Behavioral Cloning

1. Has an appropriate model architecture been employed for the task?

The Nvidia Autonomous Model has be used to train the Neural Network. It uses three 5x5 Convolution layers, followed by two 3x3 Convolution layers, 3 Fully-Connected layers and 1 Output layer. ReLU activation function is used to introduce non-linearity in the model. The data is normalized using a lambda layer with the formula: x / 255.0 – 0.5.

2. Has an attempt been made to reduce overfitting of the model?

The model has been split using a train-test split of 0.2. It was found that too many dropout layers leads to bad test performance so a single dropout of 0.75 is used after the Flatten() layer to reduce overfitting.

3. Have the model parameters been tuned appropriately?

The model hyper-parameters have been selected for best results. An Adam optimizer is used to fit the model. The default learning rate = 0.001 fits the model good enough over 1 epoch. Also, a batch size of 128 is used to train the model.

4. Is the training data chosen appropriately?

The training data has been chosen to generalize the model on the first course. The course is left turn biased going anti-clockwise so data is collected by driving the vehicle clockwise which has a right turn bias. The following runs were used to generate the data:

a. 2 laps anticlockwise with center lane driving
b. 2 laps clockwise with center lane driving
c. 1 lap anticlockwise with lane recovery driving
d. 1 lap clockwise with lane recovery driving
e. 5 laps anticlockwise around the curves
f. 5 laps clockwise around the curves

Illustration 1: Center lane Anti-Clockwise

Illustration 2: Center Lane Clockwise

5. Is the solution design documented?

After generating the data, the model was initially trained with just a single Dense layer to check whether the model generates an output to a given input. Next, two previously used Architectures: LeNet and Traffic Sign Classifier were tested on the images to check the performance. However, the model doesn't perform well on both LeNet and Traffic Sign Classifier. It is documented that the Nvidia Architecture performs very well on the Autonomous Vehicles. The architecture was initially chosen with Dropout Layers after each main layer. However, the model wasn't able to traverse the course with such an architecture. In the next, update the dropouts were removed and kernel regularizers were used to prevent overfitting. However the model drove very close to the left lane and could traverse the curves. The model was then trained without any dropouts or kernel regularizers and the model performed well over the entire track except one right curve. So, a single dropout was introduced after the flatten layer which generated the desired performance.

6. Is the model architecture documented?

The following table illustrates the Model Architecture and Layers:

Layer	Input	Output
Lambda Layer Normalization	*160x320x3*	*160x320x3*
Cropping Layer	*160x320x3*	*65x320x3*
5x5 Convolution Layer	*65x320x3*	*31x158x24*
5x5 Convolution Layer	*31x158x24*	*14x77x36*
5x5 Convolution Layer	*14x77x36*	*5x37x48*
3x3 Convolution Layer	*5x37x48*	*3x35x64*
3x3 Convolution Layer	*3x35x64*	*1x33x64*
Flatten Layer	*1x33x64*	*2122*
Dropout Layer	*2122*	*2122*
Dense Layer	*2122*	*100*
Dense Layer	*100*	*50*
Dense Layer	*50*	*10*
Output Layer	*10*	*1*

7. Is the creation of the training dataset and training process documented?

The model was trained using the Nvidia Autonomous Vehicle Architecture. The architecture is as defined above. The model was trained using the Adam optimizer and the mean squared error is used as the loss function. The hyper-parameters used were: learning rate = 0.001, batch_size = 128, epochs = 1. The dataset was generated as described in Question 2. and is split into training and validation sets in a 4:1 ratio. The images are loaded in batches using a generator. The generator reads the dataset, loads the left, right, center images and augments the images. Both the original and augmented images are used to train the model. The image augmentation changes the brightness by a random amount and adds a shadow occlusion as in the figure below.

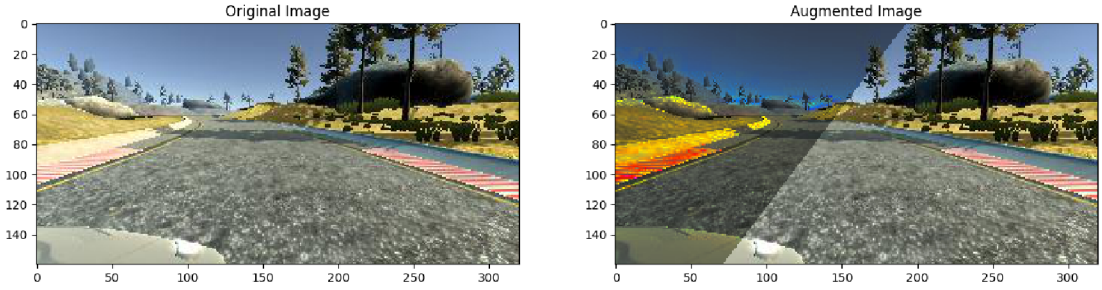

Illustration 2: Image Augmentation

8. Is the car able to navigate correctly on test data?

The car perfectly navigates the first course as displayed in the file: video.mp4