ISSN No:-2456-2165

Behavior Cloning for Self Driving Cars using Attention Models

M Shvejan Shashank
Dept. of Computer Science Engineering
Sreenidhi Institute of Science and Technology
Hyderabad,India

Saikrishna Prathapaneni
Dept. of Electronics and Communication Engineering
Sreenidhi Institute of Science and Technology
Hyderabad,India

M. Anil
Dept. of Computer Science Engineering
Sreenidhi Institute of Science and Technology
Hyderabad,India

N Thanuja Sri
Dept. of Computer Science Engineering
Sreenidhi Institute of Science and Technology
Hyderabad,India

Mohammed Owais
Dept. of Computer Science Engineering
Sreenidhi Institute of Science and Technology
Hyderabad,India

B Saikumar
Dept. of Computer Science Engineering
Sreenidhi Institute of Science and Technology
Hyderabad, India

Abstract:- Vision transformers started making waves in deep learning by replacing the typical convolutional neural networks (CNN) in tasks like image classification, Object detection, segmentation etc. the implementation of vision transformers can be further extended to autonomous cars. As it was proven that pure transformer architecture can outperform CNNs when trained over a large dataset by using comparatively less computational resources [1]. These vision transformers can be implemented in self-driving cars to calculate the optimal steering angle by capturing images of the surroundings. The vision transformers can take advantage of the attention mechanism to focus on the most important things in the image like road lanes, other vehicles, traffic signs, etc. and produce better results compared to CNN models. The paper discusses about implementation behavior cloning using vision transformers in a self-driving car which is simulated in a Udacity self-drive car simulator.

Keywords:- *Transformers, Self-Driving Vehicles, Vision Transformers, Convolutional Neural Networks CNN, Behavior Cloning.*

I. INTRODUCTION

Recent developments in self driving cars use regional networks, state of the art object detection models for getting the inference from the environment. This proved to be an efficient method for analyzing the surroundings and take control of the vehicle in challenging places like roads. These autonomous decision-making systems not only need vision but also sensing systems to take an informed decision, lidar, GPS, infrared systems, etc are most common sensor systems that can be seen in recent day self-driving cars. Vision systems and underlying algorithms that process imagery data has taken the pace in research and a lot of development is needed in this area. In this paper a new class of decision-making systems which takes the advantage of vision transformers is implemented with the data that is gathered from a simulator for training the underlying vision transformer model for behavioral cloning.

After showing outstanding performance in natural language processing (NLP), the Self-attention-based architectures are have become the industry standards over the last couple of years. Recently this architecture is also introduced in the computer vision field [1] where the transformer architecture or the Vision-Transformers, in particular, have shown significant improvement in terms of computational efficiency and scalability over the Convolutional Neural networks which were dominant in the computer vision field earlier. The Vision -Transformers paper has shown how a standard transformer with fewer modifications can outperform the Convolutional neural networks when trained over large data. The paper has proposed a technique where an image is split into multiple patches and added a sequence of linear embeddings to these patches to treat each patch as a separate word in a typical NLP application. Fig 1 shows the architecture of vision transformer followed in [1].

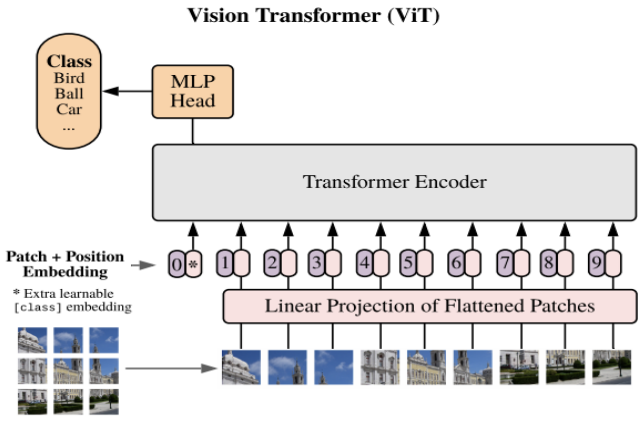

Fig1: depicts the architecture of vision transformer [1]

ISSN No:-2456-2165

While this research paper has tested the model only on classification problems, we have tried to extend the application of vision transformers by trying to use the Vision Transformer [1] in pattern recognition and behavioral cloning applications. From the research paper End to End Learning for Self-Driving Cars [2] it was proven that CNN architecture can achieve behavioral cloning with pattern recognition. We have experimented by replacing the CNN model in the End-to-End Learning for Self-Driving Cars [2] with the Vision Transformers [1]. The model is trained using Udacity's self-driving car simulator, where we have captured the images of the road, and the steering angle data of the car when a human is manually driving the car. This data is used to train the Vision Transformer [1] to determine the correct steering angle when a picture of the road is given.

II. LITERATURE REVIEW

Models for self-driving systems have been developing for decades now, First, Dickmanns et al. turned a Mercedes-Benz van into an autonomous car that drove itself up to 96 km/h on more than 20 km road without traffic based on dynamic vision [3], till the behavior cloning that is done by models in CNN which proved to be having least errors in mimicking human behavior in driving task [2]. The development has brought up new methodologies that could potentially revolutionize Autonomous self-driving systems.

CNN's have been used for most of the tasks that involved vision imagery that is captured through the camera systems that are present from the VGG [4] which is used in image classification tasks that are used for rudimentary tasks in self-driving cars. Much focus is developed on object segmentation and detection systems that have much more potential than the image classification models, two such models are RCNN [6] and YOLO [5] models, both are considered as the state of art detection models for object detection and localization in imagery data. Though these models have been good in detection tasks much more efficient methods had to be implemented for getting inference from the imagery data such as segmentation, paper by Ronneberger [7] had the UNET model developed for the biomedical image segmentation but has been proved to be efficient for the object segmentation in various aspects of road scenes, etc. Along with the biomedical image segmentation, all these models have been efficient in classifying, detecting, and segmenting for the scenes on the roads, which is being shown in fig 2 but are computationally intensive due to the huge weights they carry.

A lightweight highly efficient models are to be implemented for the detection tasks without compromising on accuracies and MAPs for any self-driving or autonomous systems for getting considerable efficiencies. The transformer networks were first introduced for developing [8] natural language processing but have been efficient in image processing tasks well [1].

Fig 2: image shows segmentation of road scene in an autonomous driving car.

In this paper behavioral cloning of an autonomous car is considered and data is considered for training is done based on the images that are captured in a simulated environment, and corresponding labels of steering angle is considered for training the images, taking the advantage of the vision transformers that has capabilities of giving high accuracies when trained on large datasets. The paper got divided into six sections. firstly,

III. METHODOLOGY

- *Data collection:*

The collection of data is done through a self-driving car simulator. The main reason for using the simulator instead of a real-life car is to reduce the cost of damages and ease of data collection, while the data which is used to train the model is completely taken from the simulator, the same practices can be replicated in real-world scenarios as well. We have used Udacity self-driving car The car used in the simulator has three cameras at the front of the car namely at the front left, front center, and front right area of the car. these cameras take snap shots of the road and the surroundings of the car while driving on the road. Along with the imagery data, the simulator also captures the seed of the car, the throttle, and the steering of the car too. The car is driven in different environments created in the simulator to get generalized training and testing data. fig3 and fig4 shows the scene that is captured from the center and right cameras. Fig 5 shows the positioning of cameras on the simulated car.

Fig 3: shows the images that are captured from the center camera in simulator

ISSN No:-2456-2165

Fig 4: sample training images used for training the transformer

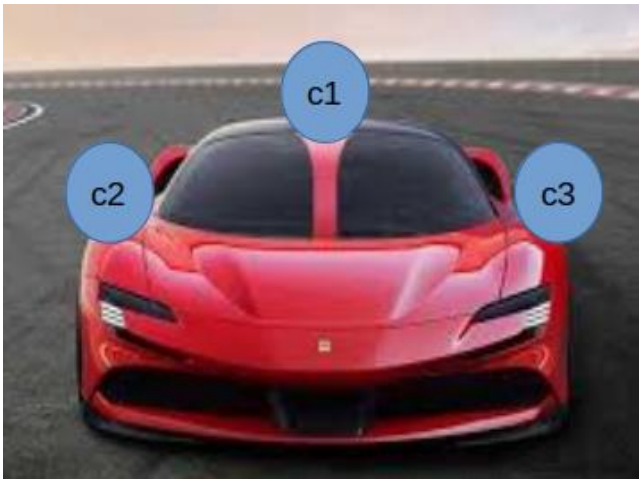

Fig 5: camera positioning on the car, c1 c2 c3 are the cameras.

➢ *Preprocessing:*

The simulator reads and stores the steering angles of the car in a float value in which all the negative values indicate the steering is turned towards the left and all the positive values indicate the steering turned towards the right, and 0 means the steering is not turned in any direction. the first task is to make sure that the data collected has an equal number of left and right steering angle data, this is to eliminate any biases on the model while training.

The second observation that can be made from the data after plotting a histogram of the steering angles data is that there are significantly more '0' steering angle values than any other values, this is mainly because most of the time, there is no need for rotating the steering frequently to stay on the road because most of the roads are usually straight and hence a maximum number of '0' steering angles can be observed. out of more than 9000 data points collected from the simulator, more than 7000 data points have a '0' steering angle value. This can be a problem for the model as it will naturally learn to predict only the '0' value as output and get very high accuracy. to resolve this, we have to reduce the number of data points that have a '0' steering angle. Since we need the model to predict a '0' steering most of the time but only turn when there is a turn or curve in the road, we will still have to keep the '0' steering angle data points high enough. The 7000 data points are filtered and cut short to just 1000 data points. The next step is to eliminate the sky, clouds, and other unwanted data in the image which can be done simply by cropping out the top portion of the image. Further, other common data annotation techniques like random zoom, gaussian blur, brightness, and normalization. the image is then converted from RGB to YUV color scale. The next step is to divide the images into patches of equal shape. First, the image is converted into a square shape, and then, the square-shaped image is divided into 9 equal patches. next, each patch is embedded with its positional data, this is to make sure that the positional data of the image is not lost while training the model with individual patches.

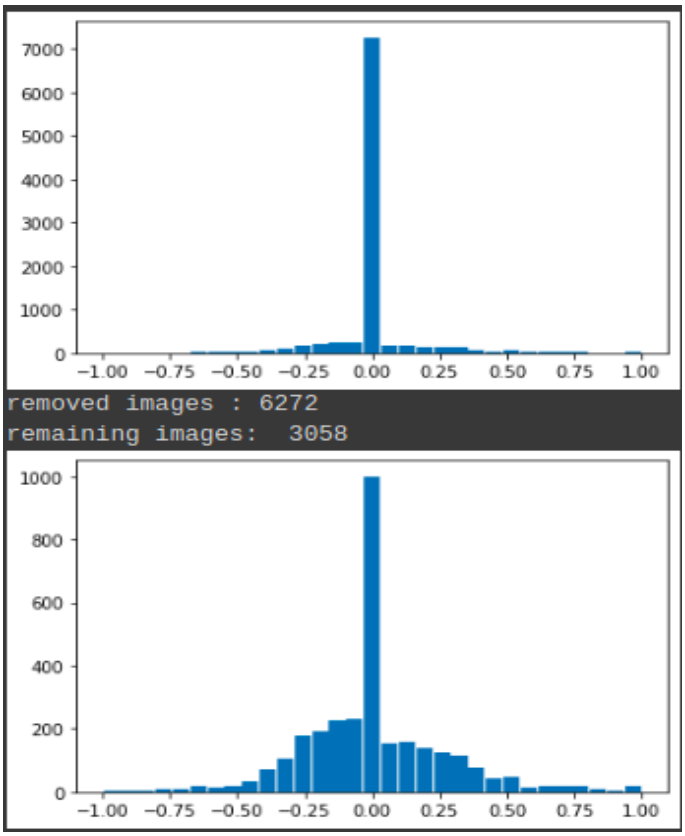

Fig 6: distribution of images for corresponding labels of steering angle.

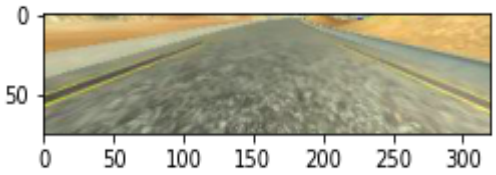

Fig 7: cropped image considered for training

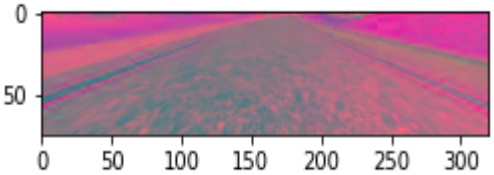

Fig 8: image converted to YUV color scale

ISSN No:-2456-2165

Fig 9: image converted to 9 X 9 patches for training with each patch size of 60 by 60

➢ *Model:*

The model used in training and deployment is ViT(vision transformer) and the architecture is depicted in the fig 10. In total 8 of single encoder of ViT models are stacked together, This had proven to perform with greater accuracies.

The model used for training is the ViT(vision transformer) model from the research paper [1]. The basic idea of training the model is to take the patches of an image as input and predict the accurate steering angle that the car should turn to stay on the road. The ViT model is used in its purest form with slight modifications to the last layer where the SoftMax layer is replaced with a normal dense layer to predict the steering angle. A total of 8 transformer encoder layers are used in the model. Fig 11 shows the architecture of transformer encoder.

Fig 10 shows the validation and training loss plotted against number of epochs trained.

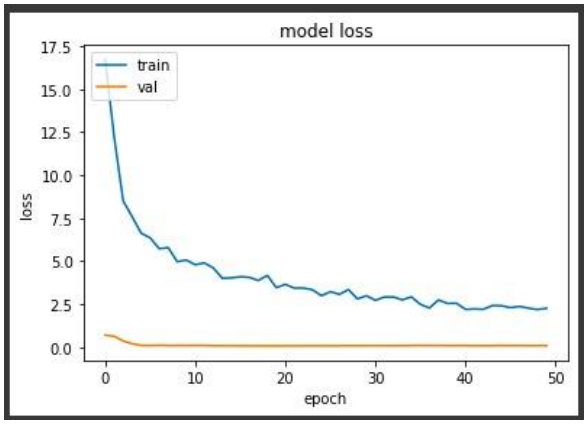

Fig 10: training and validation losses are plotted against epochs

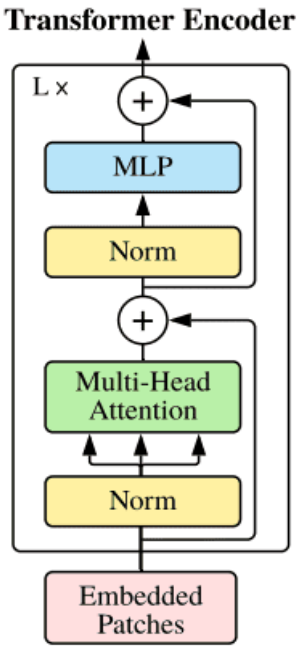

Fig 11: architecture of single transformer encoder

➢ *Deployment:*

The model is deployed on a flask server and it is connected to the car simulator using socketIO to establish a real-time connection with the car. The images which are captured from the car are sent to the server over socketIO and the images are sent through the trained model to predict the steering angle. The speed limit of the car is also monitored on the server to make sure the car stays under the speed limits. In fig 12 the cycle is depicted and the process of how images are loaded and python server is established with the simulator.

ISSN No:-2456-2165

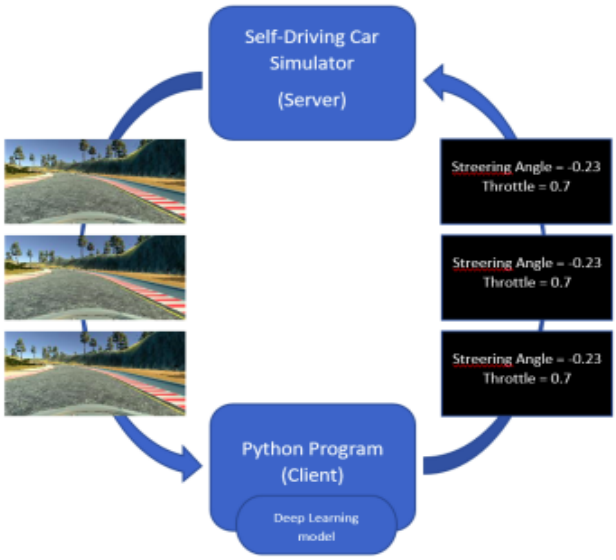

Fig 12: shows the cycle of deployment

IV. RESULTS

The socket IO connection between the car simulator and the flask server is established, and the images are streamed over socket IO to the flask server and the server successfully processes the images to predict the steering angle with an accuracy of 0.06. The speed of the car is also monitored by the server to make sure the car stays within the specified speed limits. The model is trained for 500 epochs over 2000 images and a validation loss of 0.6 .

V. CONCLUSION

The traditional methodology of using CNN to achieve behavior cloning is replaced by the attention models and Vision transformers. While we have trained the model with less than 2000 images, the results can be much more prominent when trained over 10Million plus image to which will show better training results with less computational resources over CNNs. we are This shows the applications of Vision Transformers can be extended to other sectors as well and could replace CNNs very soon in the future.

REFERENCES

[1]. Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." arXiv preprint arXiv:2010.11929 (2020).
[2]. Sumanth, Uppala, et al. "Enhanced Behavioral Cloning-Based Self-driving Car Using Transfer Learning." Data Management, Analytics and Innovation. Springer, Singapore, 2022. 185-198.
[3]. E. D. Dickmanns and V. Graefe, “Applications of dynamic monocular machine vision,” Machine Vision and Applications, vol. 1, pp. 241–261, 1988.
[4]. Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).
[5]. Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[6]. Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.
[7]. Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015.
[8]. Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).