Smart Autonomous Vehicle Using End to End Learning

Pranav Gupta[1], Vaibhav Singh[2], Anubha Parashar[3]

[1]Scholar, Manipal University, Department of Computer Science and Engineering, Jaipur, Rajasthan, India
[2]Scholar, Manipal University, Department of Information Technology, Jaipur, Rajasthan, India
[3]Scholar, Manipal University, Department of Computer Science and Engineering, Jaipur, Rajasthan, India
Email : [1]pranavg387@gmail.com ,[2]vaibhavsinghcr007@gmail.com, [3]anubhaparashar1025@gmail.com

Abstract - Autonomous Driving has passed the point of being called the next big step, as the smart car revolution is already taking shape around the world. Self-driving cars are relevant if not prevalent and the main obstacles to reach mass adoption are customer acceptance, cost, infrastructure and the reliance on several onerous algorithms that include perception, lane detection, path planning and variation in pathways. The objective of this research paper is to tackle the mentioned problems with a straightforward, reproducible and cost-effective solution, using end to end learning and replacing the numerous sensors with a camera. These were optimized directly by the proposed system with limited background processing. In this research paper authors achieved this by mapping pixels from only a single front-facing camera to direct driving instructions. The results obtained were better than state of the art and achieved the aim of the study proficiently.

Keywords - Autonomous Driving, Convolution Neural Network, Transfer learning.

1. INTRODUCTION

A smart car is generally made of 2 controlling units that dictate its actions, High-level controller, and Low-level controller. The high-level controller takes input from its components i.e. the driver (in this case the camera), its surroundings such as the traffic and the sensors in place, it then after deducing the correct actions to be taken sends signals to the low-level controller that controls the brakes, steering, engine and throttle [8][9][12]. To do this successfully it needs to understand driver psychology, how and when a specific maneuver is necessary, which changes with the terrain and the area the car drives in since driver temperament and driving style cannot be universally generalized[6][7][10]. To understand and correctly predict such outcomes is precarious. Studies show that if the driver is given even a half a second extra before a collision, 60% of accidents can be avoided and this percentage increases to 90% if one second of warning time can be provided [4]. Such results stress the importance of timely and correct decisions that face problems with the conventional architecture of a smart car system. It is also difficult for cars relying on so many sensors to be able to adapt to new surroundings and to reconfigure the system to achieve a different goal based on learning that occurs on the move, especially when noisy sensor data is received [5]. The foundation of the end to end approach was laid almost 14 years ago in an effort by Defense Advanced Research Projects Agency (DARPA) known as DARPA Autonomous Vehicle (DAVE) [10] where a radio control (RC) car drove through a junk- filled alley [9]. It demonstrated that this approach was viable and adequate for the functioning of a smart car. The end to end approach thus provided alternatives which were uncomplicated and also easier to test [8]. With the advancement in technology, the study achieved similar functionality more skillfully and with limited computational power. At the centre of this end to end learning is the Convolutional Neural Network which is trained based on the actions of the driver while driving. The data collection was automated which made it possible to do rigorous experimentation as the system itself could be reproduced efficiently at a low cost. Also, it can be shared among multiple systems in order to make the learning faster and for better adaptation in different terrains.

2. LITERATURE REVIEW

This journey to make a Smart Autonomous vehicle began with the invention of modern cruise control in 1948. Many dedicated steps have been taken in this direction to make self-driven cars a reality. ALVINN (An Autonomous Land Vehicle in a Neural Network) diversified the field by combining End to End learning (Shibata et al. 1997) with a Neural Network to resolve the problem (Dean A. Pomerleau 1989). DARPA (Defense Advanced Research Projects Agency) then gave perspective to what can be achieved with the technical know-how of the time (Net-Scale Technologies Inc. 2004). In the recent past when a convolutional neural network was trained to map unaltered pixels from a camera to be interpreted directly into steering commands (Mariusz Bojarskiet al. 2016), it demonstrated the reliability of this approach in contrast to the more modular alternatives[1].

3. PROPOSED METHODOLOGY

3.1 Data Collection

Initially for testing the feasibility of approach and the plan of action, Udacity's self-driving car simulator was used to generate training data [2].

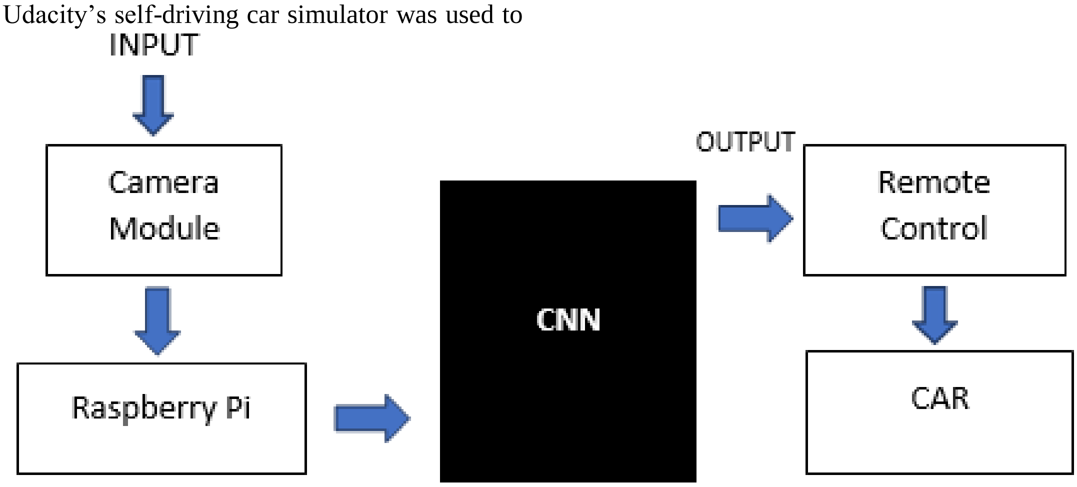

Fig. 1. The camera takes an image as input through raspberry pi where it is passed to a machine learning model (architecture as explained in Figure 3) for predicting direction which is further converted to low level output suitable for remote controllers in order to control the car.

It gives users the liberty to drive a car on preset tracks and the data is collected frame by frame along with the corresponding controls we take while driving the car. The data also contains steering angle and acceleration on each frame. This data helped actualize the direction of this study and formulate what kind of data we need so that it could be mapped onto directions. Thus, it had to be generated independently and specific to the research work. This was because of two major reasons, first that the RC car was not taking directions in the form of steering angle but only left, right and forward. The second was that the camera angle played an important role in how the car understood the road ahead which greatly affected its decision-making capabilities.

We used a simple yet effective data collection method which is shown in Fig. 1 The camera view was visible on the screen of a remote device using the VNC viewer to stream the Camera's Input [13].The car was steered on self- laid tracks of different orientations that were similar but not the same to which the car had to be tested on. It mimicked unmarked roads and the environment where it will be tested, the car had to stop, turn and avoid a collision or going off-road. The frames were captured with the help of the pi camera mounted on top of the RC car and it also recorded the directions at every moment which were time- synchronized and embedded in the final data set. And this was repeated on diverse tracks until enough data that could be useful was collected. Thus, the data set had frames containing the tracks as viewed by the pi camera along with the directions as shown in Fig. 2.

3.2 Subsystems Input unit (camera)

A Raspberry Pi board that draws its power from an external power bank, attached with a Pi camera is used to collect input data. The camera was fitted on top of the car in such a way that it provided vision to the car and was overlooking the road and immediate surroundings. The client program runs on Raspberry Pi for sending frames to the Neural Network which is also stored in the pi.

Fig. 2. Sample Section of the Collected Dataset

Processing unit (Raspberry Pi)

The processing unit (computer) manages multiple tasks: receiving data from the Pi, inferencing the neural network prediction (steering commands), and sending the resultant commands to the remote control for the final movement as shown in fig.1

RC car control unit

The RC car has an on/off switch type controller. A Raspberry Pi is used to simulate button-press actions by connecting the remote to GPIO pins [3]. The model returns the resultant commands to the pi using the serial interface, the Pi then reads the commands and writes out LOW or HIGH signals, simulating action of a button being pressed to drive the RC car.

3.3 Feature Extraction

In each image unnecessary data was cropped out i.e. the height of the images was reduced in order to increase the necessary information in the frame and the images were resized so that less memory gets occupied. The images were then converted into YUV color-space as it enables efficient reduction of the quantity of information necessary to represent the image of subjectively parallel quality [11]. To increase the sample data, images containing turns were flipped with the corresponding directions. The brightness of frames was changed to random degrees which enables our system to perform well even in different lighting conditions thus making our model more robust. Furthermore, Gaussian blur was applied to each frame for smoothing the image and to reduce noise in the captured frame. Also, it was observed that unfairly large amount of the recorded data was corresponded to the car driving in a straight direction as compared to that of recorded while taking right or left turn, thus the data had to be normalized to avoid the possibility of a biased model [1]

3.4 Network Architecture

An image of size $3 \times 224 \times 224$ as input was fed into the pre-trained VGG 16 for feature extraction. The top layer that is fully connected layers of vgg16 which is used for prediction purposes is not loaded as transfer learning technique was used for the custom dataset. The output was flattened from the 5th block shown in fig. 3, after a pooling layer of shape $7 \times 7 \times 512$ and applied two fully connected layers, each of size 128 followed by an output layer having three neurons predicting left, right and forward. We used Adam optimizer with a learning rate of 0.01 and categorical cross entropy loss.

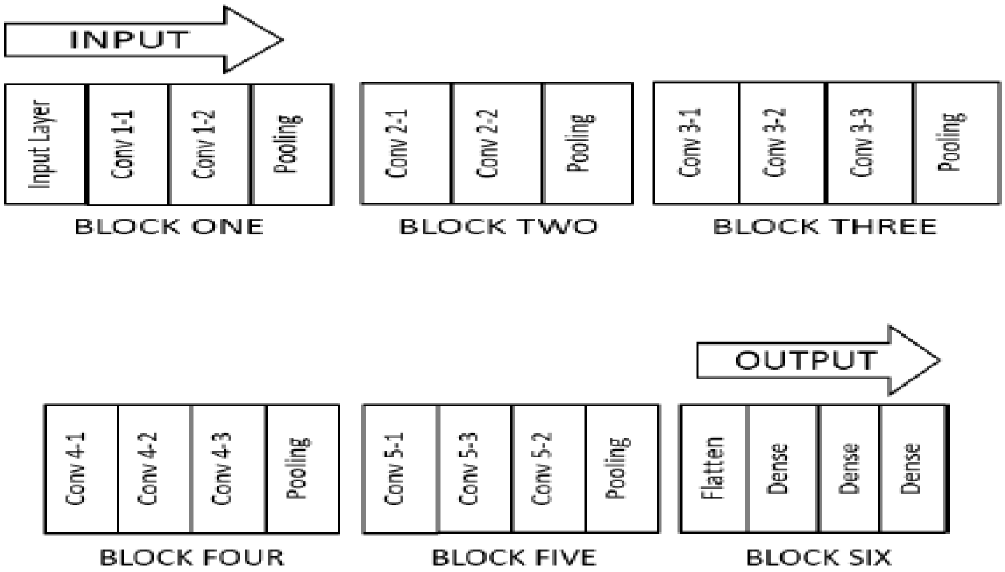

Fig . 3 The architecture of the neural network implemented for the project

4. RESULT ANALYSIS

4.1 Simulation

Before testing the hardware on the actual field, the working of the model and the systems in place were to be tested. To achieve this, a video was fed through the model frame by frame corresponding to which directions were generated. After obtaining these results it was apparent that the basic functionality was to the point as shown in Figure 4, which was also confirmed by the accuracy graph shown in Figure 5, but due to the hardware restrictions i.e. having only left, right, and forward controls any concrete conclusions could not be drawn from this effort and field test was the only way to be able to understand the working and the shortcomings at length.

4.2 Result Analysis

After rigorous test runs on tracks apart from which the car was trained on, it was concluded that the car was able to function competently in a controlled environment.

Fig. 4. The prediction and target directions are shown on bottom left and bottom right of an image simultaneously which enables smooth testing of the model

The neural network was also working fittingly and gave better results than those obtained by previous studies as shown in Table 1. The only complications faced were pertaining to the type of car used, as implementation of calculated output was coarse while making sharp turns. Such complications were later assessed could be comfortably resolved.

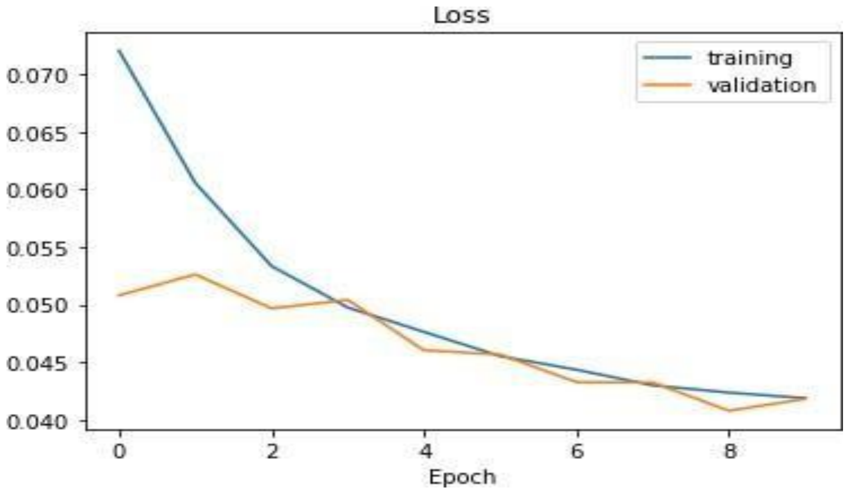

Fig. 5. Accuracy graph for the obtained CNN

Table 1. **A comparison of the Accuracy of the current study with that of the others**

pb nm 1-5	Title of Paper	Accuracy
Muller, U (2006)	Off-Road Obstacle Avoidance through End-to-End Learning	75 %
Mori,K (2019)	Visual Explanation by Attention Branch Network for End-to-end Learning-based Self-driving	79 %
Xu, H (2017)	End-to-end Learning of Driving Models from Large- scale Video Datasets	84.6 %
Bojarski, M (2016)	End to End Learning for Self-Driving Cars	88 %
Proposed System	Smart Autonomous Vehicle using End to End Learning	98 %

5. CONCLUSION

The End to End Learning approach in building an autonomous vehicle is an effective alternative to the traditional one. The car learned driving mannerisms and could detect road without the need for explicit labels, it proved to give viable results in a limited time frame which was also very cost- effective. The project thus developed should be viewed as independent of the vehicle like a CNG kit as the vision was to develop a setup that can be incorporated into any vehicle given that its acceleration and direction can be controlled. Precise angle control will bear better results and more work needs to be done on how to make hardware implement the output of the neural network more precisely.

REFERENCES

[1] Jiakai Zhang, End-to-end Learning for Autonomous Driving. Dissertation, New York University, 2019.

[2] Udacity, Inc. Self-Driving Car Simulator. https://github.com/udacity/self-driving-car-sim. Accessed on 5 Feb 2019.

[3] Zheng Wang (2018) Self Driving RC Car.https://zhengludwig.wordpress.com/projects/self- driving-rc-car/. Accessed 22 Jan 2019.

[4] Desai, S. and Desai, S., Smart vehicle automation. Int. J. Comput. Sci. Mobile Comput, 6(9), 2017.

[5] Hussain, R. and Zeadally, S., Autonomous cars: Research results, issues, and future challenges. IEEE Communications Surveys & Tutorials, 21(2), pp.1275-1313, 2018.

[6] Rajasekhar, M.V. and Jaswal, A.K., August. Autonomous vehicles: The future of automobiles. In 2015 IEEE International Transportation Electrification Conference (ITEC), pp. 1-6. 2015.

[7] Rosenzweig, J. and Bartl, M., A review and analysis of literature on autonomous driving. E-Journal Making-of Innovation, pp.1-57, 2015.

[8] Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L.D., Monfort, M., Muller, U., Zhang, J. and Zhang, X., End to end learning for self-driving cars. arXiv preprint arXiv:1604.07316, 2016.

[9] Jackel, L.D., Krotkov, E., Perschbacher, M., Pippine, J. and Sullivan, C., The DARPA LAGR program: Goals, challenges, methodology, and phase I results. Journal of Field robotics, 23(11- 12), pp.945-973, 2006.

[10] Lecun, Y., Cosatto, E., Ben, J., Muller, U. and Flepp, B., Dave: Autonomous off-road vehicle control using end-to-end learning. Courant Institute/CBLL, http://www. cs. nyu. edu/yann/research/dave/index. html, Tech. Rep. DARPA-IPTO Final Report, 2004.

[11] Podpora, M., Korbas, G.P. and Kawala-Janik, A., YUV vs RGB-Choosing a Color Space for Human-Machine Interaction. FedCSIS Position Papers, 18, pp.29-34, 2014

[12] Pan, Y., Cheng, C.A., Saigol, K., Lee, K., Yan, X., Theodorou, E. and Boots, B., Agile autonomous driving using end-to-end deep imitation learning. arXiv preprint arXiv:1709.07174, 2017.

[13] Wang, Z., Lim, E.G., Wang, W., Leach, M. and Man, K.L., November. Design of an arduino-based smart car. In 2014 International SoC Design Conference (ISOCC), (pp. 175-176, 2014.