

Collecting Training Data

In order to start collecting training data, you'll need to do the following:

1. Enter Training Mode in the simulator.
2. Start driving the car to get a feel for the controls.
3. When you are ready, hit the record button in the top right to start recording.
4. Continue driving for a few laps or till you feel like you have enough data.
5. Hit the record button in the top right again to stop recording.

If everything went correctly, you should see the following in the directory you selected:

1. **IMG** folder - this folder contains all the frames of your driving.
2. **driving_log.csv** - each row in this sheet correlates your image with the steering angle, throttle, brake, and speed of your car. You'll mainly be using the steering angle.

Center Image	Left Image	Right Image	Steering Angle	Throttle	Break	Speed
IMG/center_2016_10_21_1	IMG/left_2016_10_21_17	IMG/right_2016_10_21_1	0	0.09803098	0	0
IMG/center_2016_10_21_1	IMG/left_2016_10_21_17	IMG/right_2016_10_21_1	0	0.09803098	0	0.08057288

An example of the contents of driving_log.csv.

Training Your Network

Now that you have training data, it's time to build and train your network!

Use Keras to train a network to do the following:

1. Take in an image from the center camera of the car. This is the input to your neural network.
2. Output a new steering angle for the car.

You don't have to worry about the throttle for this project, that will be set for you.

Save your model architecture as **model.json**, and the weights as **model.h5**.



Select Autonomous Mode in the screen above to start validating your network.

Validating Your Network

You can validate your model by launching the simulator and entering autonomous mode.

The car will just sit there until your Python server connects to it and provides it steering angles. Here's how you start your Python server:

1. Install Python Dependencies with Anaconda (**conda install ...**)
 - numpy
 - flask-socketio
 - eventlet
 - pillow
 - h5py
2. Install Python Dependencies with pip (**pip install ...**)
 - keras
3. Download [drive.py](#).
4. Run Server
 - **python drive.py model.json**
 - If you're using Docker for this project: **docker run -it --rm -p 4567:4567 -v `pwd`:/src udacity/carnd-term1-starter-kit python drive.py model.json** or **docker run -it --rm -p 4567:4567 -v \${pwd}:/src udacity/carnd-term1-starter-kit python drive.py model.json**. Port 4567 is used by the simulator to communicate.

Once the model is up and running in **drive.py**, you should see the car move around (and hopefully not off) the track!

the track!

NEXT >