

# Self-Driving Car Using Raspberry Pi and Arduino with Neural Networks

Sufyan Parkar<sup>1</sup>, Chaitanya Dandekar<sup>2</sup>, Sushil Hambir<sup>3</sup>, Shoeb Shaikh<sup>4</sup>

<sup>1</sup>Department of Electronics and Telecommunication Engineering, Mumbai University, India  
Email: sufyanparkar@gmail.com

<sup>2</sup>Department of Electronics and Telecommunication Engineering, Mumbai University, India  
Email: chaitanya.s.dandekar@gmail.com

<sup>3</sup>Department of Electronics and Telecommunication Engineering, Mumbai University, India  
Email: hambirsushil248@gmail.com

<sup>4</sup>Department of Electronics and Telecommunication Engineering, Mumbai University, India  
Email: shoeb.sk.028@gmail.com

**Abstract**— Today technology has advanced to an extent that it would seem science fiction in the past. Self-driving has been a hot topic since the concept of AI has emerged, because of the complexity of the problem, seemingly endless applications, and capital gain. Although this concept has been looked at and many solutions came across, they weren't accurate. The accident-scenarios Self-driving cars might face are the key examples. The main purpose of this study is to find the issues and suggest possible solutions regarding the implementation of Self-driving cars on roads whilst keeping in mind the Accident avoiding algorithms. A Self-driving car is fairly in its infancy today, but once implemented this can revolutionize our traffic management and transport system, moreover it'll reduce the number of accidents and also aid the disabled to drive. This paper tries to solve them using some basic electronics. A Raspberry-Pi along with sensors and a camera are used to solve the problem. A path will be given to the car so that it will traverse along the path while detecting obstacles, responding to signals thus following traffic rules and reach its destination with zero or no casualties. OpenCV along with Haar cascade classifier is used for object detection and classification of pre-trained images and necessary actions will be taken to execute them.

**Keywords**—Autonomous car, Haar Cascade, Neural Networks, OpenCV, pygame, traffic congestion.

## I. INTRODUCTION

### 1.1 PROBLEM STATEMENT

There has been an insurmountable rise in the amounts of accidents taking place daily, and the problem of traffic congestion has been badly overlooked in many cities and urban centers.

### 1.2 OVERVIEW

There seems to be a monumental waste of time, energy and human potential due to the fact that traffic congestion has increased drastically, also the number of accidents taking place has peaked. So, there has to be a solution that needs to be designed in order to deal with this issue, whilst taking into consideration the challenges that could be faced by that system, also the ethical dilemma that could arise due to controversial or maybe anodyne discussions among the experts. For many years, the solution to this problem was simple, build new roads or expand existing ones. It worked surprisingly well for a few cities when hundreds of historical buildings were torn down to create transportation-friendly boulevards. But in urban cities, there is no space for any significant further expansion of the network, since the habitat is too dense, real estate too expensive and public finances too fragile. Other major reasons for traffic congestion could be the none availability of parking space, which leads the drivers parking their cars along the side of the roads, thus rendering less driving space.

As we progress towards a fully autonomous future, technology becomes more and more advanced and mature. What was once seen as Science-fiction, is now becoming a reality. As technology advanced, the concept of Driverless or autonomous cars surfaced. Driverless vehicles have the potential to revolutionize transportation networks [12] and radically transform urban design strategies. This concept of Driverless cars [9] has existed since the days of automobile development, this [13] idea of Self

Driving cars can be thought of as another force in the nature that has the potential to make the urban traffic mesmerizing. If we think about it, the current generation of driverless cars is just trying to earn its way into a traffic grid made by and for humans. They're trying to learn traffic rules, which is relatively simple, and are trying to cope up with human unpredictability, which is more challenging. Due to software and algorithmic innovation, there have been advances in machine learning that improve the ability to perceive the world, new tracking and planning algorithms allow for better, safer and smoother driving, and the software structure simulates and analyses vast amounts of data in data centers, have all been key contributors towards making self-driving cars a reality. Although software and hardware advances have also played a critical role, they are being implemented, still, there hasn't been any up-gradation in terms of intelligence besides the quality of components used in them. Self-driving cars must be fully ready to perform safely and reliably for use in real life. Applying the technologies based on simulation directly on the real world makes it difficult to run vehicles itself in an actual road environment.

Keeping in mind the impact it will have on our day-to-day lives, it'll drastically reduce the number of accidents, as humans being termed as monkey drivers with slow reaction time and short attention spans fail to react quickly to any certain situation. Traffic congestion like city-sized gridlock cascades happen due to uncoordinated driving at the intersection. Even on highways with no intersections, due to unplanned braking or slowing down of certain vehicles, it renders the formation of a phantom intersection on the highway. Only if the drivers were able to coordinate properly and drive with equal speeds and accelerations, easy driving could be achieved. On multi-lane highways, the cars that cross through lanes cause gridlocks, which again results in congestion that could last hours. Tailgating a term used by experts refers to the drivers that drive too close to cars in front of them, the drivers should have a rule of maintaining the same distance from the car ahead as from the car behind at all times.

Though wishing that everyone lives in a proper manner and that would make the world a better place is a terrible idea, what we need is a Structurally Systematized Solution, which is what exactly self-driving cars are, having an interconnected network of cars, it'll be possible to achieve a vast increase in the throughput. Also, in cities such as Mumbai, Bangalore, Delhi and Chennai, traffic congestion is an everyday scenario. With non-standard roads, lack of proper parking facilities and a broken traffic system, cities like these are a nightmare for self-driving car's progress. Despite these major concerns, it would be foolish to reap the benefits of this inevitable technology. While the respective authorities are taking measures like building flyovers, more roads, the issue is still likely to be persistent in the future given the increasing population of cars and humans. To make an autonomous car at low cost in a very high-risk environment is going to be challenging. In this project, we try to implement such a car using Raspberry-Pi 4B, an Arduino UNO Board with some sensors in a limited price range and yet try to yield the safest and most efficacious outcome possible.

## II. PREVIOUS WORK

Johann Borenstein and Yoram Koren, designed and built a mobile robot system, that can accomplish various tasks for the physically disabled. It is basically a nursing robot that acts as a constant 24\*7 support to a bedridden individual, it has few ultrasonic range finder sensors attached to it, so as to avoid colliding with the obstacles like table, wall, any medical unit etc. Ivan Culjak, David Abramet al [1] introduced and made a reader familiar with OpenCV (Open Source Computer Vision) basics without having to read through the lengthy reference manuals and books. OpenCV is an open-source library for image and video analysis, originally introduced by Intel. Sander Soo [2] focused purely on what object detection is and how the Haar-cascade classifier is used for object classification and recognition. The further parts highlighted the actual work carried out by the researchers to detect and count cars, with the use of advanced background algorithms and various filtering techniques the cars were detected with the least error. Jeremy Straub [3] published a paper that focuses on the testing of a self-driving car through a computer simulation. The performance of self-driving vehicles is evaluated using the simulation testing system that generates random scenarios under randomly, manually and genetically modified algorithms. The efficiency of this testing approach is taken into consideration from numerous perspectives. Finally, the paper discusses the technical difficulties and the important role of automated testing process and how automated testing fits within the roadmap to a wide-scale technology application.

Gurjashan Singh, Mohammad Dawud et al. [4] presented a driverless car that is built using Raspberry Pi board as the processing chip, an HD camera along with an Ultrasonic sensor that is used to provide the car with the visual and distance measurement

information of the vicinity of the vehicle, and the objects around it. This autonomous car is designed in a manner that is capable of reaching the given destination with no casualties or any errors using the implementation of algorithms like lane detection and obstacle avoidance. Aditya Kumar Jain [6] proposed a working model of self driving car capable of driving from one location to another also on different terrains which may be straight or curved. A camera was mounted on top along with the raspberry pi on the car which sends images from the real world to the Convolutional Neural Network which predicts one result. This result is then sent to the controller via the Arduino and as a result the car moves in the direction desired with any human interference. [5] T. Banerjee, S. Bose et al. published a paper in which a unique controller design of a driverless car is described that is powered by green energy, has collision protection algorithms and sensors and uses GSM guided destination assistance, and a GPS module to accurately track the location, to map its source and destination. Also, an audio-based navigation guidance was provided. Sana Urooj, Irum Feroz et al. presents an overview of a systematic literature survey conducted on research reports that concerns self-driving cars or autonomous cars based on different perspectives like decision making and its ethics, responsibilities of these cars, potential drawbacks or harm that could be possessed to the passenger by using self-driving cars, people's perspective about them and challenges faced by them. And thus, they present their own views and ideas on it and also suggest some ways or a different approach at this issue.

### III. MATERIAL AND METHOD

#### 3.1 COMPONENTS USED

##### 3.1.1 HARDWARE

The hardware part includes Raspberry Pi, an Arduino UNO, BO Motors, L293D motor driver IC and a Camera Module. The raspberry pi can be termed as a mini computer that can be plugged into a computer monitor or a TV, it has onboard RAM and ROM with built-in Wi-Fi module, and can be programmed using languages like Python, R and Scratch. An Arduino Uno is a development board that has an 8bit microcontroller ATmega328P and a flash memory of 32K. Even this board can be programmed to carry out certain tasks. The BO motors are simple voltage driven DC motors capable of handling 500gm force/cm torque, and are used as driving units for the prototype car, it has a speed of 200 rpm. The L293D is a motor driver IC it is also referred to as a Dual H bridge IC that has 16 pins and is designed to provide DC voltages to two motors at a time. The Camera module used here is a USB webcam or a Pi-Camera, this is the main data capturing unit of the entire architecture, that sends the captured/recorded images and video feed to the system for further processing. An Arena was built as a replica of the human city that has roads, intersections, buildings, traffic lights, other vehicles, etc.

##### 3.1.2 SOFTWARE

###### 3.1.2.1 NEURAL NETWORKS

Neural networks [7][10] are modelled loosely on the human brain, it is a means of doing machine learning, wherein a computer perform tasks by learning from the given dataset of the training examples. It is a series of algorithms that aim to recognize fundamental relationships within the dataset and it tries to mimic the way a human brain operates. Most of the neural networks (Fig 1) are organized in layers and are feed-forwards type where it sends data to other nodes in only one direction. Neural networks are designed to be able to adapt to changing input; so, the network can generate the best possible result without needing to redesign the output criteria.

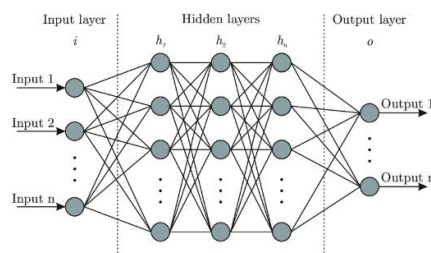


FIGURE1:Neural Network

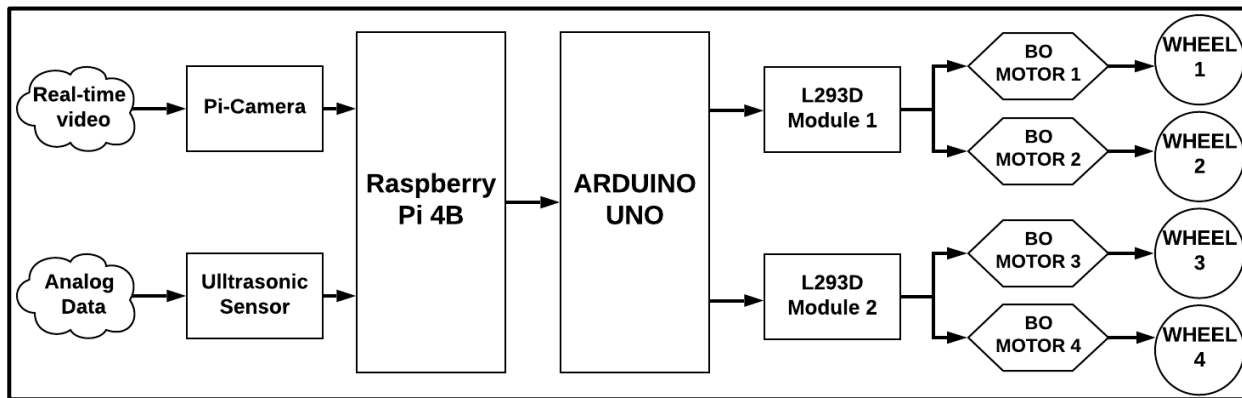
###### 3.1.2.2 OBJECT DETECTION USING HAAR CASCADE CLASSIFIER

A Haar feature based cascade classifier was fed with an image dataset to train the model, in order for it to be capable enough to identify the given image and categorize it accordingly. Haar cascade classifier [12] uses several feature extraction techniques to significantly distinguish an object within few seconds, its accuracy is totally dependent on the amount of training dataset and the hyperparameter tuning.

### 3.1.2.3 OBSTACLE AVOIDANCE USING ULTRASONIC SENSOR AND ARDUINO

An ultrasonic range finding sensor is used to avoid collision of the car with the obstacles in its path. An Arduino code written in Arduino IDE is flashed on the Arduino UNO board that helps facilitate this task of obstacle avoidance and safe navigation[11].

## 3.2 PROPOSED METHODOLOGY



**FIGURE2: Block Diagram**

We tried to achieve the simplest design possible. Four BO motors were attached to the chassis. The BO Motors will be connected to IC L293D that is a Motor driver IC which will provide the motors with the necessary supply needed. Also, the Arduino is connected to the raspberry Pi via the serial Port. The raspberry Pi will be mounted on the top along with the Arduino. The car's BO Motors will be connected with the Arduino. The Pi-Camera will be connected to the Raspberry Pi. Lastly ultrasonic sensor is connected to the Raspberry Pi.

### 3.2.1 TRAINING

The car was first trained on the premade arena and then tested on the arena. The Raspberry pi was loaded with the program to train the car. Here the Raspberry Pi and Arduino weren't connected with each other. Arduino was connected with the PC and was flashed with the cars control program, this program had instructions to drive the car. Now the computer ran the program designed to control the car with the keyboard using pygame [8] library. The raspberry pi connected with pi camera took images as the keyboard buttons were pressed, these images were sent on the local machine where a server was built using socket library, the Raspberry pi acted as a client. When enough data was collected, the data was given to the neural network to train. The data collected was typically optimized for the set of feature networks of the respective arena. The Pi Camera calibration was done to calibrate the pi camera so that it works at its best condition, multiple chess board images were taken using the camera at various angles and put them into the folder. Then, a program to calibrate the pi camera was run. It returned the camera matrix which was to be entered into driving program. This matrix was used for distance measurement by the car while it's self-driving and for efficient steering and navigation.

### 3.2.2 TESTING

When the model's training was finished, it was tested on an arena. While testing, the raspberry pi will be connected to the Arduino via the serial port and rest of the connection will be the same. Now Arduino will be flashed with the program to control the car and raspberry pi will execute the code to run the model. When the Neural Network model is called, it will return a predicted output of the path. This output will be sent to the Arduino via the serial port. Then with the functions built

for every movement for effective navigation, the Arduino will react to it accordingly. After observing its driving accuracy, stop sign and traffic light detection accuracy, tweaks to be made were noted.

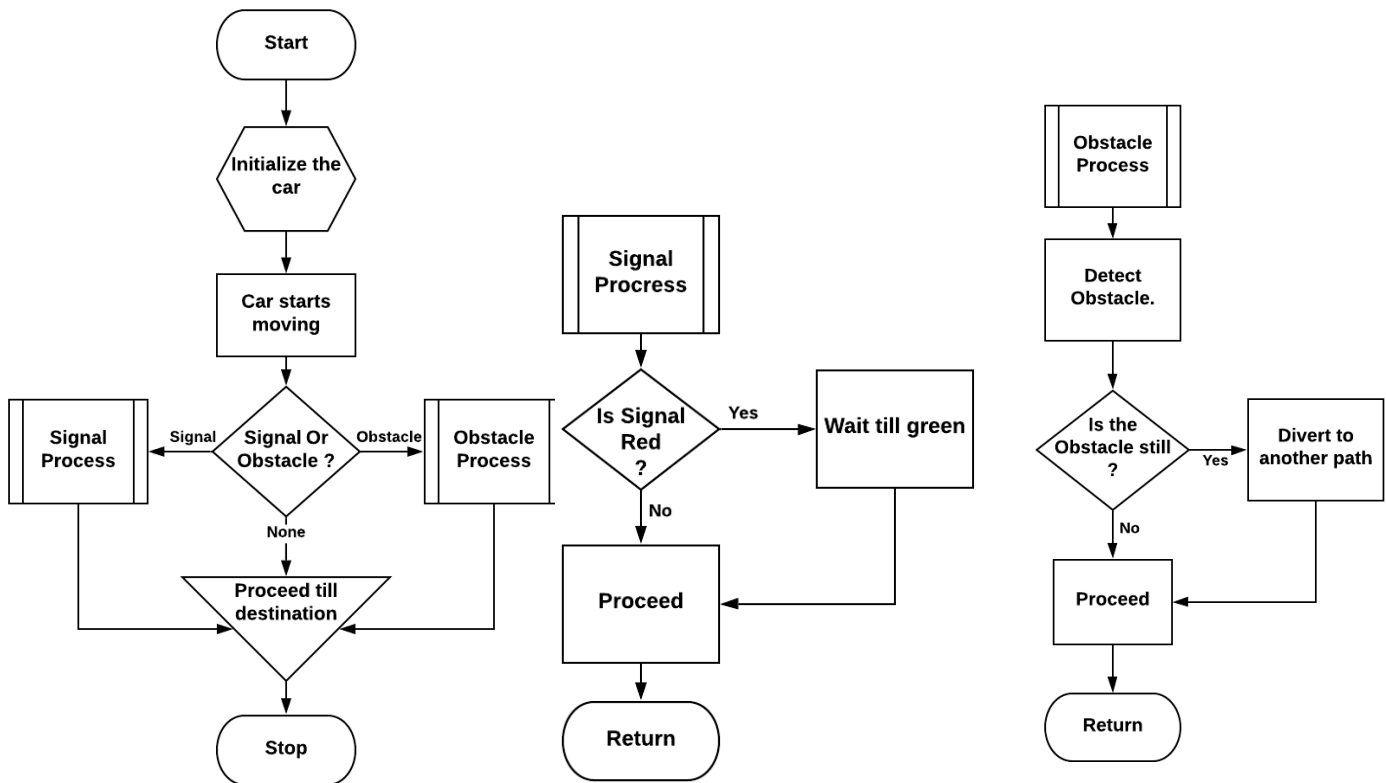


FIGURE3:Flowchart of the entire system

#### IV. RESULTS AND DISCUSSIONS

We present a partial result obtained by performing for the self-driving car project. A stop sign detection algorithm was executed and the obtained snippet (Figure 4) shows the detection of the stop sign. Figure 5 shows the chassis built from various components. The components used are 4 BO Motors connected to two L293D Motor driver modules, two each, and BO Wheels connected to BO Motors. Here power is given through two 9V batteries connected in parallel to the Motor driver modules. The modules are connected to the Raspberry Pi 4. The cars then operate according to as it is programmed.



FIGURE4: STOP Sign Detection

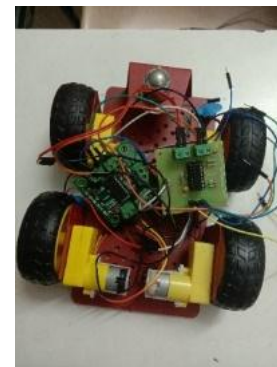


FIGURE5: Chassis built Car

## V. CONCLUSION

In the proposed approach, a method to make a self-driving robot car is presented with different hardware components and their assembly are clearly described. A novel method to determine the uneven, marked or unmarked road edges including some objects is explained in detail relying upon neural networks and computer vision. Using ultrasonic sensor, distance between other vehicles or obstacles is derived. The algorithm proposed has been successfully implemented on a small autonomous car. The autonomous car would surely prove out to be a boon in the automation industry and would be preferred over many traditional techniques. As they won't require any drivers, accidents caused by the carelessness of the drivers carrying goods, would be reduced and would ensure better logistic flow. The human driver has a lesser response time and fails in cognitive decision making at times of chaos and disorder, the self-driving cars won't face such problems. If the cities become totally driverless, we won't even need traffic rule lanes or speed limits. And when all the cars are driverless and connected, everything will be predictable with quick reflexes. They can take any rational initiative that can speed them up or the cars around them. So instead of rigid traffic rules, flow will be regulated by a mesh of dynamic and constantly self-improving algorithms. Hence, due to greater autonomous nature and efficiency, an autonomous car of this nature can be practical and is highly beneficial for better regulation in the goods and people mover's section.

## REFERENCES

- [1] I. Culjak, et al. A brief introduction to OpenCV Proceedings of the 35th International MIPRO Convention, IEEE (2013), pp. 2142-2147
- [2] S. Soo, "Object Detection Using Haar-Cascade Classifier", Institute of Computer Science, University of Tartu, 2014.
- [3] Straub, J. Automated Testing of A Self-Driving Vehicle System. In Proceedings of the 2017 IEEE AUTOTESTCON, Schaumburg, IL, USA, 9–15 September 2017; pp. 1–6.
- [4] Singh Pannu, G., Dawud Ansari, M., and Gupta, P., "Design and implementation of autonomous car using Raspberry Pi," International Journal of Computer Applications 113(9), 22–29 (2015).
- [5] Urooj, S., et al. Systematic literature review on user interfaces of autonomous cars: Liabilities and responsibilities. 2018 International Conference on Advancements in Computational Sciences (ICACS), 2018
- [6] K. J. Aditya, "Working model of self-driving car using Convolutional Neural Network, Raspberry Pi and Arduino", in 2018 Second International Conference on Electronics, Communication and Aerospace Technology, IEEE, 2018, 1630–1635.
- [7] J. J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities", *Proc. Natl. Acad. Sci. USA*, vol. 79, pp. 2554-2558, April 1982.
- [8] S. Kelly, "Basic introduction to pygame," in Python, PyGame and Raspberry Pi Game Development. Springer, 2016, pp. 59–65.
- [9] Thrun, S. Toward robotic cars. *Commun. ACM*, 2010, 53(4), 99–106.
- [10] J. Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61: 85-117, 2015.
- [11] Bhagat, K.; Deshmukh, S.; Dhonde, S.; Ghag, S. Obstacle Avoidance Robot. *Int. J. Sci. Eng. Technol. Res.* 2016, 5, 439–442.
- [12] Y. Y. Qin, H. Wang, B. Ran, "Impact of connected and automated vehicles on passenger comfort of traffic flow with vehicle-to-vehicle communications", *KSCE J. Civil Eng.*, vol. 23, no. 2, pp. 821-832, Feb. 2019.
- [13] Á. Takács, I. J. Rudas, D. Bösl, T. Haidegger, "Highly automated vehicles and self-driving cars", *IEEE Robot. Autom. Mag.*, vol. 25, no. 4, pp. 106-112, Dec. 2018.