International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 3 March 2023, pp: 961-971 www.ijaem.net ISSN: 2395-5252

Image Processing Technique for Self-Driving Car

Dr. Jayanti Mehra Jharia[1], Dev kumar Chouhan[2]
Lakshmi Narain College of Technology (LNCT), Bhopal, India [1] [2]

Date of Submission: 09-03-2023 Date of Acceptance: 18-03-2023

ABSTRACT— The number of accidents and traffic congestion is at an all-time high and the need for self-driving cars is now more than ever. Safety, comfort, and convenience are the most sought features in the transportation system and automation plays a very important role in providing all three factors. Artificial intelligence, machine learning, and image processing can be used to achieve this automation. In this work, a model was developed to follow the lane and improve the automation tasks in the car that can avoid most accidents and save lives. A car with chassis was built and interfaced with the motor driver to move the vehicle. The Raspberry pi camera is the image sensor that takes the input and provides it to the Raspberry pi processor which is further processed with algorithms. The machine learning algorithm was implemented on Raspberry Pi, an open-source platform.

Artificial intelligence, currently the most popular technology, was used to create the model. Some of the pre-processing techniques include image processing. Conversion of the colored pictures into grayscale and image processing techniques to detect the lane. The camera takes input and converts it into grayscale and the neural networks use a color detection algorithm to detect the color. The color detection algorithm processes the path and predicts the steering angle to move in that direction. Hence the motor driver operates on the steering command received and moves in that direction.

Keywords: Artificial Intelligence; Machine Learning; Image Processing; Neural Networks; Data Visualisation

I. INTRODUCTION

Nearly all automotive businesses are developing autopilot automobiles on today's trendy planet. In a variety of high-end cars, human-driven cars use technologies to ensure safety, identify obstructions, and trigger auto stops, but none of them are fully autonomous. The automation capability necessary for cars to drive independently is not present in the current generation of automobiles. Drivers are always needed since, without them, the car can be operated. By upgrading them to become self-driving vehicles, cars can be made constantly available. In conventional cars, the driver must always be aware of the signals, traffic safety signs, obstructions, and lane markings and must make decisions in accordance with those observations. Self-driving vehicles could be used to circumvent this as well.

Self-driving cars are rapidly becoming a reality and are no longer just science fiction. Companies announce their commitment to creating and introducing autonomous vehicles every week, and many of them discuss the "degree" of autonomy being created. Although autonomous driving has advantages, it might also be unsettling to certain people. It will lead to less traffic congestion, less pollution, lower transportation expenses for everyone, and lower costs for building new infrastructure and roads. Additionally, it would significantly improve the mobility of elderly and physically challenged people.

The areas being studied involve self-driving model automobiles in a model environment with few sensors, similar to how Tesla uses several sensors like radarand lidar. By simulating 1/10-size radio-controlled cars, we intend to achieve car automation. With the help of a pi camera and an ultrasonic sensor, the car will be able to sense its surroundings. Data from both sensors is streamed to a server via a raspberry pi, where a neural network will process the images to detect lane markings. A haar cascade classifier will also be used to detect stop signs and traffic signals. Sensor data will also be processed to prevent front collisions by braking the car when it detects an obstacle at a predetermined distance. Once trained, the automobile will autonomously follow lane markings and make navigating decisions by itself.

Automation can be implemented in actual automobiles using the same methodology and

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 3 March 2023, pp: 961-971 www.ijaem.net ISSN: 2395-5252

methods. We have noticed an increase in accidents and discomfort on the roads due to the highly increasing traffic congestion and human drivers' faults for various reasons.

Hardcoded regulations make it highly uncommon for autonomous vehicles to make mistakes similar to human drivers. Unlike human drivers, autonomous vehicles can respond to situations with the same quickness and judgment.

Consequently, to lessen traffic congestion, and front collisions, make commuting more enjoyable, promote safety, and to better adhere to traffic laws.

II. RELATED WORK

Convolution neural networks were used in the creation of an autonomous vehicle by Mariusz Bojarski et al. [1]. The network has been taught to immediately transfer front-facing camera raw pixels to steering commands. When tested on the road at a speed of 10 mph with no intercepts, this approach was 98 percent successful. It was 90% successful in the simulation.

A self-driving car model that functions and navigates using its intelligence was proposed by Hiral Thadeshwar et al. [2]. The paper's central concept is the creation of a 1/10-size radio-controlled car to represent an automated vehicle. This solution addresses the issue by sensing distance and objects, such as stop signs, signals, and other obstructions, using just two sensors.

Two approaches were suggested by Syed Owais Ali Chishti et al. [3], supervised learning: 3,000+ data points were gathered while the car was being driven on a road or surroundings. This led to the training of a CNN model, which had a 73 percent test accuracy and an 89 percent train accuracy. Reward-based learning Using DQN and the pre-existing CNN model, the car is taught for three separate road signs: Stop, No left, and Traffic light. OpenCV cascade classifiers are used to find these road signs in the surrounding area.

A self-driving car model was given by Wuttichai Vijitkunsawat et al. [4] employing the three machine learning algorithms SVM, ANN-MLP, and CNN-LSTM in three-speed levels to examine behaviors. The model is more accurate at higher speeds but less efficient at lower speeds.

A self-driving car prototype based on monocular vision and employing a Deep Neural Network on a Raspberry Pi was proposed by Truong-Dong Do et al. [5]. The experimental findings show how well the autopilot model performs and how reliable it is when performing a lane-keeping duty. Regardless of the presence or absence of lane markings, the vehicle's top speed is approximately 5–6 km/h in a range of driving situations.

Pride Cai and others[6], In this research, end-to-end deep learning systems for autonomous driving and racing have demonstrated encouraging outcomes. This approach is used on a real-world 1/20-scale RC car with minimal onboard processing power as well as in a high-fidelity driving simulation. The evaluation findings show that, in terms of sample efficiency and task performance, our method performs better than earlier IL and RL methods.

III. DESIGN AND IMPLEMENTATION

The execution of the model and the design of the electronic circuitry are both covered in this chapter.

Block Diagram and Working

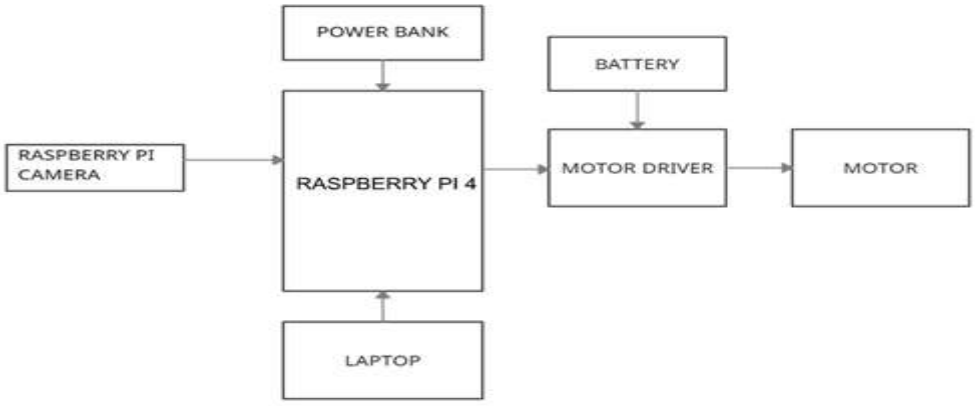

Fig. 3.1. Block Diagram

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 3 March 2023, pp: 961-971 www.ijaem.net ISSN: 2395-5252

The block diagram for the RC-Car model is shown in figure 1 above. Raspberry Pi is controlled by a laptop and is fuelled by a power bank. Motor drivers are included to supply the necessary current needed for the motor to move the automobile. Raspberry pi receives the input from the Raspberry Pi camera, processes it, and drives the motor based on the processed output from the Raspberry Pi. The battery provides constant power to motor drivers.

Circuit Diagram

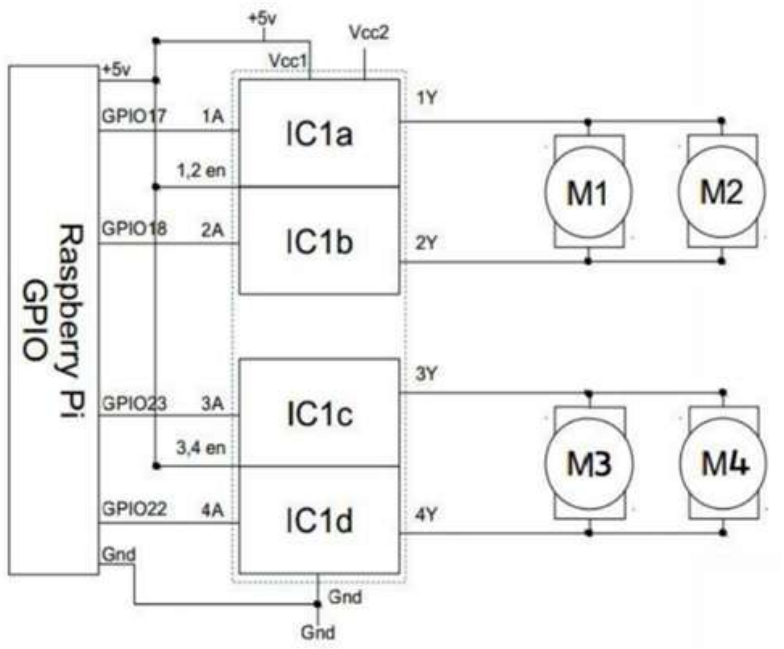

Fig. 3.2. Circuit diagram of Self-driving RC-Car

In the above image, various components are connected. Raspberry pi takes the input from the Raspberry pi camera And forwards it to the controller for further processing. The Raspberry Pi controller processes the received data based on the different algorithms, the algorithms predict the steering angle. Then generates a PWM signal to drive the motor. As current is not sufficient to drive the motor, a motor driver is used to drive the motor.

Lane detection - Image processing

There are two crucial processes involved in lane detection.

- Pre-processing of the image
- Neural networks

Finding Lane

The concept is to use edge detection or color detection to find the path, and then a histogram or pixel summation in the y direction to obtain the curve. Divide the project into five distinct steps.

Thresholding

To generate the lane in our project, we used A4 sheets instead of having to discover and train an edge. Instead, we used color detection. The video of one run was initially recorded, but it was afterward edited to just show the lane utilizing other videographic techniques and the recorded video contrast. Thresholding is the technique of capturing the sheet's maximum number of colors.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 3 March 2023, pp: 961-971 www.ijaem.net ISSN: 2395-5252

Fig 3.3. Thresholding

Warping

Warping is a method to adjust the captured video into a birds-eye view, to make it more efficient for training.

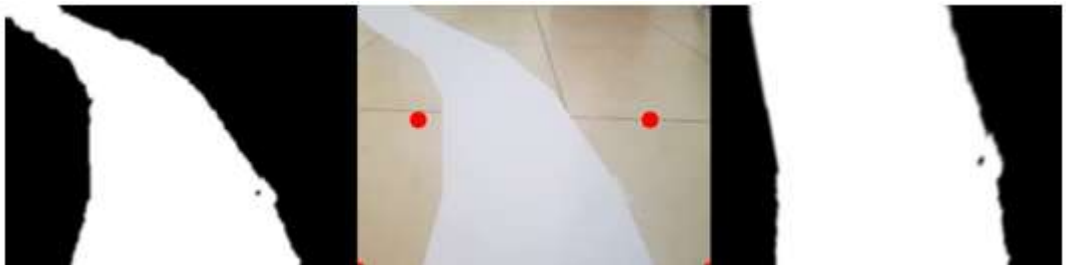

Fig 3.4. Warping

Histogram

The histogram is an efficient method to find the curve of the lane, It overcomes the drawbacks of the pixel summation technique.

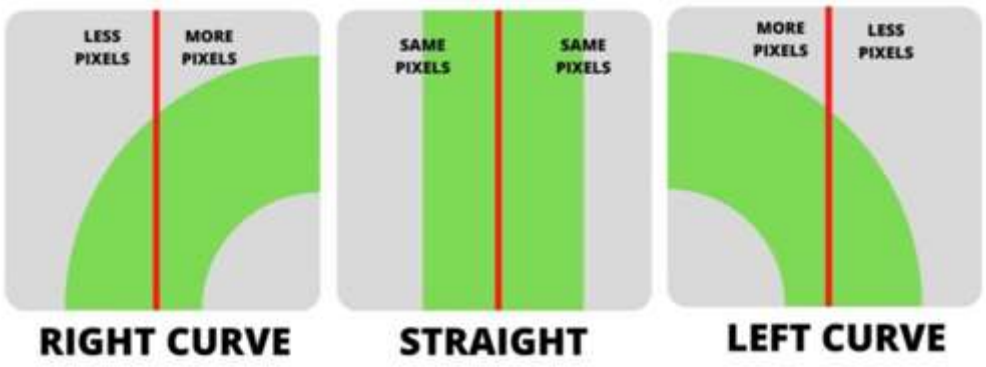

Fig 3.5. Pixel summation approach

Fig 3.6. Problem in pixel summation approach

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 3 March 2023, pp: 961-971 www.ijaem.net ISSN: 2395-5252

Although there is no curve in this instance because there are more pixels on one side, the algorithm will still produce either a left or a right curve. So how do we resolve this issue? The solution is straightforward: we move the center line.

Solution:

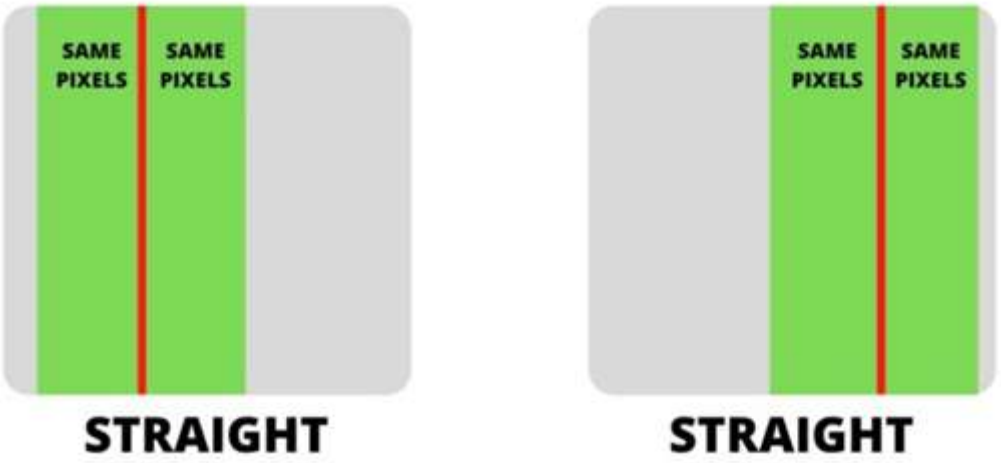

Fig 3.7. Histogram approach

Averaging

Averaging is a technique employed to allow smooth motion and avoid any dramatic movements.

Displaying

Displaying is a conventional method to avoid error and is used to display the curvevalue on the screen. An example negative value indicates left motion and a positive value indicates positive motion.

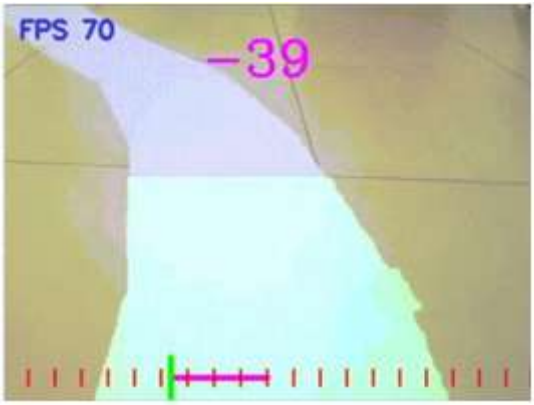

Fig 3.8. Displaying

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 3 March 2023, pp: 961-971 www.ijaem.net ISSN: 2395-5252

Lane Detection - Neural Networks

Currently, the most effective method for solving any supervised learning problem is using neural networks. In our project, the input layer contains pixels from the photographs collected during the run, and the output layer only comprises four layers—Forward, Reverse, Left, and Right.

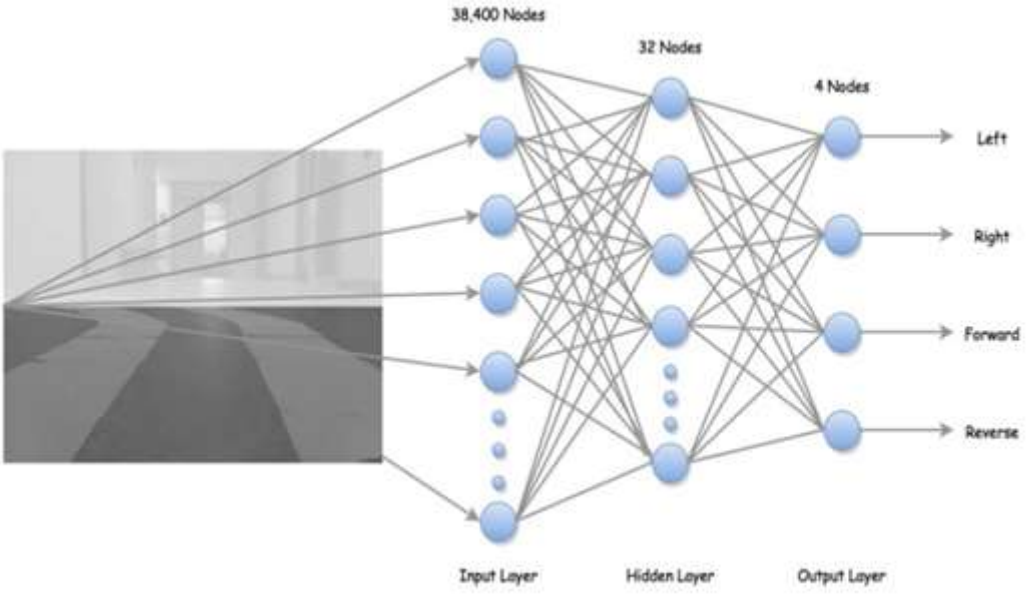

Fig 3.10. Neural network

Backpropagation Algorithm

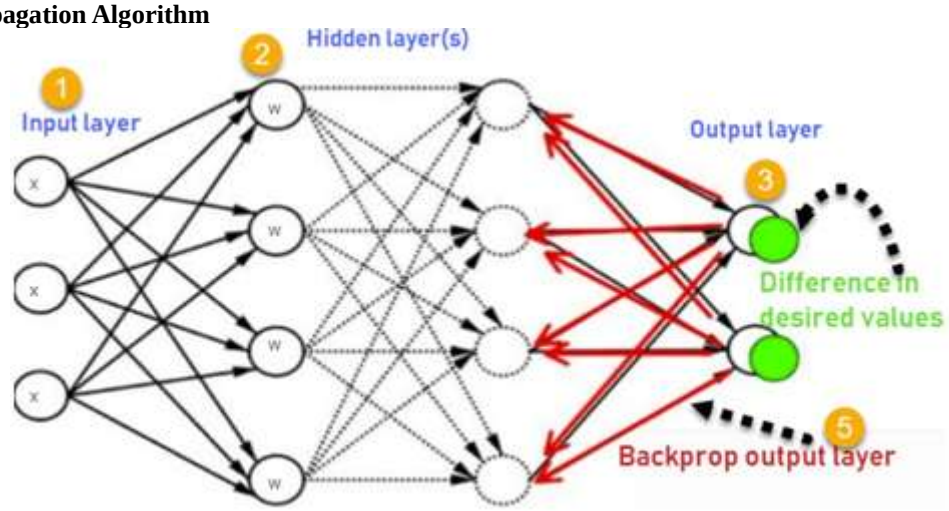

Fig 3.11. Backpropagation algorithm

The foundation of neural network training is back propagation. It is a method for modifying the weights of a neural network in accordance with the error rate observed in the previous epoch (i.e., iteration). By properly tweaking the weights, you may lower error rates and improve the model's reliability by broadening its applicability.

The term "backward propagation of errors" is shortened to "back propagation" in neural networks. It is a common technique for developing artificial neural networks. With regard to each weight in the network, this technique aids in calculating the gradient of a loss function.

The gradient of the loss function for a single weight is calculated by the neural network's back propagation algorithm using the chain rule. It

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 3 March 2023, pp: 961-971 www.ijaem.net ISSN: 2395-5252

computes one layer ata time effectively as opposed to a native computation. Although it computes the gradient, it does not define how to apply it. It expands the area of computation for the delta rule.

1. Input X enters the preconnected path.
2. Real weights W are used to model the input. Typically, weights are chosen at random.
3. Determine each neuron's output from the input layer via the hidden layersto the output layer.
4. Make a calculation of the output error
5. To change the weights such that the error is reduced, go back from the output layer to the hidden layer.

Need for back propagation
The most notable benefits of back propagation are:

1. Back propagation is a flexible method since it does not require prior knowledge of the network and is quick, easy, and simple to program.
2. It has no parameters to tweak outside the input numbers.
3. It is a common approach that usually yields positive results.
4. The characteristics of the function to be mastered don't need to be mentioned in particular.

Steps in Lane Detection Process
The three steps that make up the neural network's lane detecting process are as follows.

A) Data collection

Fig 3.12. Data collection

Using a motor module to drive the automobile and a webcam module to take the picture, the image of the lane is taken using a joystick. The gathered information is kept in both picture and log file formats.

A log file has,

- Image with a time stamp to distinguish the images.
- Steering angle.

B) Training

Fig 3.13. Training

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 3 March 2023, pp: 961-971 www.ijaem.net ISSN: 2395-5252

The training is conducted for 50 epochs using the log file provided in the data gathering module. A model.h5 file is created after training. A multidimensional array of the data produced during training is contained in the Model.h5 file.

C) Implementation

Fig 3.14. Implementation

The implementation of the created model is the final stage. The steering angles of the incoming webcam data are predicted using an h5 file.

IV. RESULTS

The model developed was tested with 11,000 images based on different case scenarios. The inputs in the form of steering angle vs No of Samples are plotted below.

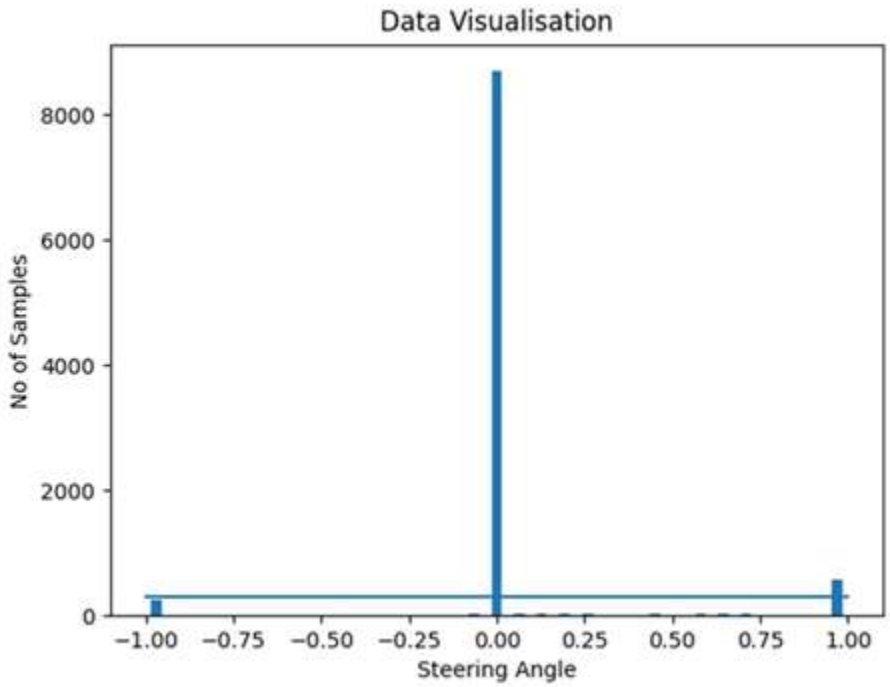

Fig 4.1. Data Visualisation

When the graph of sample size vs steering angle is anticipated, the steering angle is zero because most of the time our car was driven straight. This data visualization is shown in figure 4.1 above. However, we must have an equal distribution of the data, therefore we limited the number of samples to 350 and plotted the graph; Figure 4.1.2 displays the balanced data.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 3 March 2023, pp: 961-971 www.ijaem.net ISSN: 2395-5252

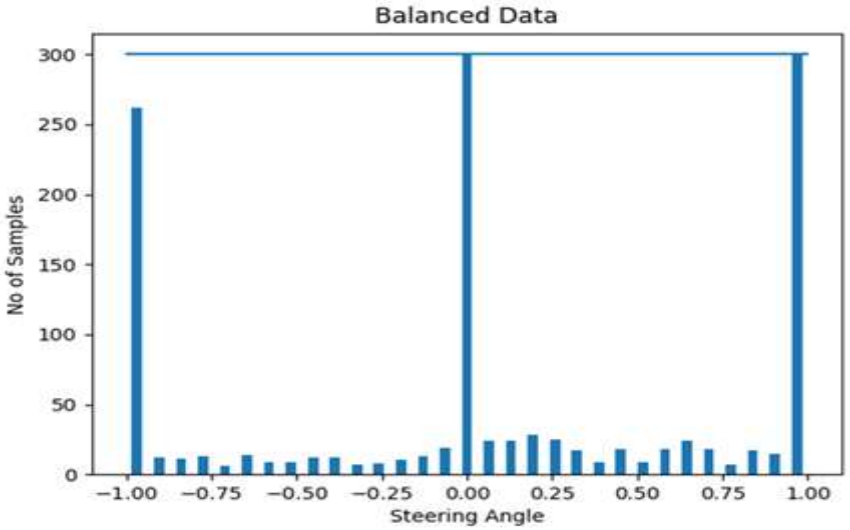

Fig 4.2. Balanced data

In order to incorporate the testing results as effectively as possible, the steering angle is balanced in the image above.

The Loss curve after the model has been trained is depicted in figure 4.1.3 below. Most of the time, the curve remains true to its nature. Training data is less noisy than validation data. The model underwent 50 epochs of training. to improve the steering prediction's accuracy.

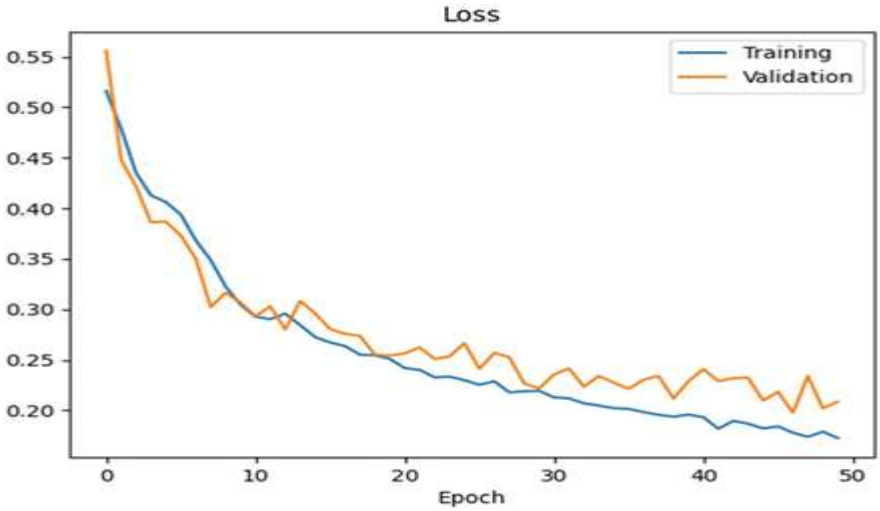

Fig 4.3. Loss curve

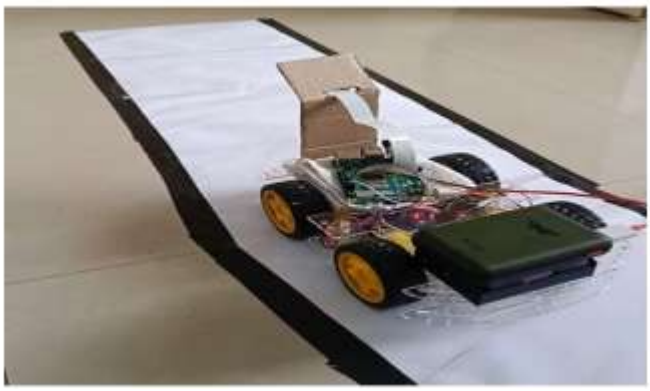

Fig 4.4. Snapshot of the developed model

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 3 March 2023, pp: 961-971 www.ijaem.net ISSN: 2395-5252

V. CONCLUSION AND FUTURE ENHANCEMENTS

Few conclusions were drawn from the analysis and research. The main conclusions are listed below and also the scope of the work in the future was also predicted in the domain.

Conclusion

In this study, an end-to-end deep learning-based real-time control framework for an autonomous vehicle has been constructed. The goal of this research is to autonomously drive a car using only the visual observations made by its camera, which is a novel problem in computer vision.

The optimistic outcome showed that it is possible to estimate the steering angles of the car directly from camera input data in real-time using a deep convolutional neural network. Embedded computer platforms like the Raspberry Pi 4 are strong enough to support the vision and end-to-end deep learning-based real-time control applications despite the complexity of the neural network. One of the most crucial lessons from this project is that we would not have been able to create a reliable model without the data provided by the numerous photos, steering angles, and possibly limitless augmentations. Camera latency is the problem we noticed when training and testing the network. The delay between the camera sensor capturing the scene and the computer reading the digitized image data is what the term means. Unfortunately, this period, which is typically between 300 and 350 milliseconds, can be considerably longer depending on the camera and the Pi's capability. The latency of human perception, which is known to be as fast as 13 milliseconds, is noticeably lower than this. Because the deep neural network would examine stale scenes, higher camera delay could significantly impact control performance, especially for safety-critical applications. As can be seen, there are a variety of directions we may go in order to advance this project and produce even more compelling outcomes. We will keep looking into ways to improve prediction accuracy while training the network, find and employ low-latency cameras, and enhance the functionality of the RC car platform, particularly with regard to precise steering angle control. In order to achieve higher levels of autonomous vehicle development, we will also incorporate throttle into the model.

Future Enhancements

By enhancing the algorithm and using cutting-edge machine learning, the work could be improved. For quick processing, multi-layered processors might be used. The current model just detects the lane and follows it. In the future obstacle detection systems to detect the obstacle and pause can be developed, which can beenhanced by bypassing the obstacle and utilizing a more sophisticated algorithm to choose a different route.

As can be seen, there are a variety of directions we may go in order to advance this project and produce even more compelling outcomes. We will keep looking into ways to improve prediction accuracy while training the network, find and employ low-latency cameras, and enhance the functionality of the RC car platform, particularly with regard to precise steering angle control. In order to achieve higher levels of autonomous vehicle development, we will also incorporate throttle into the model.

Camera latency is the problem we noticed while training and testing the network.

1. Algorithm for detecting obstacles
2. To perform as well as human experts
3. When new high-speed levels are added, the accuracy rate of all algorithms should be increased.

REFERENCES

[1]. Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, Xin Zhang, Jake Zhao, Karol Zieba, "End to End Learning for Self-Driving Cars ", (2016).

[2]. Hiral Thadeshwar, Vinit Shah, Mahek Jain, Prof. Rujata Chaudhari, Prof. Vishal Badgujar, "Artificial Intelligence-based Self-Driving Car",4th International Conference on Computer, Communication and Signal Processing (ICCCSP), (2020). [3]Syed Owais Ali Chishti, Sana Riaz, Muhammad Bilal Zaib, Mohammad Nauman, "Self-driving Cars Using CNN and Q-learning", (2018).

[3]. Wuttichai Vijitkunsawat, Peerasak Chantngarm, "Comparison of Machine Learning Algorithms on Self-Driving Car Navigation using Nvidia Jetson Nano".17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), (2020).

[4]. Truong-Dong Do, Minh-Thien Duong, Quoc-Vu Dang, and My-Ha Le*, "Real-Time Self-Driving Car Navigation Using Deep Neural Network". 4th International

Conference on Green Technology and Sustainable Development (GTSD), (2018).

[5]. Pride Cai, Hengli Wang, Huaiyang Huang, Yuxuan Liu, and Ming Liu, "Vision-Based Autonomous Car Racing Using Deep Imitative Reinforcement Learning" , IEEE ROBOTICS AND AUTOMATION LETTERS, VOL. 6, NO. 4, (2021).

[6]. Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, Pearson Education, Inc, Third Edition, (2008).

[7]. Tom M. Mitchell, Machine Learning, McGraw-Hill Science/Engineering/Math, First Edition, (1997).