

WRITEUP

Criteria 1: The submission includes a model.py file, drive.py, model.h5 a writeup report and video.mp4.

Answer: All the files are provided in the folder

Criteria 2: The model provided can be used to successfully operate the simulation.

Answer: The model model.h5 can be used to rerun the NN.

Criteria 3: Is the code usable and readable?

Answer: The model.py data is well structured and commented.

Criteria 4: Has appropriate model architecture been employed for the task?

Answer: The model uses the **NVIDIA Neural network** to train the model. The model is sufficiently deep and achieves good results.

The approach was to use a solid neural network to train the model rather than trying to reinvent the wheel. I tried with LeNet Neural Network but was not able to achieve the desired drivability on all the road types. I tried to train it with a very huge training set ~100k but still the car would veer off the road.

Then I migrated to the NVIDIA Neural Network explained in the chapter by the lecturer.

The 5 convolutional layers followed by 4 fully connected layers are sufficiently deep to train the model to run on different roads. The dropout layers help to reduce over fitting.

Input Layer (160*320)	Shape 160*320
Lambda layer (for normalization)	Shape 160*320
Cropping	Shape 65*320
Convolution 2D, (5*5)filter, 2*2 stride, ReLu Activation	24 layers ,shape 31*160
Dropout	24 layers, shape 14*80
Convolution 2D, (5*5) filter, 2*2 stride, ReLu Activation	36 layers, shape 6*40
Convolution 2D (5*5) filter, 2*2 stride, ReLu Activation	48 layers , shape 4*38
Convolution 2D (3*3) filter, 1*1 stride, ReLu Activation	64 layers , shape 2*38
Dropout	64 layers , shape 2*38
Convolution 2D (3*3) filter, 1*1 stride, ReLu Activation	64 layers , shape 1*38
Flatten	2432 Neurons
Fully Connected layer	100 Neurons
Fully Connected layer	50 Neurons
Fully Connected layer	10 Neurons
Fully Connected layer (1 output)	Logits

Criteria 5: Has an attempt been made to reduce overfitting of the model?

The model employs all the tactics such as dropout and convolutions to reduce over fitting. Also data has **been collected from both the tracks.**

Criteria 6: Criteria Have the model parameters been tuned appropriately?

Adam optimizer has been used.

Criteria 7: Is the training data chosen appropriately?

Model Description:

Criteria Is the training data chosen appropriately?

Step 1: I collected data from both the tracks. I particularly paid attention to high steering angles. I recorded only the parts of the track whereby I would move to the side and record coming to center again. This helped the NN to generalize better and learn coming back to middle better.



Step 2: I processed each image to specifically highlight the side path details. The side walk can be recognized by colors such as white, red stripes and unbroken yellow and white line. I preprocessed to highlight these colors and darken other unwanted regions.

Step 3: I used the images from all the three cameras. The correction factors tried were 0.1 and -0.1. But 0.2 gave the best results.

Step 4: Only images with a high steering angle (>0.2) were flipped. Flipping of images with low steering angles had very small effect.

Criteria 8: Is the solution design documented?

The approach taken is discussed in detail here.

Criteria : Is the car able to navigate correctly on test data?

The car can drive around the track well, without touching the boundaries. The car does not leave the track at any time. The video has been saved in the repository as run1.mp4.