The project includes the following files:

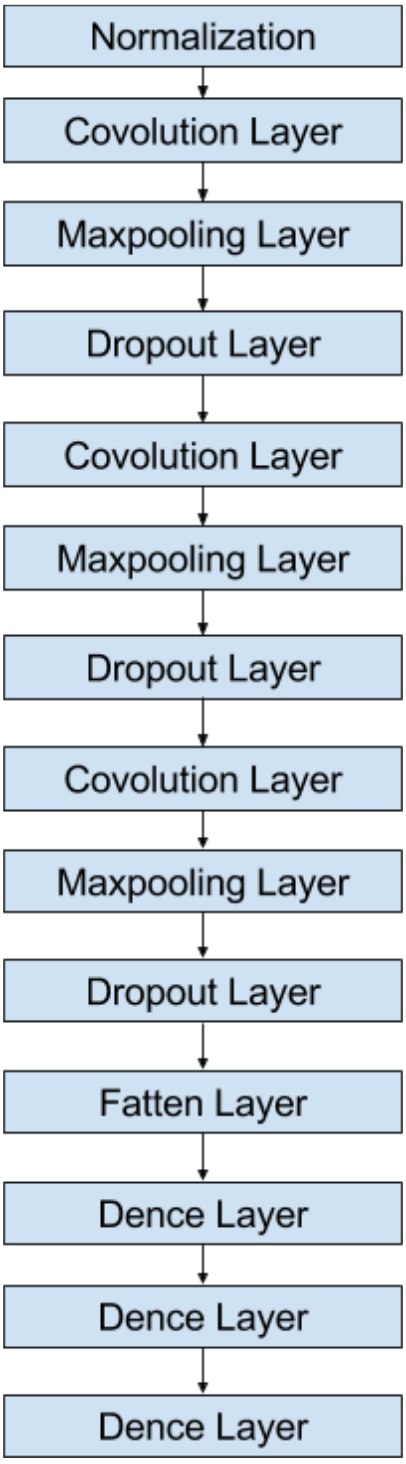
- model.py containing the script to create and train the model
- drive.py for driving the car in autonomous mode
- model.h5 containing a trained convolution neural network
- writeup_report_P3.pdf summarizing the results

Model Architecture

At the beginning I tried implementing NVidia model. It took about 15mins for each echo. The model was trained for 5 echos. However the outcome was not upto the requirement. The car was able to drive 10 seconds and it got out of the track.

Then I tired different models with the help of other students. The final model was has convolution neural network with 3x3 filter sizes and depths between 32, 64 and 128. It also has RELU layers to break the linearity and dropout layers to reduce overfitting.

The images were resized to 64x64 before feeding into this model. This gave a significant improvement in during the training with reduced time and increased accuracy.


Normalization
Covolution Layer
Maxpooling Layer
Dropout Layer
Covolution Layer
Maxpooling Layer
Dropout Layer
Covolution Layer
Maxpooling Layer
Dropout Layer
Fatten Layer
Dence Layer
Dence Layer
Dence Layer

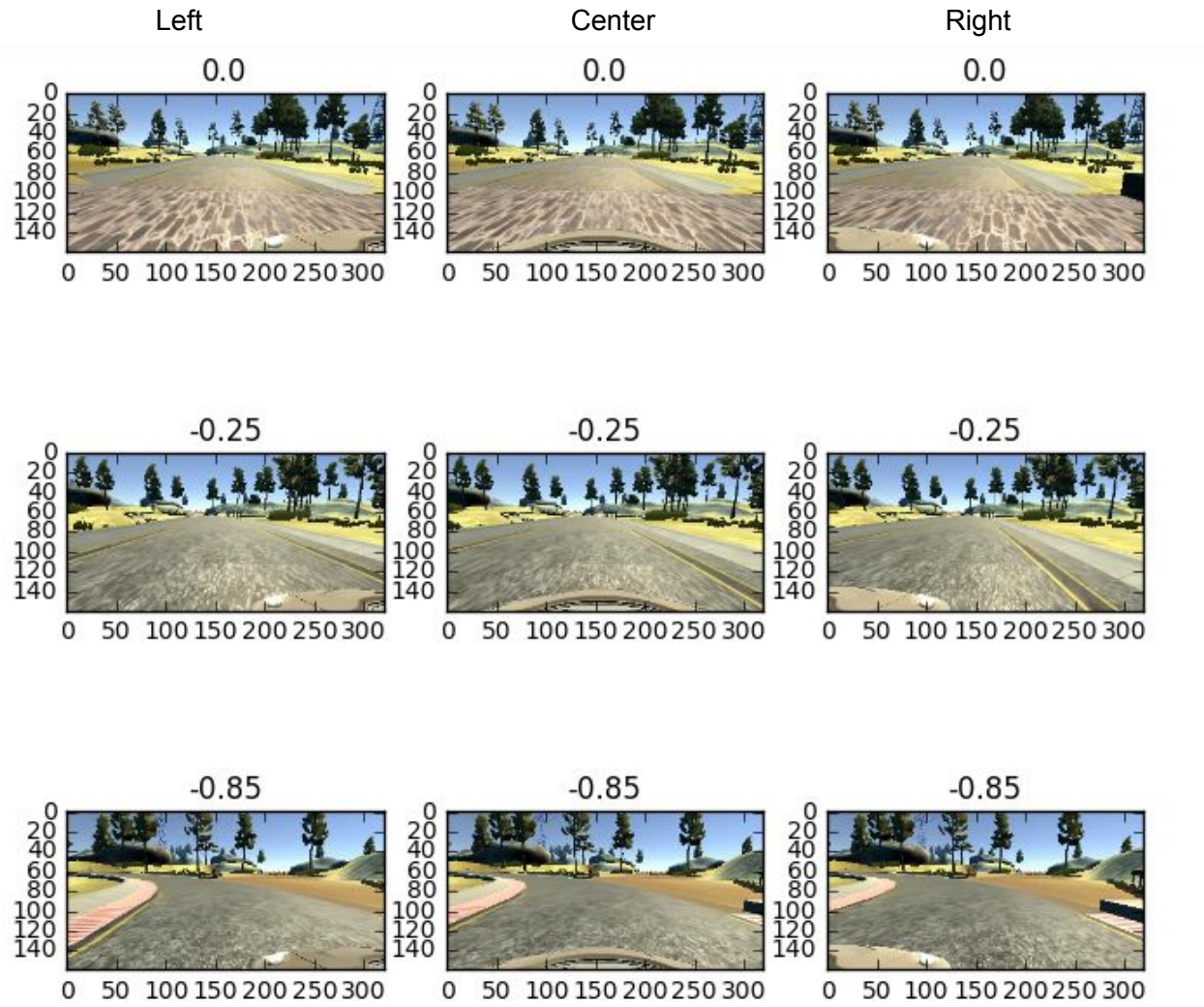
Traing

At the beginning I used the data set provided by Udacity. However it was found that the car in making deliberate right turn and get into a side track (after passing the bridge).

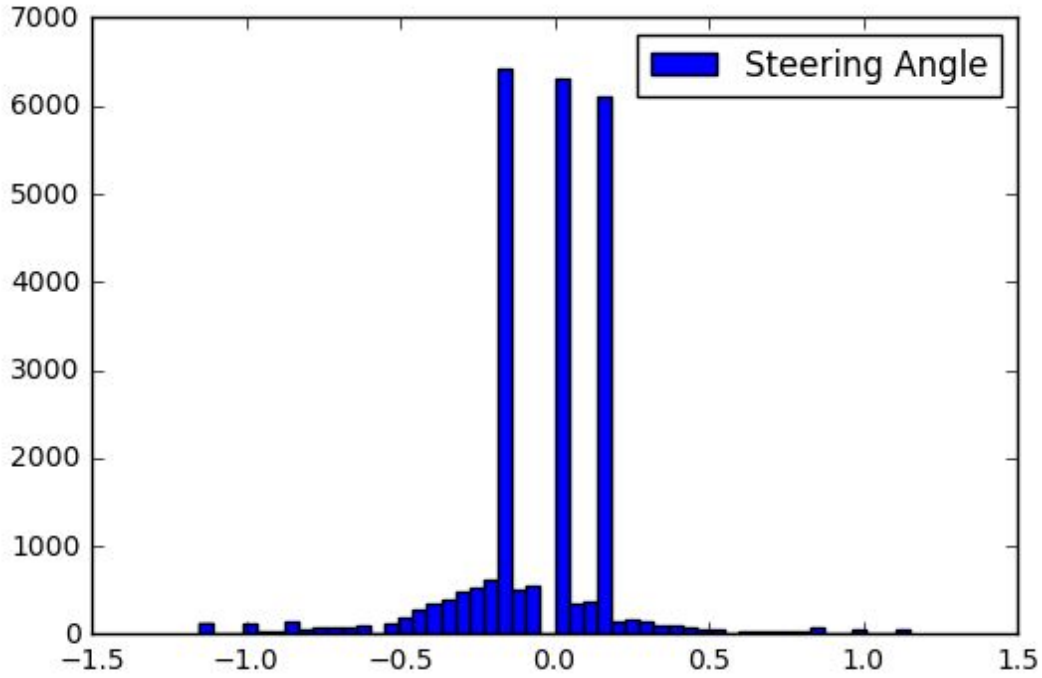
Augmenting new training data set for that section did not solve the problem. Therefore a new set of data by driving several rounds around the circuit was generated and trained model for 20 ephos.

This data set consist of 25686 images. Out of that 80% was used to train the model and 20% was used to validate the model.

Here are some of the trained images



Distribution of Steering Angles in training images



After 20 ephos of training, the validation error was reduced to 4%.

This new training data set was able to train the model to make the car drive autonomously around the circuit .

Areas to be improved

The autonomous driving tend go very close to left side of the road. This may mainly due to because the training data were mainly towards left side.

Since I was not good in playing games and I was using arrow keys to guide the training car, there were many abrupt manuring in the training data set. This could be smoothen out by getting more experience about the track and using a game steering system to control the vehicle.

The model Architecture could be improved with adding more layers to the model.