Behavioral Cloning

Project Writeup

Model Architecture and Training Strategy

1. Solution Design Approach

Firstly I tried Lenet architecture. Car was not passing the bridge. I tried to modify the training set as proposed in lessons with no success.

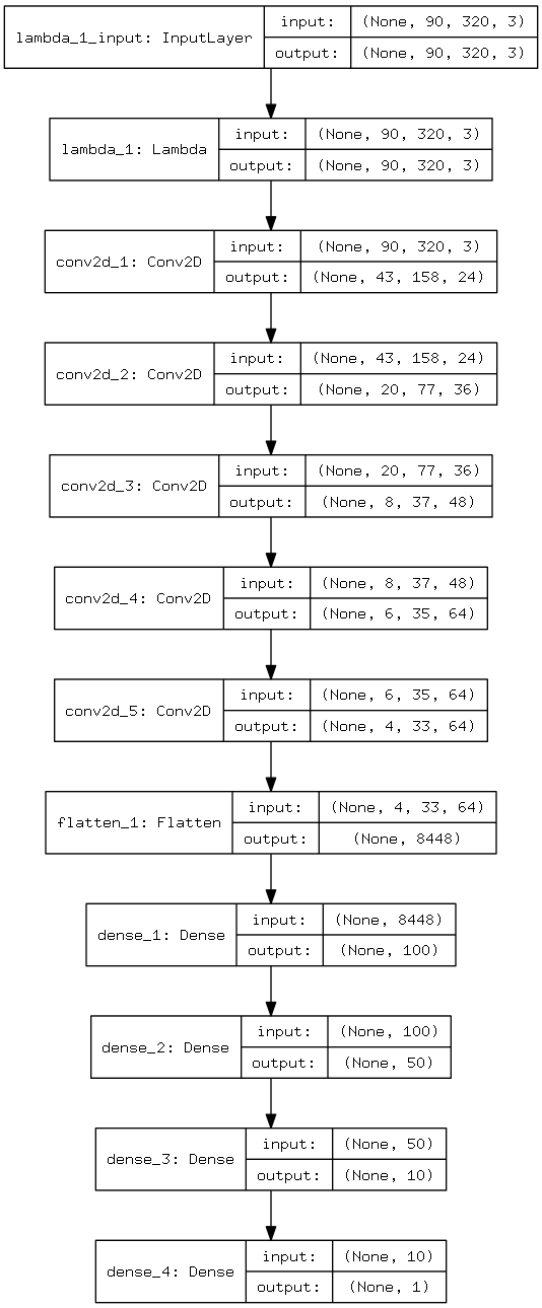
After this trial I used a convolution neural network model similar to the NVIDIA model proposed in the lesson.

In order to gauge how well the model was working, I split my image and steering angle data into a training and validation set. I found that training error and validation error was close.

The final step was to run the simulator to see how well the car was driving around track one.

At the end of the process, the vehicle is able to drive autonomously around the track without leaving the road. I tried higher speed by changing drive.py. Car was driving OK at 15 and 20 mph.

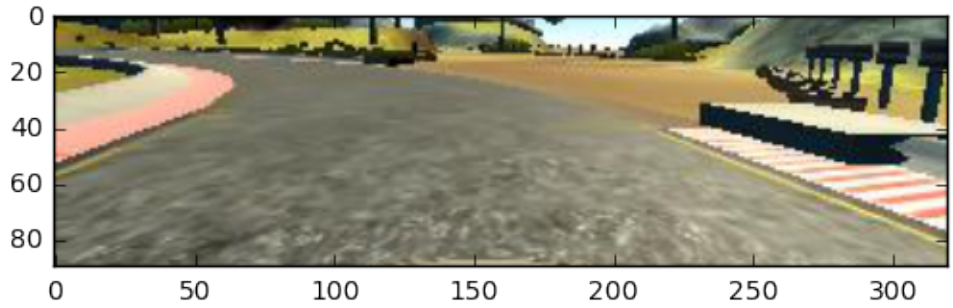
2. Final Model Architecture



3. Creation of the Training Set & Training Process

I tried to record training set as suggested. But it was not easy to drive the car. Not to lose too much time I decided to use the data provided.

I employed center, left and right images as suggested. I also added flipped images. While importing the data images cropped from bottom and top, as seen in the following picture.



To be able to use the cropped images drive.py at line 64 modified also.

I finally randomly shuffled the data set and put 20% of the data into a validation set.

I used this training data for training the model. The validation set helped determine if the model was over or under fitting. The ideal number of epochs was 5. After 5 epochs improvement was not high. I used an Adam optimizer so that manually training the learning rate wasn't necessary.