International Journal of Control and Automation
Vol. 13, No. 2, (2020), pp. 757 - 763

Self Driving Car Using Machine Learning Implementation

S P Maniraj*, Deep Chaudhary, Vankayala Hari Deep, Vishesh Pratap Singh, Anant Bhardwaj

SRM Institute of Science and Technology, Ramapuram Campus
manirajp@srmist.edu.in,chaudharyluvvy@gmail.com,harideep0000@gmail.com, vishesh712@gmail.com,anantbhardwaj1999@gmail.com

Abstract

Self-driving vehicles or autonomous vehicles were being implemented by the use of AI and machine learning tool like computer vision for executing this process it is needed to use sensory inputs to train the AI to make decisions the sensory input consists of cameras, sonar, laser, radars like tools to give the input which is then processed by the model by using object detection which has two aspects that is image classification and image localization which is used to detect the object type and its distance from the observer point of view. Lane detection is the key property of the model which is used to keep the vehicles in the same lane, it depends on the combination of methods like line selection, hough transform, and spatial CNN.

All these features are necessary to train the algorithm and it also requires adaptive parameter tuning according to different times of the day and makes it versatile for different conditions that's why adaptive features are necessary. This model is implemented using python 3 libraries like OpenCV to get a good accuracy for the model.

Keywords: *Lane detection; hough transform; image processing; computer vision.*

1.Introduction

For lane detection we use OpenCV library which is used to perform hough transform, firstly OpenCV needs to engaged for the process then the video is processed by making the frames of the video it is done by making frames at an interval or rate of 10 milliseconds then the canny detector is applied to the model this algorithm is used to detect edges on the fast real-time environment.

This algorithm is optimal to detect changes in the gradients which are large or luminosity to define the path by identifying edges this is instantiated by doing several other processes like noise reduction which is necessary as it can lead to false output so it is used to smooth the image by using a Gaussian filter which is 5*5 to reduce the sensitivity of the detector by setting the pixel values to the weighted average of the nearby pixels followed by intensity gradient that detects the relative orientation of the edge that if it's vertical, diagonal or horizontal, then the non-maximum suppression is done to thin and sharp the edges and it is followed by Gaussian blurring of the false objects from the frame by applying hysteresis thresholding in which the edges are considered if it lies between the minimum and maximum threshold given by the user by the insight of strong pixels.

After that lane segmenting is done to discard the irrelevant areas and then finally visualization is performed after using hough transform to identify the left and right boundaries.

Another approach for lane detection includes spatial CNN as the Hough transform algorithm is only applicable to the straight lines it can fail in conditions like the sharp curves or dotted lines that will lead to the failure of the model. Spatial CNN is a quite progressive algorithm for serving this purpose as it is unlike the CNN which is used to extract the semantics from the images by extracting it from pixels rather than that it could not identify spatial relationships of the pixels present in the frame it includes rotational and translational relationships which are used to detect hard objects like poles on the way to distinguish the path perfectly.

Spatial CNN detects the lanes by passing up the information between the layers also called slice passing to transfer the data to the adjacent layer for giving a better insight into the spatial relationships to the model. It works accurately in different weather and lighting conditions and thus it proves to be a better algorithm for lane detection it is then trained and tested for better visualization.

In this object, classification is also done to make the model able to detect the objects and the obstacles to detect and aid in the better ride quality, therefore, it ensures it by doing image classification and localization.

2.Concept

Self-driving or autonomous driving requires the object detection method to understand its surroundings. The object detection consists of object classification and localization for this algorithm that supports are Sliding window algorithm and the YOLO algorithm.

3.Sliding Window Algorithm

This is used to traverse the sliding window over the image which detects the presence of the objects like human, animals, cars or traffic signals it is used to return true prediction based on the data fed but it has shortcomings that are the images were not enough small or big for detection so multiple size windows would be needed which will require high processing power.

4.YOLO Algorithm

In this, the image is split up into the grid and it is then being fed to the convolution neural networks that give the class probability map which enables the object classification and it proves to be more efficient.

For object localization, a non-max suppression algorithm is used as it is being efficient in winding which kind of bounding box to the object needs to be used.

The key aspect of self-driving is to build the architecture of the system that signifies the implementation of the model.

Figure 1. Sensor Observation

It uses techniques like computer vision, use of sensors data, running deep learning algorithms, planning the path for driving, use localization to detect the distance of the objects taking itself as the frame of reference and the actuator system that controls the mechanical movements.

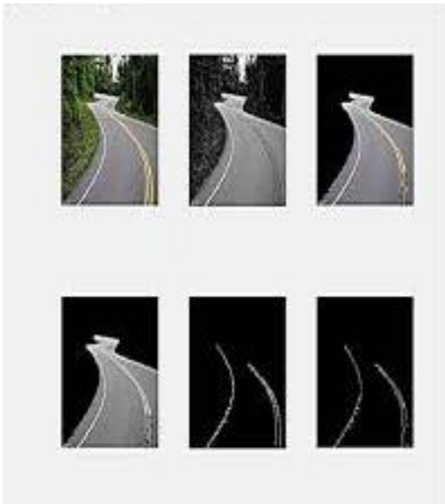

Figure 2. Sensor Analysis

5.Background

These self-driving models depend on the software and hardware processing to drive on the road without the interference of the user in these models hardware gathers the input which is then processed by the software it compiles the input run machine learning algorithms to test and train the model to make the decisions.

These model gets better and optimal in use depending on the data to be processed it becomes smarter and works better.

Self-driving Cars hardware which is used to collect data is being kept the same and only the software needs to be updated and changed.

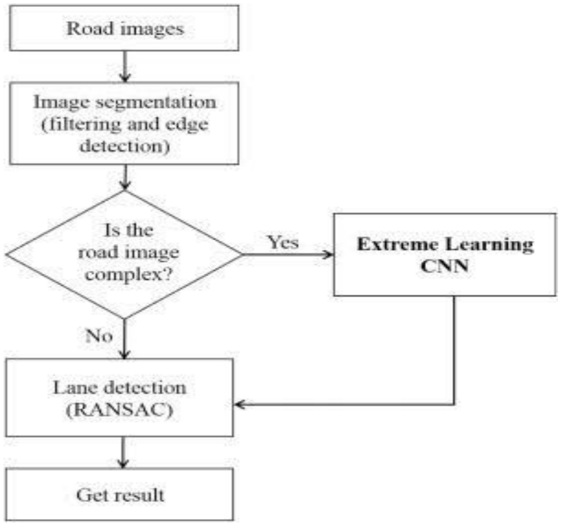

Figure 3. Sensor Flow

6. Hardware used

6.1.Camera

It is being the most important sensor for self-driving technology, there was the development of new technology that enables the computer to form a 3D visualization data based on the visual data provided. Based on research 8 cameras should be optimal to use.

6.2.Radar and LiDAR sensors

These sensors work with each other in it Radar is having low resolution and it is a radio wave-based sensor so it will work better in snow or rain whereas LiDAR sensors were light-based these sensors were having work to generate a high-resolution 3D map of surrounding areas.

6.3.Computing power

Based on the need for real-time data processing all the graphical processor units are being inefficient thus technology of NNA (Neural Network Accelerator chip) is being discovered to process real-time data effectively. NNA’s have high computing power compared to GPU's and CPU's.

6.4.EDGE Detection algorithm

Edges obtained from the real-time are generally hampered due to fragmentation or the presence of the missing and false edge segments. Thus Edge detection is the most prevalent feature it is used to be applied on the image it is the fundamental method of the image processing part it is important as it analyses an image, recognizes the pattern and use computer vision algorithms to detect the edges it will lead to the formation of connected curves that denotes the boundaries and also surface marking because it allows checking the discontinuities and orientation on the surface.

ISSN: 2005-4297 IJCA

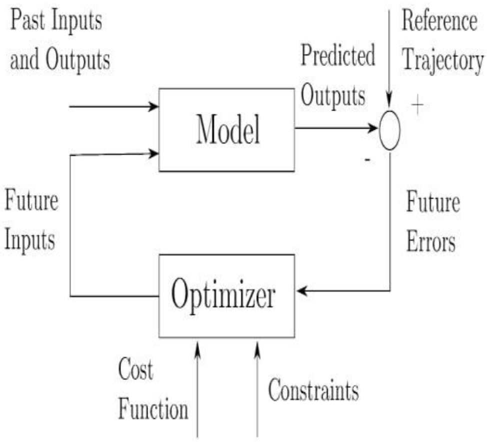

Figure 4. Sensor Flow Analysis

However, it is difficult to obtain the ideal edges that the algorithm uses to detect from real-life images that are having some medium complexity.

6.5. Design and Implementation

In the lane detection algorithm, Hough transform approach signifies the formation of the image and it is being done by taking input frame on which it applies the grayscale which allows it further to perform the Gaussian blur on the image to remove irrelevant areas and consider only the areas that are under the max and min threshold thus it allows canny detectors to detect the edges on the fast real-time environment and thus it makes the edge segment which results progressively to the hough frame which is then being tested and the resulting output frame is fed to the mechanical instrument clusters that allow the motion of car according to the model.

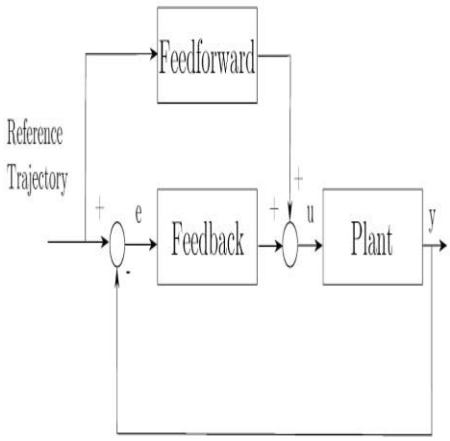

Figure 5. Output

The object classification on the model is being done by the linear support vector machine which is being fed the positive and negative data for the image classification.

ISSN: 2005-4297 IJCA

Spatial CNN proves to be more effective in different lightning and other weather conditions it also works better in case of edge detection because it can detect edges which are not clear to the view and works better on sharp curves.

7.Conclusion

Spatial CNN is a more efficient algorithm as it enables the adaptive changing parameters during the day. Overall both the methods were being successfully implementing lane detection by the implementation under python to detect road lane and also object detection is done successfully.

8.References

8.1. Implementaion of lane detection algorithm 2017 "ICEVT" by Mohamad Vicky Ghani Aziz, Ary SEtijadi Prihatmanto, Hilwadi Hindershaw

8.2. Shapiro Linda Stockman George Computer Vision, "Prentice Hall", Inc. 2001

8.3. Jiman Kim, Changing Park; 2017 IEEE conference on Computer vision & pattern recognition

8.4. Takafumi Okuyama, Jaychand Upadhyay;2018 Internation Conference on Intelligent AI

8.5. Kasyap Suresh, Varun Jyoti, Michael Drieberg; 2019 ICSCC

8.6. LiDAR presentation Portland State university

8.7. Self driving cars Towards Datascience

URL: https://towardsdatascience.com/tutorial-build-a-lane-detector-679fd8953132#bbac

Authors

S P MANIRAJ, Assistant Professor(Senior Grade),Completed B.E(Computer Science and Engineering)in Anand Institute of Higher Techology in 2004-2008,completed M.tech(Embedded System Technologies) in Anna University Tirunelveli in 2008-2010, Current pursuing Ph.D. in the school of computing in the domain of Medical Image Processing, Having working experience of 9 years in Engineering and technology.Chennai, Tamil Nadu,India.

DEEP CHAUDHARY, Student Pursing Bachelor Degree in Computer Science and Engineering in SRM Institute of Science and Technology,Ramapuram Campus,Chennai,Tamil Nadu,India.

VANKAYALA HARI DEEP, Student Pursing Bachelor Degree in Computer Science and Engineering in SRM Institute of Science and Technology,Ramapuram Campus,Chennai,Tamil Nadu,India.

VISHESH PRATAP SINGH, Student Pursing Bachelor Degree in Computer Science and Engineering in SRM Institute of Science and Technology,Ramapuram Campus,Chennai,Tamil Nadu,India.

ANANT BHARDWAJ, Student Pursing Bachelor Degree in Computer Science and Engineering in SRM Institute of Science and Technology,Ramapuram Campus,Chennai,Tamil Nadu,India.

ISSN: 2005-4297 IJCA