UDACITY

PROJECT

Use Deep Learning to Clone Driving Behavior

A part of the Self-Driving Car Program

PROJECT REVIEW

CODE REVIEW

NOTES

SHARE YOUR ACCOMPLISHMENT!

Meets Specifications

Well documented code, modular structure, a very detailed and to the point writeup. It's amazing. You should be happy because you have done a great job indeed. It's impressive.
It's not that often to read a good writeup like your. Good job.
There is one typo in the document where you are talking about horizontal flipping. The heading is using the word "vertical". You might want to change that for your future readers.
Overall, it is great and that is the reason I am not suggesting a resubmission. But, do not forget to change that in your code repository on profile page. Congratulations and good luck :)

Required Files

The submission includes a model.py file, drive.py, model.h5 a writeup report and video.mp4.

Great job. Thanks for naming the files correctly. Also, I am amazed by your detailed writeup and demonstration video specially the last part where the recovery of car back on track is shown. Awesome !!!

Quality of Code

The model provided can be used to successfully operate the simulation.

Well done for implementing the neural network that controls the car in the simulator. The drive through the path 1 is relatively very smooth.

The code in `model.py` uses a Python generator, if needed, to generate data for training rather than storing the training data in memory. The `model.py` code is clearly organized and comments are included where needed.

The code is well formatted and appropriately commented which makes it easy to read the code and understand the algorithm. Good job in generating additional data using python generator and collecting the recovery data. Well done !!!

Model Architecture and Training Strategy

The neural network uses convolution layers with appropriate filter sizes. Layers exist to introduce nonlinearity into the model. The data is normalized in the model.

Great Job. Model uses convolution layers, batch normalization and dropout. Batch Normalization makes the model very interesting and its contribution is evident in the result. I am impressed by the little changes made in Nvidia's model to incorporate dropouts.
Great job explaining the layers with different filter sizes.

Train/validation/test splits have been used, and the model uses dropout layers or other methods to reduce overfitting.

An awesome job to monitor and avoid overfitting. The training of the model includes also validation set that allows monitoring the performance of the model and avoids overfitting. Batch normalization and drop out techniques are used to avoid overfitting. Great Job.

Learning rate parameters are chosen with explanation, or an Adam optimizer is used.

Adam Optimizer is used. You might be interested in reading some other reasons why it is a better choice. The tf.train.AdamOptimizer uses Kingma and Ba's Adam algorithm to control the learning rate. Adam offers several advantages over the simple tf.train.GradientDescentOptimizer. Foremost is that it uses moving averages of the parameters (momentum). Simply put, this enables Adam to use a larger effective step size, and the algorithm will converge to this step size without fine tuning. The main down side of the algorithm is that Adam requires more computation to be performed for each parameter in each training step (to maintain the moving averages and variance, and calculate the scaled gradient); and more state to be retained for each parameter (approximately tripling the size of the model to store the average and variance for each parameter). A simple tf.train.GradientDescentOptimizer could equally be used, but would require more hyperparameter tuning before it would converge as quickly

Training data has been chosen to induce the desired behavior in the simulation (i.e. keeping the car on the track).

Nice Job of doing more with less data. Training data was properly chosen to induce the desired driving behavior in the simulation. The training set includes maneuvers that keep the car away from the lane line and make sure that the car stays on the road throughout the simulation. Collecting the recovery data is important and your collection of additional 11,000 images is impressive as it is not a lot of data and it does the job very well.

Architecture and Training Documentation

The README thoroughly discusses the approach taken for deriving and designing a model architecture fit for solving the given problem.

Very good job explaining your approach for solving the given problem. It's impressive.

The README provides sufficient details of the characteristics and qualities of the architecture, such as the type of model used, the number of layers, the size of each layer. Visualizations emphasizing particular qualities of the architecture are encouraged.

The architecture of the implemented model is perfectly detailed and illustrated in the README. Great job here. I cannot tell you how great reading your document was. Precise and very articulate. Well done.

The README describes how the model was trained and what the characteristics of the dataset are. Information such as how the dataset was generated and examples of images from the dataset must be included.

Great job once again. The training and dataset preprocessing are properly described in the README. Use of images from all three cameras is properly explained.
One suggestion to include in your write up would be to right about why you did not train more then four epochs. You can reason that model was overfitting and more epochs would result in car leaving the track. Longer or shorter training might cause overfitting/underfitting of the model to the training set. Therefore, it is important to find the right balance Instead. You can explain that by identifying the number of the epoch that provides the maximum performance (or convergence) on the validation set.

Simulation

No tire may leave the drivable portion of the track surface. The car may not pop up onto ledges or roll over any surfaces that would otherwise be considered unsafe (if humans were in the vehicle).

Drive through the track is very smooth. Great Job. Congratulations :)

DOWNLOAD PROJECT

RETURN TO PATH

Student FAQ