Artificial Intelligence based Self-Driving Car

Hiral Thadeshwar
Department of Information Technology
A. P. Shah Institute of Technology
Thane (MH), India 400615
hiralthadeshwar13@gmail.com

Vinit Shah
Department of Information Technology
A. P. Shah Institute of Technology
Thane (MH), India 400615
shahvinit95@yahoo.com

Mahek Jain
Department of Information Technology
A. P. Shah Institute of Technology
Thane (MH), India 400615
jmahek09@gmail.com

Prof. Rujata Chaudhari
Department of Information Technology
A. P. Shah Institute of Technology
Thane (MH), India 400615
rhchaudhari@apsit.edu.in

Prof. Vishal Badgujar
Department of Information Technology
A. P. Shah Institute of Technology
Thane (MH), India 400615
vsbadgujar@apsit.edu.in

2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP) | 978-1-7281-6509-7/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICCCSP49186.2020.9315223

***Abstract*—The paper proposes a self-driving car model also called autonomous, robotic or driver-less car is one that operates and navigates using its intelligence. The basic idea behind the paper is to develop a 1/10 scale RC car to portray an automated car. The model consists of the following software and hardware components such as CNN (Convolutional neural network), Monocular vision algorithm, Haar cascade classifier, Raspberry Pi Board model B+, Pi camera, Arduino, and an Ultrasonic sensor. The Pi camera and ultrasonic sensor are attached to the raspberry pi board to collect input images along with sensor data to stream these data to the server which in our case is the laptop. The (CNN) convolutional neural network running on the server will be used to enable lane detection to provide steering predictions that are left, right, forward based on the input image. The haar cascade classifier will be used to detect signals and stop sign and monocular vision algorithms to calculate distance from them. The ultrasonic sensor will be used for front collision avoidance by stopping the car at a certain distance before the obstacle ahead. The navigation commands such as right, left, forward, stop will be sent to the car through Arduino which is connected to the RC car's remote, this will make the car drive autonomously based on neural network predictions and some hard coded rules. Thus this model will enable autonomous driving by self-navigation via lane detection, stopping at detection of stop signs, red lights and moving on green signal and front collision avoidance in a cost-effective manner. Hardcoded rules, obstacle avoidance mechanisms, machine learning models, and smart object discrimination will help the system follow traffic rules and navigate the car using artificial intelligence.**

Keywords—Self-driving car, real-time navigation, convolutional neural network, lane detection, Signal detection, Stop sign detection, machine learning, monocular vision.

I. Introduction

Human-driven cars use technologies to provide safety and detect obstacles and auto stops in various high-end cars but none of them works completely driverless. The existing cars do not contain the feature of automation to the extent that cars can drive autonomously. There is a constant need for drivers without it the car becomes unavailable but with self-driving cars, we can make the availability of cars constant on roads. In traditional cars, the driver constantly needs to keep a check on the signals, road safety signs, obstacles and lanes and needs to make decisions accordingly. Self-driving is no longer a futuristic dream, but it's becoming a reality. Every week, companies are proclaiming their dedication to develop and launch autonomous vehicles, and many of them talk about the "level" of autonomy being developed. Autonomous driving can surely be frightening to some, but it has its benefits too. It will result in reduced traffic congestion, lowered pollution, cheaper transportation costs for everyone as well as a reduction in the cost of new roads and infrastructure. It would also immensely ameliorate the mobility of old and physically disabled people.

The areas researched are self driving model cars in a model area with few sensors[7] as Tesla and such cars use numerous sensors like lidar and radar. The approach with which we aim to gain autonomy of cars is by modelling it on 1/10 scale rc cars. The car will be able to sense its environment with the aid of pi camera and ultrasonic sensor and data collected by both are streamed to the server via raspberry pi to the server on which we will run the neural network that will process the images to detect the lane marking, also haar cascade classifier will be used to detect stop sign and traffic signal and sensor data will be processed to avoid front collision by braking the car at detection of obstacle at certain distance. The car will drive according to the lane marking autonomously taking the navigational decisions on its own once it's trained. The same algorithm and techniques can be used in actual vehicles to implement automation.

II. Literature Survey

In literature[1], in this paper, they have presented an autonomous car platform based on the softmax function squashes the outputs of each unit to be between 0 and 1, similar to a sigmoid function. The softmax function acts as a sigmoid function by ranging the output while an actual softmax function does not do that. Using a neural network helps in giving real time output. They have implemented the model on the MATLAB simulator before actually implementing it. The system uses only one single camera for all inputs and it drives at about 5-6 km/hr whether the lane markings are present or not. This model only detects lane markings and turn signs. It just hovers the car left or right and does not sense signals or stop sign.

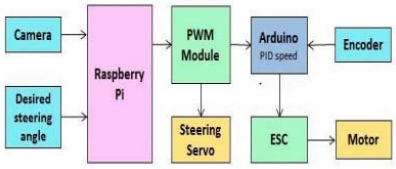

Fig. 1 . Block diagram of RC self-driving car platform

In literature [2], they have built an autonomous RC Car that uses Artificial Neural Network (ANN) for control. It explains the thesis behind the neural network and autonomous vehicles. Using L298N IC and motor driver a car is made which can be managed by a micro controller and then in return sent to the model car. Using CNN helps in only detection of gray scale parts and it ignores the unnecessary data in the detection. The system use is very limited but accurate. Using an embedded pi camera for input and gray scale of images for training in neural networks. The system detects lane markings for each direction and does not offer any other functionality other than that.

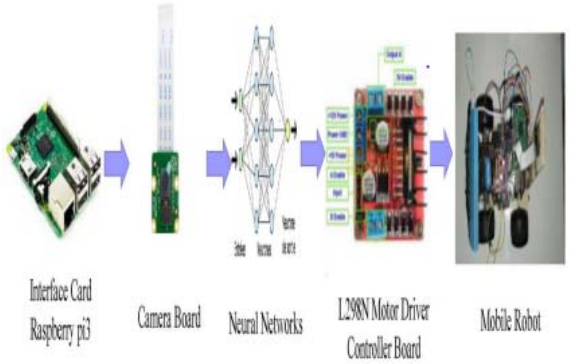

Fig. 2 . Block diagram of the system

In literature [3], in this paper, they have presented a reinforcement-learning based approach with Deep Q Network implemented in autonomous driving. Using lidar sensors detects objects at a very far distance. The whole system is developed on a simulator depicting actual roads and city streets with traffic. Using fusion of camera and lidar helps in better knowing of the surroundings and all kinds of obstacles. They have implemented a model using lidar(laser sensor) which is a very costly sensor and it is applicable for large scale cars.

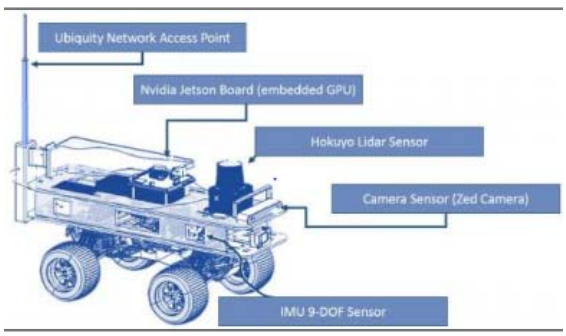

Fig. 3 . The car prototype for the Real-Time DQN implementation

In literature [4], the issue talked about right now object detection utilizing deep neural systems, particularly convolution neural systems. Object detection was before done utilizing just traditional profound convolutional neural systems while utilizing a regional based convolution arrangement upgrades the exactness and furthermore diminishes the time required to finish the program. Training a neural system without any preparation takes additional time and handling power as it is extremely hard to discover the dataset of adequate size and ground truth. Using Regional Convolutional Neural Network (RCNN) helps in finding appropriate regions in image and it enables the system to give real time outputs. This deep neural network is used for image processing, mainly for medical uses like tumor and such where the data set is too complex to detect regions in comparison to a model road environment.

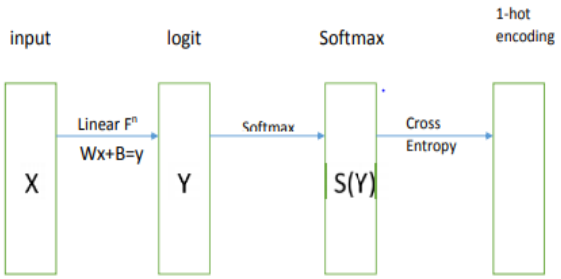

Fig. 4 . Multinomial logistic classification

In literature [5], their proposed model takes a picture with the assistance of Pi cam appended with Raspberry Pi on the vehicle. The Raspberry-Pi and the PC are associated with a similar system, the Raspberry Pi sends the picture caught which fills in as the information picture to the Convolutional Neural Network. The picture is changed over to grayscale before passing it to the Neural Network. Upon expectation the model gives one of the four yields that are left, right, forward or stop. At the point when the outcome is anticipated the related Arduino signal is made high or activated which thus encourages the vehicle to move a specific way with the assistance of its controller. Their vehicle was prepared under various mixes of the track for example straight, bended, blend of straight and bended and so forth. Aggregate of 24 recordings were recorded out of which pictures were extracted.10868 pictures were extricated and were completely set in various envelopes like left, right, straight and stop. Right now, a technique to make a model of self-driving vehicle is introduced. The diverse equipment segments alongside programming and neural system design are obviously depicted. With the guide of Image Processing and Machine Learning a fruitful model was created which filled in according to desire. Along these lines the model was effectively structured, actualized and tried. The vehicle marginally moves out of the track which can be a difficult issue in the event that it hits close by objects in the event that we think about a genuine vehicle.

In literature [6], they have developed a car which is prepared for three different street signs : Stop, No left, and Traffic light utilizing DQN with a current CNN model. These street signs are distinguished in the earth utilizing OpenCV cascade classifiers .Supervised learning was finished utilizing Convolutional Neural Network and we accomplished 73% test and 89% train accuracy. Notwithstanding this they prepared the vehicle with fortification learning for three diverse sign boards; Stop, No left and Traffic light utilizing a deep Q-learning system with CNN model. Right now, the preparation is done in an obliged domain, numerous

978-1-7281-6509-7/20/$31.00 ©2020 IEEE

components of the genuine condition can influence the forecast of the model. Helping impacts and climate conditions impacts the pictures they get from cameras which can greatly affect model prediction.

Fig. 5 . Sign detection with OpenCV cascade classifiers

III. Problem Statement

With the highly rising traffic congestion , human drivers errors due to several reasons, we have seen an increasing number of accidents and inconvenience on the road. With hardcoded rules autonomous cars are very unlikely to make blunders like human drivers. Unlike human drivers, autonomous cars will be able to operate with the same efficiency and decision making under any situation. Hence, to reduce traffic congestion, front collision, provide relaxed travelling experience, provide increased safety and better observe traffic rules, autonomous cars prove to be an ideal solution.

As mentioned above there are many different algorithms, hardwares and softwares used to implement one or more features of automation individually. So our aim is to pick the best suited hardwares and algorithms to address automation. In our project we aim to depict how automation will work in an actual car by creating a prototype made up of 1/10 scale rc car. With our design we desire to achieve autonomous driving through lane detection, traffic rules observation through signal detection and stop sign detection and implement front collision avoidance in an efficient and cost effective manner.

IV. Proposed System

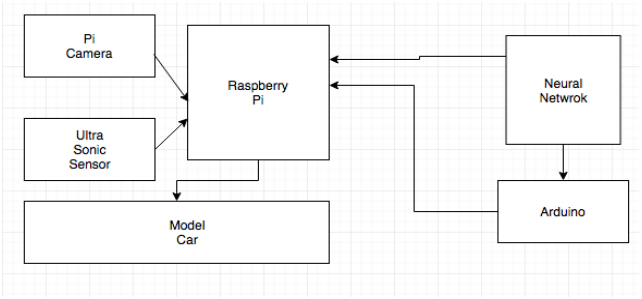

Fig. 6 . Block Diagram of the Proposed System

The model prototype which is developed aims to implement automation by handling tasks such as self-driving through lane line detection, stop sign and traffic signal detection and fore collision avoidance. The system design for implementing the same will consist of three units first is input unit containing a pi camera and ultrasonic sensor, the second is processing unit which is our laptop that will act as a server, the neural network will be running over here and third, is RC control unit which is Arduino. Firstly, in the input unit, the raspberry pi board of the B+ model will be connected with the raspberry pi camera and an ultrasonic sensor to stream input data[9]. There will be two client programs running on raspberry pi one to stream images collected by the pi camera and another to send sensor data through a local wi-fi connection.

Then the processing handles more than one tasks such as collecting data from raspberry pi, neural network training and steering prediction, stop sign and signal detection, distance measurement using monocular vision and sending commands to Arduino through a USB connection. A multithreaded TCP server program will execute on the raspberry pi to receive image frames and sensor data. The image frames will be converted to grayscale and decoded into NumPy arrays. The neural network which is the convolutional neural network will be trained to make steering predictions based on detected lane markings. The lower half from the input image will be used for training and prediction purposes. There will be input, hidden and output layers, there will be four nodes in the output layer each corresponding to steering control instructions that are left, right, forward and reverse. The training data can be collected or the datasets already available can be used. For training, each frame will be cropped and transformed to a NumPy array. Then the train image will be paired with the train label. Then all the paired image data and labels will be stored in the npz file. OpenCV will be used to train the neural network. After training the weights will be stored in an XML file and for generating predictions the same neural network will be created and loaded with the trained XML file. For signal and stop sign detection that is part of object detection will be done using a shape-based approach; Haar feature-based cascade classifier will be used for object detection. Since every object requires its own classifier Haar cascade is used.

The maximum speed we are aiming for is 10-15 kilometres per hour due to the size of the model. Monocular vision will be used for distance measurement for object detection, it will be helpful in determining at what distance from the object the car should take hold. Also, it will be more appropriate for the design as the raspberry pi has only single pi camera slot and utilizing a USB web-camera will also increase weight to the car.The braking distance between the car and obstacle is hardcoded and set to 5.5 centimetres and the threshold for stopping is 30 centimetres. The RC car used in our prototype has an on/off or high/low switch type controller. Therefore, the Arduino board will be used to imitate button press actions. There will be four Arduino pins used to connect four chip pins on the rc remote controller, corresponding to forward, reverse, left and right actions respectively. The Arduino will be connected to the computer through USB and the computer will send the output commands and write out low or high signals, simulating button press actions to drive the car autonomously.

978-1-7281-6509-7/20/$31.00 ©2020 IEEE

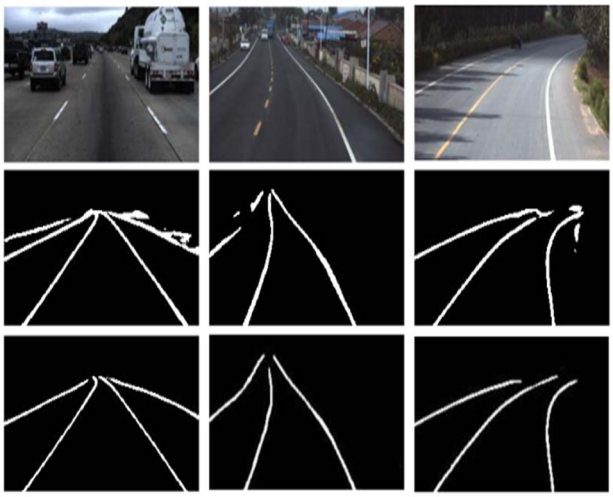

Fig. 7 . Lane line detection using OpenCV

V. FLOWCHART

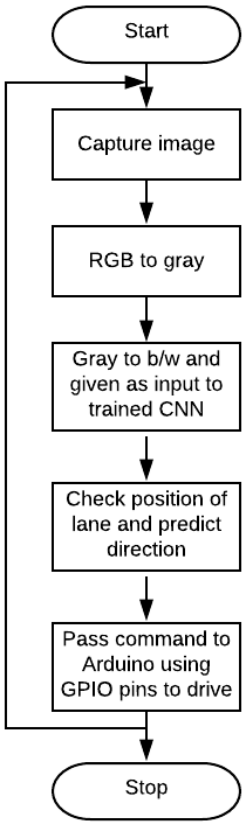

Fig. 8. Flowchart for Lane detection

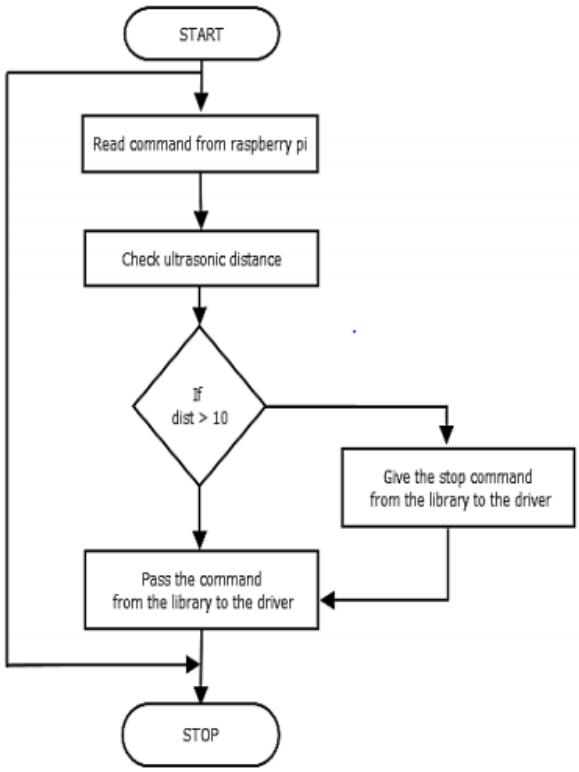

Fig. 9 . Flowchart for Lane detection

VI. CNN

The dataset used for detection of safety signs and signals is COCO. The COCO dataset is used with a model of MobileNet v2 COCO Quantized model. Quantization helps in making model interferences run faster by storing the model parameters not as double values but integral values, which has very low effect on accuracy.

There are numerous layers, for example, a convolutional layer, pooling layer, non-linearity layer, and completely associated layer. The convolutional and completely associated layers contain parameters however pooling and non-linearity layers don't contain parameters. CNN has extraordinary execution in AI issues. Particularly the applications that manage picture information, for example, the biggest picture grouping informational collection (Image Net), computer vision, and natural language preparing (NLP) and the outcomes accomplished were astounding. Consequently recognizing its fruitful application in different territories just as thinking that its appropriate to our application to give precise outcomes for an enormous number of information pictures and performing better over different models, CNN is the better alternative for path line identification and forecast of development.

VII. CONCLUSION

Automation in cars in real world is a very big field, where many sensors come into action, our idea is to solve many of those into two sensors by detecting distance and objects like stop sign, signals and other obstacles with a single method of monocular vision which can be enhanced in future from a scaled car to an actual car. The prototype focuses on these functionalities which we are developing in a model rc car while others focus on only one aspect of it.

978-1-7281-6509-7/20/$31.00 ©2020 IEEE

REFERENCES

[1] Truong-Dong Do, Minh-Thien Duong, Quoc-Vu Dang and My-Ha Le, "Real Time Self-Driving Car Navigation Using Deep Neural Network" in International Conference on Green Technology and Sustainable Development (GTSD), 2018

[2] Hajer Omrane, Mohamed Slim Masmoudi and Mohamed Masmoudi, "Neural controller of autonomous driving mobile robot by an embedded camera" in International Conference on Advanced Technologies For Signal and Image Processing - ATSIP, 2018

[3] Abdur R. Fayjie, Sabir Hossain, Doukhi Oualid, and Deok-Jin Lee, "Driverless Car: Autonomous Driving Using Deep Reinforcement Learning In Urban Environment" in 15th International Conference on Ubiquitous Robots (UR) Hawaii Convention Center, Hawai'i, USA, June 27-30, 2018

[4] F. Shumaila Mateenuddin, Mrs. V.S. Jahagirdar, "An Android Controlled Mini Rover for real time surveillance using Raspberry Pi 3", in International Journal Of Advanced Research in Engineering & Management (IJAREM), 2017.

[5] Malay Shah, Prof, Rupal Kapdi, "Object Detection Using Deep Neural Networks", in International Conference on Intelligent Computing and Control Systems ICICCS 2017

[6] Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 580-587

[7] Aditya Kumar Jain, "Working model of Self-driving car using Convolutional Neural Network, Raspberry Pi and Arduino", in Proceedings of the 2nd International conference on Electronics, Communication and Aerospace Technology (ICECA 2018) IEEE Conference Record # 42487; IEEE Xplore ISBN:978-1-5386-0965-1

[8] Syed Owais Ali Chishti, Sana Riaz, Muhammad Bilal Zaib, Mohammad Nauman, "Self-driving Cars Using CNN and Q-learning", in 2018 IEEE 21st International Multi-Topic Conference (INMIC)

[9] U. Karni, S. S. Ramachandran, K. Sivaraman and A. K. Veeraraghavan, "Development Of Autonomous Downscaled Model Car Using Neural Networks And Machine Learning," 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2019, pp. 1089-1094

978-1-7281-6509-7/20/$31.00 ©2020 IEEE