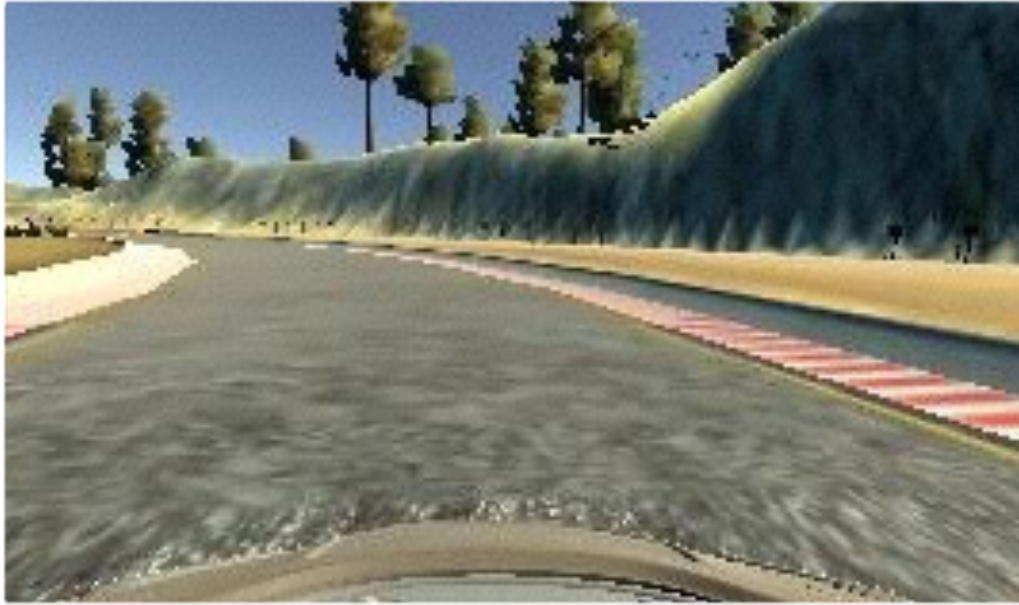


Behavioral-cloning

Build your future self-driving car



Example of
Training dataset



How To Get Data

How to collect representative training data

I have capture the following driving datas: 3 lap driving in the center of the road, 1 lap driving fiercely on the bridge where fail most time, 1 lap driving on the 2nd track, 1 lap driving around the curve, 1 lap driving counter-

1

HOW TO GET DATA

How to collect
representative training
data

2

MODEL ARCHITECT

How to design the
model, choose
parameter and
optimizer

3

PREPROCESS DATA

How to preprocess the
data to capture useful
information

clockwise ,1 lap driving intentionally out of track and recovery back. During those training process, steering was took when it is/about to off the track, don't steering to off the track because it will confuse the data. See example of dataset above.

Model Architect

How model was chosen and optimize

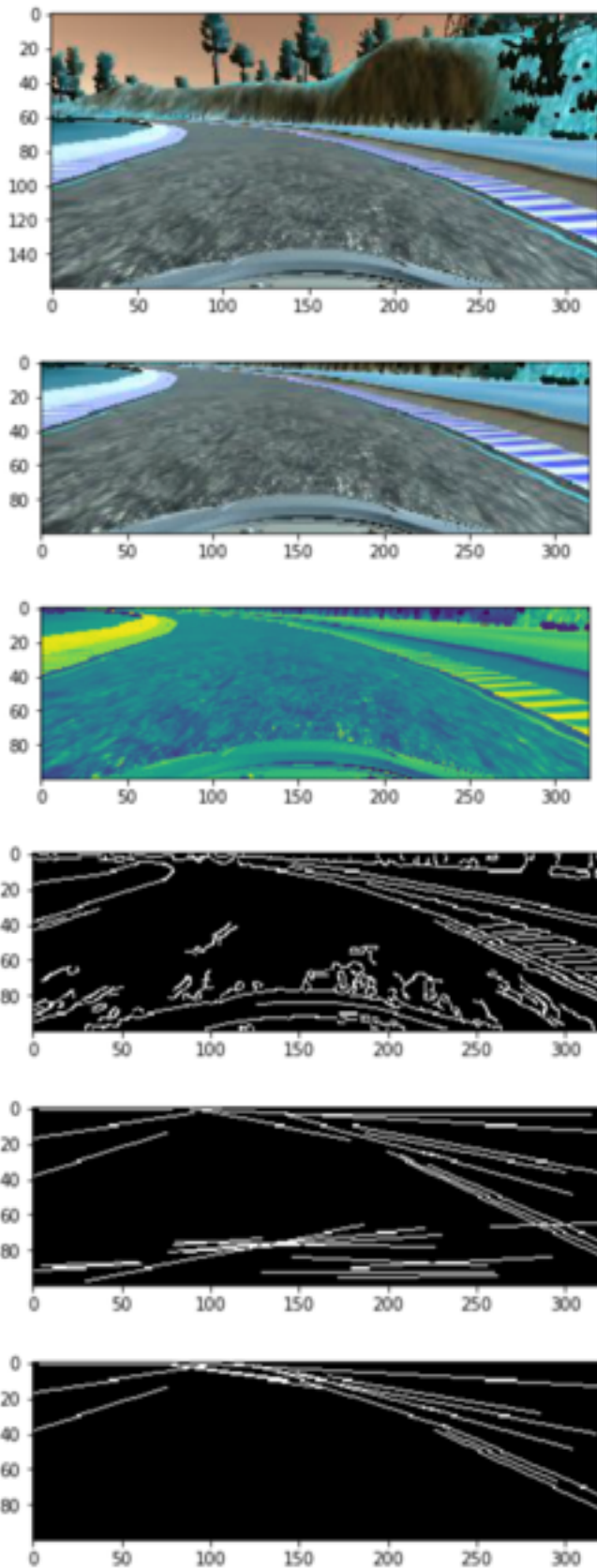
The model was chosen based on Nvidia's self-driving pipeline. After the data has been normalized, it go into 3 5x5 convolution layers following with 2 3x3 convolution layers,1 dropout and 4 fully connect layer. Totally 11 layer was used, the convolution layers has 24,36,48,64,64 numbers of filter each.And it use Adam optimization.The filter size and pipeline is good enough to capture enough feature.



Preprocess Data

How to preprocess the data to capture useful information

As we can see the image captured, the color has less information, and only the low half of the image is important and what is most useful information is the boundary of the road point to which direction which make edge and line is the key feature for this model.



from RGB 3)GaussianBlur 4)Edge detection 5)
Line detection 6)Select interested Area

“result, code, and files”

Utility.py contain most of the common function, the preprocess steps need pipe into driver.py as well to capture feature from raw pic.Video.mp4 contain the video capture in the one lap successful autonomous drive on track using the trained model.

Here is our process pipeline, 1)Crop the image, leave only the bottom half, 2)Convert to Gray