Self-Driving Car using Convolutional Neural Network and Road Lane Detector

Eadhunath V[1], Amir Suhail[2], Jayant Waghmare[3], Rishab Mishra[4], Prof. K. U. Jadhav[5]
Department of Computer Engineering, Sinhgad Academy of Engineering,
Pune – 411048, India

***Abstract*— This research paper provides a detailed look into the of making a prototype of an autonomous, self-driving car that is powered by YOLO object detection algorithm and Road Lane Detection. The final product will be a car able to detect traffic and maneuver its way on the roads by keeping track of the road lanes on its either sides. A lot of emphasis has been laid on Computer Vision as it is the main driver of the car. With the development of such technology, we hope to minimize the amount of traffic accidents caused on the roads due to driver carelessness.**

Index Terms— YOLO, CNN, Deep Learning, Road Lane Detector.

I. INTRODUCTION

In India, according to a survey, more than 150,000 fatalities are caused each year in traffic accidents. Thats about 400 fatalities a day and far higher than developed auto markets like the US, which in 2016 logged about 40,000. Nearly 55% of these involve 4 wheel vehicles or buses.

The idea of a self-driving car is an effort to minimize the accidents caused by careless and violent driving of 4 wheelers. The car should be such that it can recognize the distinct objects found along the roads and identify the elements of interest such as other vehicles, pedestrians, traffic signals etc. Another major challenge is building its ability to navigate properly in between the road lanes while complying with the traffic rules. In order to make a sustainable self-driving car, we need three main parts: object detection, lane detection, and controller. In this paper, we have combined these parts to help the driver with more info about the object in their driving environment and the road lane guidance. The proposed method combines YOLO for the object detection, polynomial regression with thresholding as the road lane guidance, and a controller that manages the information between those two systems. The proposed methods can be used for object detection, object position detection (left, front, or right), steering suggestion,
and road lanes guidance.

II. PROPOSED SYSTEM

The major components of this system include:

- Camera Module
- Wireless Video Transmission Module
- Python code running the Algorithms
- Arduino Micro-controller
- Dummy Car

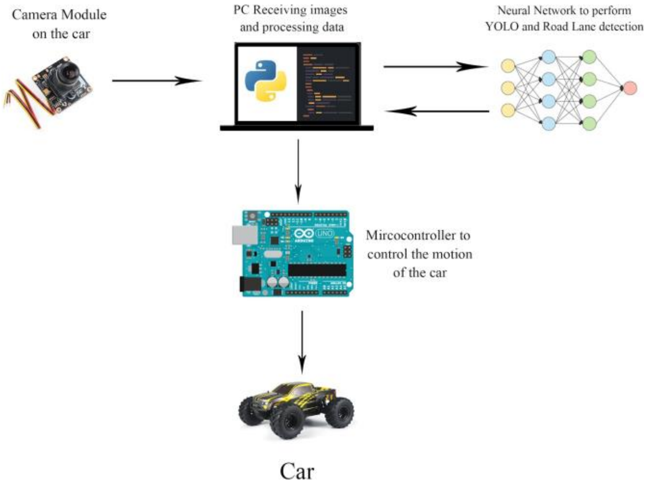

Fig. 1. Warping Process

A. Camera Module

This is the main source of data input arriving from the car to the computer. The same video feed of the Camera can be simultaneously used for both - Object Detection and Road Lane Detection. The camera being used in this case is an FPV camera

B. Wireless Video Transmission Module

A simple Radio Frequency based transmitter-receiver module that is used to send the live video feed from the camera on the car to the Machine. In this particular case, we have used the TS 832 transmitter and the RS 832 receiver.

C. Python Code

This is the main power house of the system that is capable of conducting the computations and manipulations on the incoming video feed, The same code takes care of YOLO and Road Lane detection algorithm, The Python Language was chosen to implement is due to its fast paced nature and the ready availability of support libraries like Tensorflow and OpenCV which are instrumental to the working of this system.

D. Arduino Micro-controller

A micro-controller is a much better option from controlling hardware devices than a micro-processor (PC). Arduino is our go to choice from a micro-controller due to its ease of use, simplicity and the vast community.

E. Dummy Car

For implementing a car, we have used a toy RC car and re-purposed it. Arduino has been fit into it and the rear motors are connected with the Arduino through a L298 Motor-Driver. For steering, a servo motor has been attached which turns the front wheels of the car, exactly the way real cars turn. Now, the Arduino can simply write the specific angle for the amount of turn.

III. ALGORITHMS USED

A. You Only Look Once (YOLO)

For detecting simple objects from images captured by camera, a deep learning technique named YOLO - You Only Look Once, was proposed. YOLO is a new approach to object detection. Here, we frame object detection as a regression problem to spatially separated bounding boxes and associated class probabilities. From full images a single neural network predicts bounding boxes and class probabilities directly in one evaluation. The detection pipeline can be optimized directly on detection performance as the detection is single network.

Prior detection systems re-purposed classifiers or localizers to perform detection. They apply the model to an image at multiple locations and scales. High scoring regions of the image are considered for detection.

In YOLO algorithm, we apply a single neural network to the full image which divides the image into regions and predicts bounding boxes and probabilities for each region. These bounding boxes are weighted by the predicted probabilities. This model looks at the whole image at test time so its predictions are informed by global context in the image. It also makes predictions with a single network evaluation unlike systems like R-CNN which require thousands for a single image. This makes it extremely fast, more than 1000x faster than R-CNN and 100x faster than Fast R-CNN.

B. Road Lane Detection

The road lane detector consists of four main parts: cropping, filtering, detecting the road lane, and adding.

The initial step is the warping process. In this we crop the Region of Interest(ROI) and change the perspective of the image. Figure 2 shows the resultant of this process. It clearly represent how the way in which the road's perspective has been changed so as to make computations easier.

The next step is to filter this cropped and warped image to differentiate between the road points and the background of the image. In the filtering process, we pick the range of white and yellow colour out of the images as most road lines are painted white or yellow colour. After the filtering phase, we detect the road lane points by getting the non-zero readings from the previous process. First, we crop the images into 10 sub-images and divide them into left and right. This helps us in getting the peak of nonzero values of each lane (left and right). Finally, we add the area around the peak values to the list of left and right points into the collected points.

The next step is obviously the road-lane detection for which the polynomial regression is used. This can be achieved easily using the Numpy function *np.polyfit()* in Python. It enables us to approximate the lane/curve by fitting lane points (y) that have been collected with coefficient points (p) into the best approximated lane points (Polyfit) by minimizing the error (E) from the fitted lanes. Hence, by getting right and left points, we could make functions to approximate the lane structures by feeding them into the regression method to find the best fit of the road lane detection.

The final step is to now convert this processed image into a real-time projection image by reversing the warping process that was applied in step 1.

IV. EXPERIMENTAL RESULTS

A. Raw Video Feed

This is the feed that is received by the Python Code. To obtain to from the RS 832 receiver unit, an A/V to USB converter is used and the feed is read using OpenCV's *openVideo()* function. This feed is used in the raw format for YOLO and converted to required part for Road Lane Detection.

Fig. 2. Raw Feed from the camera.

B. YOLO Algorithm

After receiving the raw input, the feed is passed onto a Convolutional Neural Network. This network is pre-trained by Google and we are required to add the classes of items that we need to categorize. We chose all the classes that are generally visible in the road like all vehicles, road signs, traffic lights, people and animals etc. The following is the output of the YOLO algorithm in its attempt in classifying objects from a sample video feed. It shows the Confidence score and the pixel distance from the camera, which is used for braking of the car.

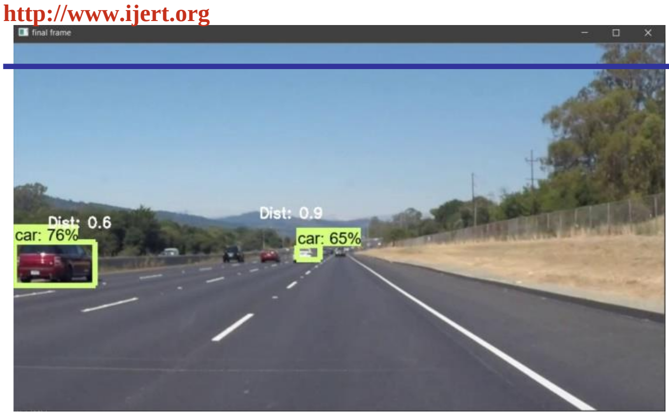

Fig. 3. YOLO Algorithm making predictions on the input feed.

C. Road Lane Detection

These are all the phases that go into the Road Lane Detection. They represent cropping and Warping, using Canny Edge Detection, and finally making the line,

Following this, the prediction of the center of the road by considering the center of the road lane is made and the car is made to steer in that direction. The angle of the servo is calibrated based on how sharp the turn needs to be taken.

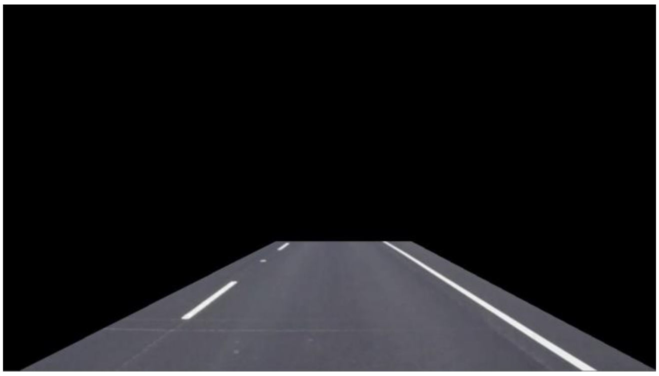

Fig. 4. Region Of Interest (ROI)

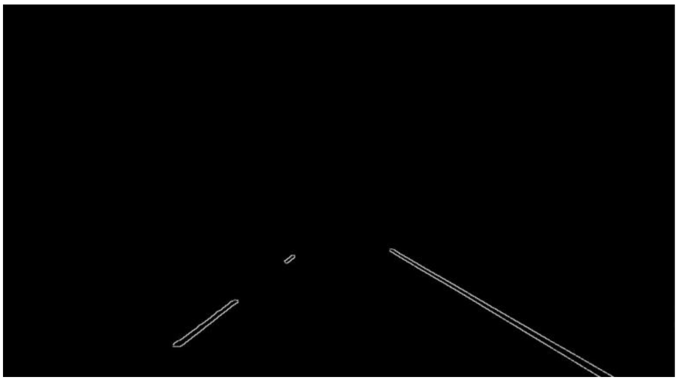

Fig. 5. Canny Edge Detection.

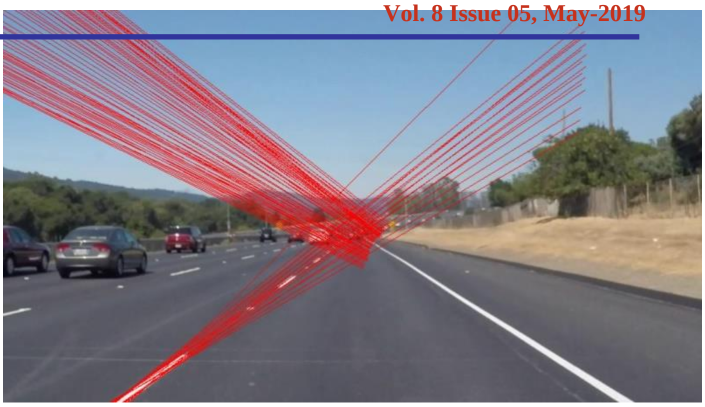

Fig. 6. Hough Lines.

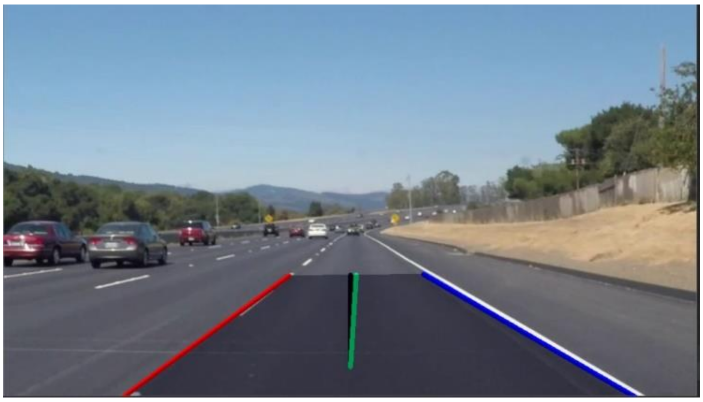

Fig. 7. Final lane.

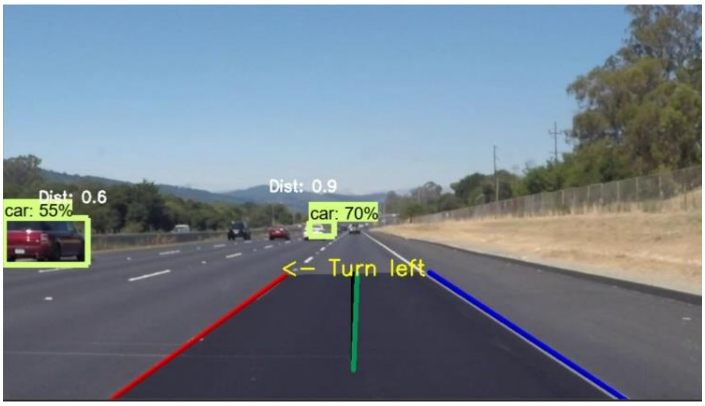

Fig. 8. Final output.

FUTURE SCOPE

This prototype model is just a base of a simple implementation of a self-driving vehicle. This field offers a lot of scope to work on in the future mainly involving the improving the fault tolerance of the car and its ability to adapt in various environmental conditions. Some such future scopes are listed below.

- Adding a GPS module to self-navigate to required place.
- Providing an on-board controller for managing its settings
- Using multiple cameras on different angles to make the car capable of overtaking and by-passing other vehicles in the same lane.
- Adding sensors that can help when the camera does not work.

CONCLUSION

By combining all of these technologies together were able to put together this prototype of a self-driving car with features like obstacle detection and crash avoidance and automatic navigation in a road with road lanes.

With the use of this prototype as the base, a lot of advancements can be made in the field of self-driving vehicles and they can be made more and more secure and fault tolerant.

ACKNOWLEDGMENT

We would like to thank our Project Guide Prof. K.U.Jadhav for guiding us through the process of making a literature survey. We would also like to extend our thanks to our Head of Department and the University for allowing us to choose this topic and work on it.

REFERENCES

[1] Towards Self-driving Car Using Convolutional Neural Network and Road Lane Detector.

[2] You Only Look Once: Unied, Real-Time Object Detection. Authors :
Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi

[3] YOLO9000: Better, Faster, Stronger, IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Authors: J. Redmon and A.
Farhadi

[4] John Canny : A Computational Approach to Edge Detection