IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 3, March 2018

End-To-End aDriving Controls Prediction From Images Using CNN

Sumit Joshi[1], Narayan Pawar[2], Vivek Sonara[3], Sudhir Sul[4], Prof. S. A. Mulay[5]

Department of Information Technology, PVG's College of Engg. &Tech., Pune, India [1,5]

Abstract: Autonomous driving is a promising technology to improve transportation in our society. In this paper an end-to-end approach to learn how to steer a car autonomously from images is proposed. The approach proposes to map an input image to a small number of key perception indicators that directly relate to the affordance of a road/traffic state for driving. This representation provides a set of compact yet task-specific complete summary of the scene to enable a simple controller to drive a car autonomously. Convolutional Neural Networks (CNNs) trained to output steering wheel angle commands from front camera images centered on the road. For this demonstration deep Convolutional Neural Network is trained using recording of human driving and show that our model can work well to drive a car in a very diverse set of virtual environments. Results show that this approach can generalize well to real driving images.

Keywords: Image Preprocessing, Convolutional Neural Networks (CNN), Object Detection, Machine Learning, OpenCV.

I. INTRODUCTION

In most of the current self-driving technologies, the input provided to the system is based on human engineered features. This referred to as mediated perception. The control of the vehicle is based on criteria such as vehicles position and velocity and position of the car in the lane. Based on these heuristic criteria the embedded controller will optimize some cost functions to drive the vehicle safely. However, these cost function and heuristic models are hard to find and since they are based on human heuristic there is no guarantee of optimality. Another drawback is that the resulting systems are not well scalable to different scenarios. To date, most of these systems can be categorized into two major paradigms: mediated perception approaches and behavior reflex approaches.[3].

In this paper, the idea in removing every modeling assumption and engineered features and let the system learn how to accomplish the desired task using supervised learning techniques. Using this approach could lead to a robust driving capability while relaxing any modeling assumptions. The work described is a promising step in this direction, using Convolutional Neural Networks (CNNs).

The input of the system is a raw image from a camera on the autonomous vehicle, and the output is a driving control. This paper will focus on predicting steering wheel angle commands from a center camera view. In this paper, predictor will be trained using an available labeled dataset and analyze its generalization performance.[1][3]

II. LITERATURE SURVEY

Most autonomous driving set-up based on mediated perception approach. Car detection and lane detection are two key approaches of an autonomous driving system.[1] Typical algorithms output bounding boxes on detected cars and splines on detected lane markings.

2.1 Mediated perception approaches

In Mediated perception approaches Consist multiple sub-elements to identify driving-relevant objects, such as lanes, traffic signals, traffic lights, traffic vehicles, etc. The recognition results are then combined into a consistent world representation of the car's immediate surroundings. To control the car, an AI-based controller will take all of this information into consideration before making every decision to predict driving command. Since only a small portion of the detected objects are indeed relevant to driving decisions, this level of total scene understanding may add unnecessary complexity to an already difficult task.[1][12] The mediated perception estimates a high-dimensional world representation, possibly including repetitious information. The individual sub-tasks involved in mediated perception are themselves considered open research questions in computer vision. Despite the fact mediated perception encompasses the current modern up to the minute approaches for autonomous driving, most of these systems have to depend up on laser range finders, GPS, radar and very accurate maps of the environment to reliably parse objects in a scene. Basically due to this it increases the complexity and the cost of a system.

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 3, March 2018

2.2 Behavior reflex approaches

Construct a direct mapping from the sensory input to a driving action. This idea dates back to the late 1980s when used a neural network to construct a direct mapping from an image to steering angles. To develop the model, a human drives the car along the road in realtime while the system records the images and steering angles as the training data. Even if this idea is very elegant, it can struggle to deal with traffic and complicated driving maneuvers for several reasons.[1] The network's function is the same as , mapping the image directly to the steering angles, with the objective to keep the car on track. Firstly, with other cars on the road, even if the input images are identical, different human drivers may take a completely different action, which results in an ill-posed problem that is confusing when training a regressor. For example, with a car directly ahead, one may decide diferent actions such as to follow the car, to pass the car from the left, or to pass the car from the right. When all these scenarios exist in the training data, a machine learning model will have difficulty deciding what to do given almost the same images. Secondly, the decision-making for behavior reflex is too low-level. The direct mapping cannot see a bigger picture of the situation.[1] For example, from the model's view, overtaking a vehicle and switching back to the same lane are just a series of very low level decisions for turning the steering wheel in one direction and then in the other direction for some amount of time. This level of abstraction fails to capture and understand what is current scenario, and it increases the difficulty of the task unnecessarily.

Finally, because the input to the model is the whole image, the learning algorithm must conclude which parts of the image are irrelevant and which is relevant. However, the level of supervision to train a behavior reflex model, i.e. the steering angle, may be too weak to force the algorithm to learn this critical and relevant information.

III. PROPOSED WORK

This paper gives a representation that directly predicts the steering command for driving actions, instead of visually parsing the entire scene to steering angles. A proposed approach for autonomous driving – a third paradigm that falls in between mediated perception and behavior reflex.[1] This propose to learn a mapping from an image to several meaningful indicators of the road situation, along with the angle of the car relative to the road, lane markings, and the distance to cars in the both current and adjacent lanes. With this compact representation as perception output, this demonstrate that a very simple controller can then make driving decisions at a high level and drive the car smoothly.

In this paper, model is built upon the state-of-the-art deep Convolutional Neural Network (CNN) framework to extract image features for estimating controls related to autonomous driving. The reason to use Convolutional Neural Networks(CNN) model is that it just like regular Neural Networks, which does work particularly well on images. Same as Neural Networks, CNNs use hidden layers between the input and output. Key difference between two is that in case of regular Neural Networks these hidden layers are all fully connected. While working with images as input, images has the three dimensions image height, width, and the color representation. Fully connecting all layers then would lead to a hardly manageable number of connections. CNNs instead connect each neuron in a hidden layer for the most part only to a small part of all the neurons in the preceding layers, depending on the type of layer. Together with the simple controller, this model can make meaningful predictions for indicators and autonomously drive a car in different tracks, under different traffic conditions and lane configurations. Meanwhile, it takes advantage of simpler structure than the typical mediated perception approach. Proposed approach offers a task-specific, compact description for scene understanding in autonomous driving. Fig. 1 shows system architecture.

In this paper, we used images from the open source dataset provided by Udacity.[8] In the training phase, we decided to use the center camera images (first person driving view) and the corresponding ground truth values for steering angle prediction. For obstacle detection we decided to use trained model of Tensorflow SSD Mobilenet .

In the testing phase, at each time step, the trained model takes a driving scene frame as input and estimates the indicators for driving e.g. steering angle, the distance to the preceding cars, number of vehicles detected. A driving controller processes the indicators and computes the steering and acceleration/brake commands. The driving commands are then used back to drive the host car.

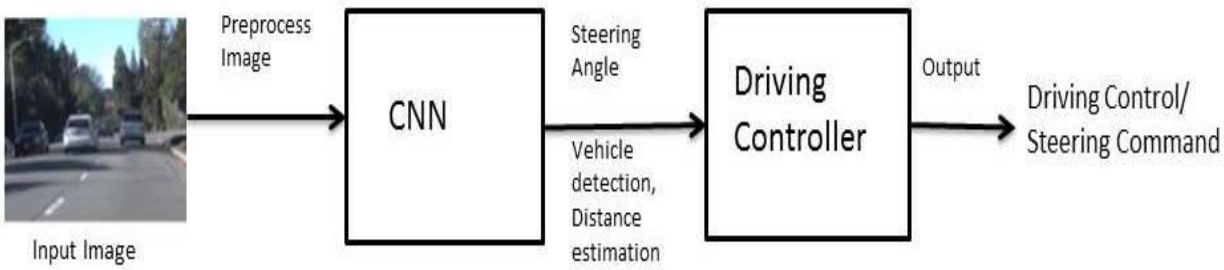

Fig 1: System Architecture

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 3, March 2018

Fig 2: Block diagram representing the input output behavior of our system

III. METHODOLOGY

This paper proposes the driving controller logic that predicts the driving controls on basis of distance to the preceding cars, number of vehicles detected, steering angle and object detection.

4.1 Steering Angle Prediction

To train model for prediction of steering angle, training data is images from the open source dataset provided by Udacity[8]. While the dataset may include auxiliary information, you may only use camera imagery and steering wheel angle information to train. The topic containing steering wheel information will not be present in the testing dataset. In particular, dataset comprise images from three cameras at the front of the car at 20 frames per second with are solution of 640×480 pixels, as well as the steering wheel angles and further information relevant for car controls at a higher rate. After extracting the data from the available dataset some pre-processing steps to prepare it for our computations are performed. First, we interpolated the steering angles to the timestamps of the front camera images. Next, we created new folders with center camera images only and their associated steering angles, and renamed the files to make them more easily access able. We decided to use center camera images only. Different tracks and different traffic cars are used to collect training data with different lane configurations. The captured frames are converted to gray scale image and it resized to 256 x 192 resolution using opencv functions. This data will be used for training a model to build CNN. The CNN architecture (Fig 3) for predicting steering angle consists of three 5x5 Convolutional layers followed by two 3x3 Convolutional layers with activation function Exponential linear unit (ELU). Dropout function is used to prevent overfitting by setting neurons to zero with a specified probability. Three fully connected layers are used having 100,50,10 neurons respectively and last Fully connected layer with 1 neuron which gives output of model.[4] The convolution layers are meant to handle feature engineering. The fully connected layers are for predicting the steering angle. ELU function takes care of the Vanishing gradient problem. ELU helps to push the mean activation of neurons closer to zero which is beneficial for learning and it helps to learn representations that are more robust to noise. Tensorflow is used to build model having learing rate 1.0e-4.

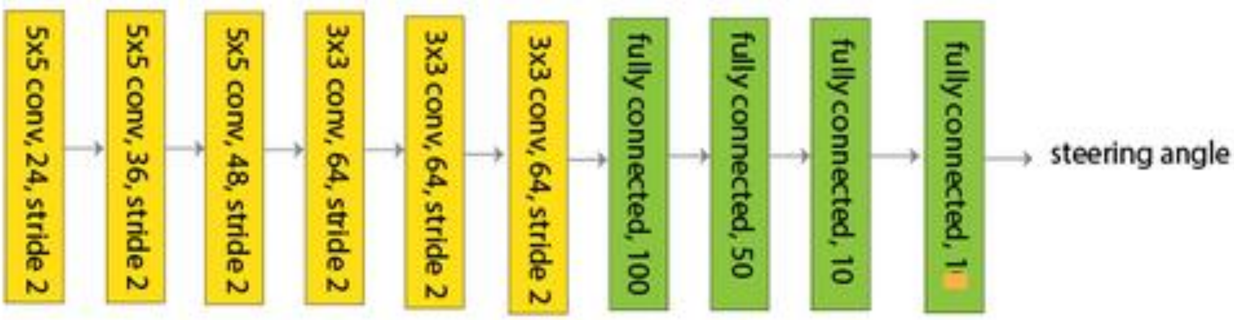

Fig 3: Architecture of CNN model Steering Angle Prediction

4.2 Vehicle Detection and Distance Estimation

There are many ways we could use object detection in environment, but one of the most obvious is to detect other vehicles and determine whether or not they are too close. Distance information for potential obstacles in the path of a vehicle is critical for intelligent automotive systems in order to avoid collision. In this paper for vehicle and object detection Tenserflow object Detection API use with pre trained SSD MobileNet model of Tensorflow. MobileNets are small, low-latency, low-power models parameterized to meet the resource constraints of a variety of use cases. They can be built upon for classification, detection, embeddings and segmentation similar to how other popular large scale models, such as Inception, are used. MobileNets can be run efficiently on mobile devices with TensorFlow Mobile. For tracking too close vehicle in frame, we're interested in some specific classes of objects.

However, that, to determine distance between vehicle, you need to know the object's size before-hand. To estimate the distance the width of the detected object is used. This can be done by asking how many pixels-wide the object is and adding some granularity to it we can get approximate distance between vehicle.[10] The task is just to see that, even if the object is close, it might be far enough to the side(maybe in other lane) that it's not a likely collision issue. But for example, if car is going through an intersection, and there's a car coming across, a reasonable person would know that, despite the car not being in front, that vehicle is a collision risk. With this, it is possible to track a relative distance to vehicles around car, and determine if they're too close and signal a warning and/or even perform some decision making like stop the car or slow down the speed of the car. By using the value of steering angle along with object and vehicle detection, relative distance between cars are used for decision making to output driving command for vehicle.

The predicted steering angle is used to decide the movement of steering wheel. The object/vehicle detection helps in understanding the situation and vehicle distance estimation helps to signal warning or even in decision making whether to stop car or slow down the speed if vehicle in front of car comes too close. Using this data driving controller can predict the driving command effectively by considering lane availability, relative distance between cars, and steering angle and this command will be sent back to the host car

4.3 Limitations

4.3.1 Input image to CNN should be clear and it should not be blurred.

4.3.2 Proposed system makes wrong predictions when road conditions are bad e.g. roads with potholes that makes capturing image difficult.

4.3.3 The proposed system works fine in sunny day. It’s difficult to make predictions in dark or in shadow where picture is not clear.

V. CONCLUSION

In this paper, the challenging task of end to end vehicle control in terms of both steering angle and vehicle distance estimation is proposed through a noble autonomous driving paradigm. The model takes front-view camera images / frames as input and gives driving control command.

This representation leverages a deep CNN architecture to estimate the affordance for driving actions instead of parsing entire scenes (mediated perception approaches), or blindly mapping an image directly to driving commands (behavior reflex approaches).

REFERENCES

[1] Chen yi Chen, AriSeff, Alain Kornhauser, and Jianxiong Xiao. Deep driving: Learning affordance for direct perception in autonomous driving. In Proceedings of the IEEE International Conference on Computer Vision, pages 2722–2730, 2015.
[2] Fei-Fei Li, Andrej Karpathy, and J Johnson. CS231n: Convolutional Neural Networks for Visual Recognition. http: //cs231n.github.io/convolutional-networks/, 2015.
[3] Jan Felix Heyse and MaximeBouton , End-to-end Driving Controls Predictions from Images, CS229, Final Report
[4] End to End Learning for Self-Driving Cars - Nvidia https://arxiv.org/abs/1604.0731
[5] P. Lenz, J. Ziegler, A. Geiger, and M. Roser.Sparse scene flow segmentation for moving object detection in urban environments. In Intelligent Vehicles Symposium (IV), 2011 IEEE, pages 926–932. IEEE, 2011.
[6] U. Muller, J. Ben, E. Cosatto, B. Flepp, and Y. L. Cun. Off road obstacle avoidance through end-to-end learning. In Advances in neural information processing systems, pages739– 746, 2005. 3
[7] An Open Source Self-Driving Car https://www.udacity.com/self-driving-car
[8] Udacity open source self driving car dataset https://github.com/udacity/self-driving-car
[9] Advanced lane and vehicle detection using computer vision https://github.com/JunshengFu/vehicle-detection
[10] https://pythonprogramming.net/detecting-distances-self-driving-car/
[11] MIT 6.S094: Deep Learning for Self-Driving Cars https://selfdrivingcars.mit.edu/
[12] S. Ullman. Against direct perception. Behavioral and Brain Sciences, 3(03):373–381, 1980.1