to train this model are Keras, NumPy, Scipy, TensorFlow, Pandas and OpenCV. As we are using CNN which in our case is of 9 layers, the more data we give as input the more accurate model will run and follow its path. Also, with time the model grows more and more accurate cause it will be collecting the data and training itself.

A. Data Capturing

During the training, the simulator captures data. Also, at a given time step it recorded three images taken from left, center, and right cameras. The following figure shows an example I have collected during the training time. The collected data is processed before feeding into the deep neural network.

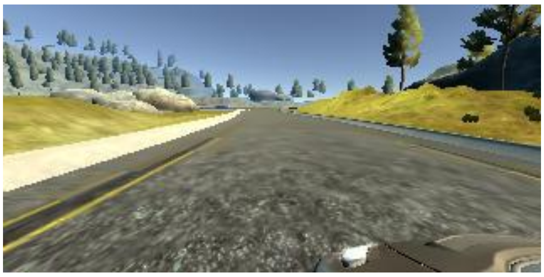

Fig -3: Image from front left camera

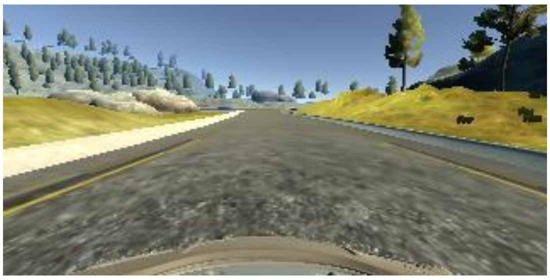

Fig -4: Image from front center camera

Fig -5: Image from front right camera

B. Data Statistics

The dataset consists of 30072 images (10024 images per camera angle). The training track contains a lot of shallow turns and straight road segments. Hence, the majority of the recorded steering angles are zeros. Therefore, preprocessing images and respective steering angles are necessary to generalize the training model for unseen tracks such as our validation track.

C. Network Architecture

Our convolutional neural network architecture was inspired by NVIDIA's End to End Learning for Self-Driving Cars paper. The main difference between our model and the NVIDIA mode is that we did use MaxPooling layers just after each Convolutional Layer to cut down training time. For more details about our network architecture please refer following figure

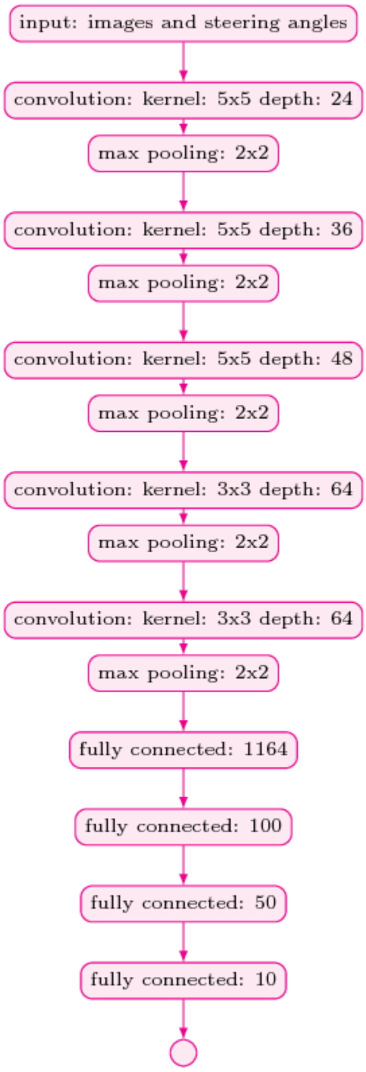

Fig -6: CNN Network Architecture

D. Training

We used 30072 (10024 images from each camera) Batch size was 64 and, the steering angle, and speed were taken while training. The model was trained using TensorFlow and our dedicated GPU Nvidia GTX 1650. The model got trained in around 10 hours.

Fig -7: Training Simulation

E. Testing

We have trained the data in a different track and simulated in a different track. As expected, the model was able to follow the track on the other track.

F. Result

In the initial stage of the project, I used a dataset generated by myself (10000 images). That dataset was small and recorded while navigating the car using the laptop keyboard. However, the model built using that dataset was not good enough to autonomously navigate the car in the simulator. So, to improve this we increased our database, and trained it again with 30000+ images in the dataset.

The model developed using that dataset works well on both tracks (one used for training, the other track is used to check if the trained model is working well).

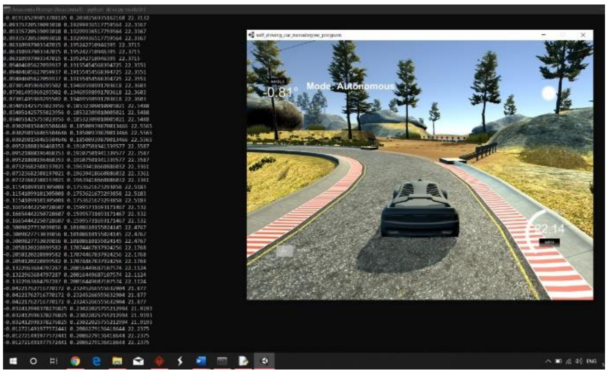

Fig -8: Autonomous Simulation

6. SALIENT FEATURES

- To maintain flow of Traffic in a proper manner, hence reducing the travel time.
- Autonomous vehicles will eliminate ineffective speeding up and braking, operating at an optimum performance level to achieve the best possible fuel efficiency.
- Intelligent Transporting.
- Use of CNN to make the model more and more accurate by time and experience.

7. FUTURE SCOPE

- The addition of features like automatic pick-up and drop will be very useful.
- We can add furthermore the detection of Signals, crosswalks and pedestrian.
- If we Implement this in all the cars then, there won't any accidents taking place and far less traffic Jams.

ACKNOWLEDGEMENT

We express our sincere thanks to all those who have provided us the valuable guidance towards the successful completion of this system as a part of syllabus for the bachelor's course. We express our sincere gratitude to our co-operative department for providing us with altheas valuable assistance and equipment for the system development. We hereby take this opportunity to sincerely thanks Prof. Shrishailappa Patil for his valuable guidance, inspiration, wholehearted involvement during every stage of this project and his experience, perception through professional knowledge which made it possible for us in successfully realizing the concept. We are also thankful to Prof. Sandip Shinde - Head of Department – Computer Engineering for his constant enlightenment, support and motivation which has been highly instrumental in the successful completion of our project phase. Finally, we would like to express our deep sense of gratitude towards our parents, friends and well-wishers who were always there for suggestions and help.

REFERENCES

1. M. Frutiger and C. Kim, "Digital Autopilot Test Platform with Reference Design and Implementation of a 2-Axis Autopilot for Small Airplanes," Department of Electrical and, pp. 1–24, 2003.
2. M. Weber, "Where to? A History of Autonomous Vehicles," 2014.
3. Product Details – Google Car
4. Fisher, "Inside Google's Quest to Popularize Self Driving Cars," Popular Science, 2013. [Online]. Available: http://www.popsci.com/cars/article/2013-09/google-self-driving-car
5. "Official google blog: Green lights for our self-driving vehicle prototypes."