

# Behavioral Cloning

---

## Behavioral Cloning Project

The goals / steps of this project are the following:

- Use the simulator to collect data of good driving behavior
- Build, a convolution neural network in Keras that predicts steering angles from images
- Train and validate the model with a training and validation set
- Test that the model successfully drives around track one without leaving the road
- Summarize the results with a written report

## Rubric Points

---

Here I will consider the [rubric points](#) individually and describe how I addressed each point in my implementation.

---

## Files Submitted & Code Quality

1. Submission includes all required files and can be used to run the simulator in autonomous mode

My project includes the following files:

- **src/model.py** - defines the model and training code , including generators to train and save the model.
- **src/preprocess.py** - shared code for cropping and converting image to YUV
  - Used in drive.py to prepare the data to be sent to the model in the same was as the training data was provided.
- **src/Preprocess\_and\_pickle.py** - takes the set of input files and produces a training and validation files appropriate for training.
- **src/drive.py** for driving the car in autonomous mode

- **src/model\_summary.py** for extracting info about the Keras model in both text and graphical format.
- **output/model.h5** containing a trained convolutional neural network
- **auto\_drive.mp4** - video of the model driving the car on Track 1
- **Behavioral Cloning Writeup.pdf** summarizing the results and approach taken.
- Model training package conforming to Google Cloud ML packaging requirements
  - **cloud-ml/trainer**
    - **model.py** - containing the script to create and compile the model
    - **task.py**
      - Load training/validation data from cloud storage
      - Generator to stream the data to the training backend
      - Save the model when improvements in the loss/accuracy so the best model can be captured automatically.
  - **setup.py** - installs PiPy required packages to run the training model
  - **config.yaml** - configuration on cloud

2. Submission includes functional code Using the Udacity provided simulator and my drive.py file, the car can be driven autonomously around the track by executing

```
python src/drive.py output/bh_clone.hdf5
```

3. Submission code is usable and readable

The model.py file contains the code for training and saving the convolution neural network. The file shows the pipeline I used for training and validating the model.

## Model Architecture and Training Strategy

1. An appropriate model architecture has been employed

My model consists of a convolution neural network based on the NVidia CNN described [here](#). It is a proven model in a similar domain and seemed like a reasonable starting point for the project. With the high level of abstraction provided by the Keras API - putting together the model was relatively straightforward. Most of the heavy lifting was in gathering the training data, training and testing the model.

This is the Keras description of the Model. (Generated with `model_summary.py`)

Layer (type)	Output Shape	Param #
lambda_1 (Lambda)	(None, 80, 320, 3)	0

conv2d_1 (Conv2D)	(None, 38, 158, 24)	1824
-------------------	---------------------	------

dropout_1 (Dropout)	(None, 38, 158, 24)	0
---------------------	---------------------	---

conv2d_2 (Conv2D)	(None, 17, 77, 36)	21636
-------------------	--------------------	-------

conv2d_3 (Conv2D)	(None, 7, 37, 48)	43248
-------------------	-------------------	-------

conv2d_4 (Conv2D)	(None, 5, 35, 64)	27712
-------------------	-------------------	-------

conv2d_5 (Conv2D)	(None, 3, 33, 64)	36928
-------------------	-------------------	-------

flatten_1 (Flatten)	(None, 6336)	0
---------------------	--------------	---

dense_1 (Dense)	(None, 100)	633700
-----------------	-------------	--------

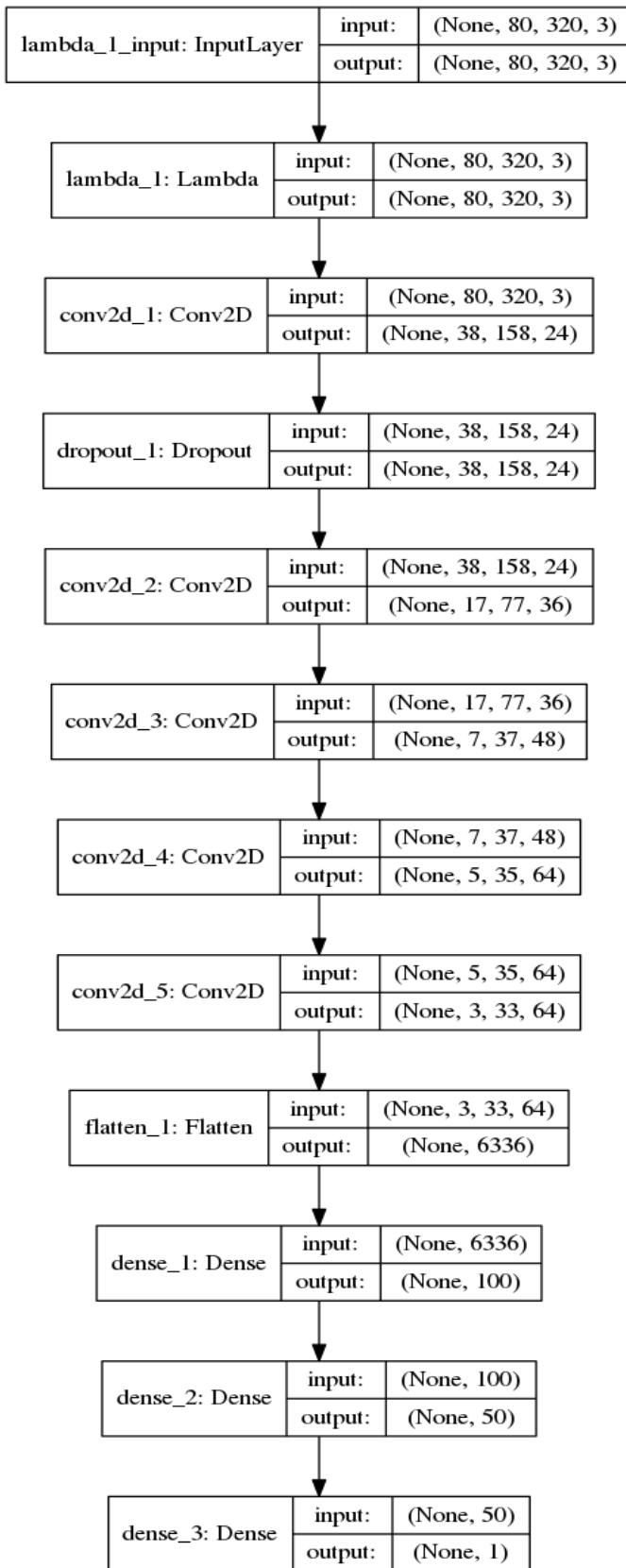
dense_2 (Dense)	(None, 50)	5050
-----------------	------------	------

dense_3 (Dense)	(None, 1)	51
-----------------	-----------	----

Total params: 770,149

Trainable params: 770,149

Keras also provides a mechanism to visualize the model as an image using GraphViz.



The model includes RELU layers to introduce nonlinearity, and the data is normalized in the model using a Keras lambda layer

## 2. Attempts to reduce overfitting in the model

The model contains a dropout layer in order to reduce overfitting.

## 3. Model parameter tuning

The model used an adam optimizer, so the learning rate was not tuned manually.

## 4. Appropriate training data

Training data was chosen to keep the vehicle driving on the road. I used a combination of center lane driving, recovering from the left and right sides of the road ...

For details about how I created the training data, see the next section.

## Model Architecture and Training Strategy

### 1. Solution Design Approach

My first step was to use a convolutional neural network model similar to the NVidia model I thought this model might be appropriate because it was developed to solve a similar problem. In addition CNNs are generally pretty good at dealing with different size input images.

I initially collected a single drive around the track with the keyboard and trained with that. At this point I spent some time reading up on the Google Cloud ML engine to see how I could potentially utilize that for faster training on GPUs. (Later I transitioned to a local machine with a GPU for faster iterations.)

In order to efficiently upload the data, I decided to preprocess the data locally and compress into numpy arrays as pickled data for uploading to the cloud.

This is an example of one of the images from the center camera.



To allow a large number of samples to be loaded without running out of memory I implemented a generator that took a set of numpy pickled files as inputs, shuffled the files and shuffled the samples within each file.

Then I collected more training data from the simulator. I collected data using the mouse to get a smoother driving experience.

The final step was to run the simulator to see how well the car was driving around track one. There were still many places where the vehicle fell off the track.

To improve the driving behavior in these cases, I initially tried to capture recovering from the side of the road to the center, but this was a major pain to do in the simulator. So then I tried adding lap where I swerved from side to side , but it was clear to me that this would create a lot of bad training samples in the input ( images where the steering angle would be very incorrect for the angle of the car to the image).

So I wrote a tool that would allow me to select which images+measurements I wanted to include. So I could capture just the parts of the swerving lap that included the recovery portions. It added column to the .csv file of images, where 0 means exclude and 1 means include.

Using the keyboard it was possible to quickly include or exclude a frame and move on to the next image.

'y' marks the image for inclusion and moves forward

'n' marks the image for exclusion and moves forward.

Using the arrow keys and "spacebar" more fine grained adjustments can be made.

Using this tool I was quickly able to remove bad training data and focus on images that would help the model drive.

These images below show the tool in action.





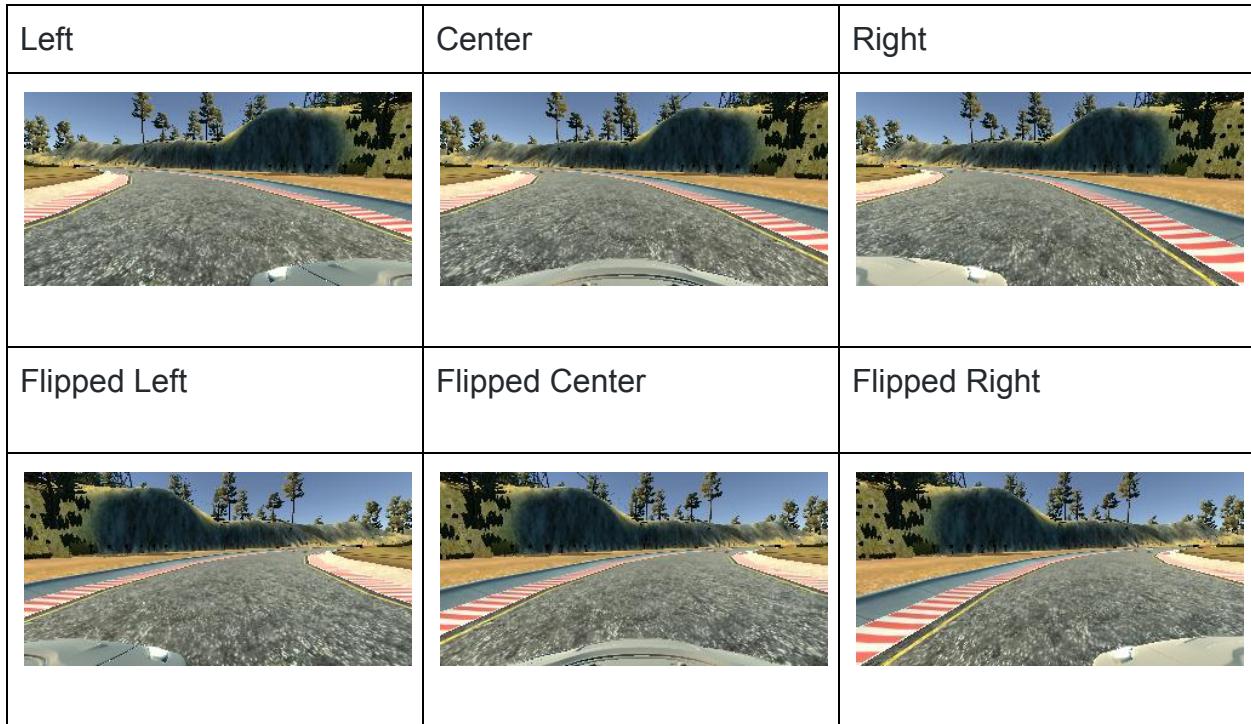
This code is available in `src/image_viewer/app.py`

I then processed these files locally. For each of the files I loaded the .csv files and only included the images that were flagged as selected in the tool. I then processed these files in the following way.

1. Cropped to 320 x 80 pixels.
2. Used the Left and right camera images with a correction angle of +/-1.5
3. Flipped the images on the vertical axis, and flipped the sign of the steering angle.
4. Converted to YUV color space.
5. I split the set of images per file into a training set and validation set with a ratio of 7:3 training to validation.
6. I stored each in `xxx_train.npz` and `xxx_valid.npz`.

Here are some examples of images from the right, left and center cameras.

And these images flipped.



I choose a variety of training laps to capture data from.

- Lap with mouse
- Lap with keyboard
- Lap with just the turns
- Lap with the turns focused on driving in the center as much as possible.
- Lap gently swerving from side to side.
- Lap with harsh steering side to side.
- Partial lap with the bridge.
- Counter clockwise lap with mouse steering.

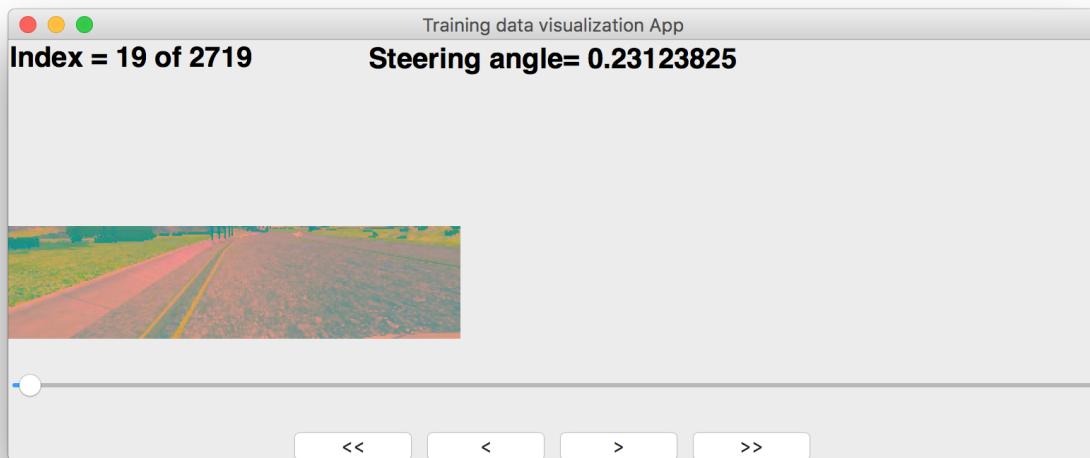
From these sets I tried various combinations until I found a model that drove the car pretty well.

## My final set

Bridge	focused on driving over the bridge
right_turn	all right turns on the track
gentle_swerving	the portions of a full lap that corrected from the edge of the road as far as the center.
focused_center_on_turns	Lap with focus on smoothly driving the middle of the turns.
lap_with_mouse	General lap with mouse
lap2_with_mouse	General lap with mouse
cc_lap_with_mouse	Counter clockwise lap with mouse

In addition I decided it might be useful to have a similar app that could visualize the data in the numpy files to make sure the flipping.

Note that my preprocessing of the image included conversion to YUV - as this is what the NVidia CNN was designed to use. I decided not to convert back to RGB for display, I could get a good enough sense of how the images were being transformed, flipped etc.



I added some scripts that made it a little easier to specify the set of files to pass to the training model.

I had the following sizes in my final training/validation set.

training\_size = 46646 , training\_steps=1457, batch\_size=32

validation\_size=19996, validation\_steps=624 , batch\_size = 32

I used this training data for training the model. The validation set helped determine if the model was over or under fitting. The ideal number of epochs was 2 as evidenced by a relatively fast drop off in the training loss - in Epoch number 3 training loss starts to oscillate and the validation set doesn't change much. So I choose the output from Epoch 2.

#### Output from training job with 5 Epochs

```
root@cf0d86bc3694:/src/src# ./local_train.sh
/src/src /src/src
./data/processed_and_pickled/bridge_train.npz
./data/processed_and_pickled/right_turn_train.npz
./data/processed_and_pickled/gentle_swerving_train.npz
./data/processed_and_pickled/focused_center_on_turns_train.npz
./data/processed_and_pickled/lap_with_mouse_train.npz
./data/processed_and_pickled/lap2_with_mouse_train.npz
./data/processed_and_pickled/cc_lap_with_mouse_train.npz
./data/processed_and_pickled/bridge_valid.npz
./data/processed_and_pickled/right_turn_valid.npz
./data/processed_and_pickled/gentle_swerving_valid.npz
./data/processed_and_pickled/focused_center_on_turns_valid.npz
./data/processed_and_pickled/lap_with_mouse_valid.npz
./data/processed_and_pickled/lap2_with_mouse_valid.npz
./data/processed_and_pickled/cc_lap_with_mouse_valid.npz
Using TensorFlow backend.
2017-11-08 06:47:13.100651: I tensorflow/core/platform/cpu_feature_guard.cc:137] Your CPU
supports instructions that this TensorFlow binary was not compiled to use: SSE4.1 SSE4.2 AVX
AVX2 FMA
2017-11-08 06:47:13.173417: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:892]
successful NUMA node read from SysFS had negative value (-1), but there must be at least one
NUMA node, so returning NUMA node zero
2017-11-08 06:47:13.173629: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1030] Found
device 0 with properties:
name: GeForce GTX 1060 6GB major: 6 minor: 1 memoryClockRate(GHz): 1.835
pciBusID: 0000:01:00.0
totalMemory: 5.93GiB freeMemory: 5.64GiB
2017-11-08 06:47:13.173651: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1120] Creating
TensorFlow device (/device:GPU:0) -> (device: 0, name: GeForce GTX 1060 6GB, pci bus id:
0000:01:00.0, compute capability: 6.1)
training_size = 46646 , training_steps=1457
validation_size=19996, validation_steps=624
..output_20171108_064712/
Serving from ../data/processed_and_pickled/bridge_train.npz
Epoch 1/5
2017-11-08 06:47:18.715893: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1120] Creating
```

```

TensorFlow device (/device:GPU:0) -> (device: 0, name: GeForce GTX 1060 6GB, pci bus id: 0000:01:00.0, compute capability: 6.1)
187/1457 [==>.....] - ETA: 35s - loss: 0.0265 - acc: 6.6845e-04Serving from ../data/proccessed_and_pickled/right_turn_train.npz
225/1457 [====>.....] - ETA: 33s - loss: 0.0247 - acc: 0.0114Serving from ../data/proccessed_and_pickled/gentle_swerving_train.npz
394/1457 [=====>.....] - ETA: 26s - loss: 0.0240 - acc: 0.0125Serving from ../data/proccessed_and_pickled/focused_center_on_turns_train.npz
897/1457 [=====>.....] - ETA: 13s - loss: 0.0126 - acc: 0.0077Serving from ../data/proccessed_and_pickled/lap_with_mouse_train.npz
1094/1457 [=====>.....] - ETA: 8s - loss: 0.0118 - acc: 0.0081Serving from ../data/proccessed_and_pickled/lap2_with_mouse_train.npz
1282/1457 [=====>.....] - ETA: 4s - loss: 0.0110 - acc: 0.0081Serving from ../data/proccessed_and_pickled/cc_lap_with_mouse_train.npz
1451/1457 [=====>.....] - ETA: 0s - loss: 0.0112 - acc: 0.0083Serving from ../data/proccessed_and_pickled/bridge_train.npz
1454/1457 [=====>.....] - ETA: 0s - loss: 0.0112 - acc: 0.0083Serving from ../data/proccessed_and_pickled/bridge_valid.npz
Serving from ../data/proccessed_and_pickled/right_turn_valid.npz
Serving from ../data/proccessed_and_pickled/gentle_swerving_valid.npz
Serving from ../data/proccessed_and_pickled/focused_center_on_turns_valid.npz
Serving from ../data/proccessed_and_pickled/lap_with_mouse_valid.npz
Serving from ../data/proccessed_and_pickled/lap2_with_mouse_valid.npz
Serving from ../data/proccessed_and_pickled/cc_lap_with_mouse_valid.npz
Serving from ../data/proccessed_and_pickled/bridge_valid.npz
Epoch 00001: val_acc improved from -inf to 0.00811, saving model to ./output_20171108_064712/weights-improvement-01-0.01.hdf5
1457/1457 [=====>.....] - 43s 29ms/step - loss: 0.0112 - acc: 0.0083 - val_loss: 0.0181 - val_acc: 0.0081
Epoch 2/5
192/1457 [==>.....] - ETA: 27s - loss: 0.0117 - acc: 6.5104e-04Serving from ../data/proccessed_and_pickled/right_turn_train.npz
232/1457 [==>.....] - ETA: 26s - loss: 0.0112 - acc: 0.0114Serving from ../data/proccessed_and_pickled/gentle_swerving_train.npz
399/1457 [=====>.....] - ETA: 23s - loss: 0.0138 - acc: 0.0123Serving from ../data/proccessed_and_pickled/focused_center_on_turns_train.npz
902/1457 [=====>.....] - ETA: 12s - loss: 0.0078 - acc: 0.0076Serving from ../data/proccessed_and_pickled/lap_with_mouse_train.npz
1099/1457 [=====>.....] - ETA: 8s - loss: 0.0076 - acc: 0.0081Serving from ../data/proccessed_and_pickled/lap2_with_mouse_train.npz
1287/1457 [=====>.....] - ETA: 3s - loss: 0.0072 - acc: 0.0081Serving from ../data/proccessed_and_pickled/cc_lap_with_mouse_train.npz
1456/1457 [=====>.....] - ETA: 0s - loss: 0.0077 - acc: 0.0083Serving from ../data/proccessed_and_pickled/right_turn_valid.npz
Serving from ../data/proccessed_and_pickled/gentle_swerving_valid.npz
Serving from ../data/proccessed_and_pickled/focused_center_on_turns_valid.npz
Serving from ../data/proccessed_and_pickled/lap_with_mouse_valid.npz
Serving from ../data/proccessed_and_pickled/lap2_with_mouse_valid.npz
Serving from ../data/proccessed_and_pickled/cc_lap_with_mouse_valid.npz
Serving from ../data/proccessed_and_pickled/bridge_valid.npz
Epoch 00002: val_acc did not improve
1457/1457 [=====>.....] - 42s 28ms/step - loss: 0.0077 - acc: 0.0083 - val_loss: 0.0189 - val_acc: 0.0081
Epoch 3/5
Serving from ../data/proccessed_and_pickled/bridge_train.npz
197/1457 [==>.....] - ETA: 29s - loss: 0.0079 - acc: 7.9315e-04Serving from ../data/proccessed_and_pickled/right_turn_train.npz
237/1457 [==>.....] - ETA: 27s - loss: 0.0080 - acc: 0.0113Serving from ../data/proccessed_and_pickled/gentle_swerving_train.npz
404/1457 [=====>.....] - ETA: 24s - loss: 0.0108 - acc: 0.0122Serving from ../data/proccessed_and_pickled/focused_center_on_turns_train.npz
907/1457 [=====>.....] - ETA: 13s - loss: 0.0063 - acc: 0.0076Serving from ../data/proccessed_and_pickled/lap_with_mouse_train.npz
1104/1457 [=====>.....] - ETA: 8s - loss: 0.0062 - acc: 0.0081Serving from ../data/proccessed_and_pickled/lap2_with_mouse_train.npz

```

```
1292/1457 [=====>....] - ETA: 3s - loss: 0.0060 - acc: 0.0081Serving from  
..../data/proccessed_and_pickled/cc_lap_with_mouse_train.npz  
1455/1457 [=====>.] - ETA: 0s - loss: 0.0064 - acc: 0.0083Serving from  
..../data/proccessed_and_pickled/right_turn_valid.npz  
Serving from ..../data/proccessed_and_pickled/gentle_swerving_valid.npz  
Serving from ..../data/proccessed_and_pickled/focused_center_on_turns_valid.npz  
Serving from ..../data/proccessed_and_pickled/lap_with_mouse_valid.npz  
Serving from ..../data/proccessed_and_pickled/lap2_with_mouse_valid.npz  
Serving from ..../data/proccessed_and_pickled/cc_lap_with_mouse_valid.npz  
Serving from ..../data/proccessed_and_pickled/bridge_valid.npz  
Epoch 00003: val_acc did not improve  
1457/1457 [=====] - 42s 29ms/step - loss: 0.0064 - acc: 0.0083 -  
val_loss: 0.0161 - val_acc: 0.0081  
Epoch 4/5  
 4/1457 [.....] - ETA: 29s - loss: 0.0057 - acc: 0.0000e+00Serving  
from ..../data/proccessed_and_pickled/bridge_train.npz  
 202/1457 [==>.....] - ETA: 29s - loss: 0.0078 - acc: 7.7351e-04Serving  
from ..../data/proccessed_and_pickled/right_turn_train.npz  
 242/1457 [==>.....] - ETA: 27s - loss: 0.0078 - acc: 0.0111Serving from  
..../data/proccessed_and_pickled/gentle_swerving_train.npz  
 409/1457 [=====>.....] - ETA: 24s - loss: 0.0100 - acc: 0.0121Serving from  
..../data/proccessed_and_pickled/focused_center_on_turns_train.npz  
 912/1457 [=====>.....] - ETA: 12s - loss: 0.0059 - acc: 0.0076Serving from  
..../data/proccessed_and_pickled/lap_with_mouse_train.npz  
1109/1457 [=====>.....] - ETA: 8s - loss: 0.0058 - acc: 0.0080Serving from  
..../data/proccessed_and_pickled/lap2_with_mouse_train.npz  
1297/1457 [=====>....] - ETA: 3s - loss: 0.0056 - acc: 0.0081Serving from  
..../data/proccessed_and_pickled/cc_lap_with_mouse_train.npz  
1454/1457 [=====>.] - ETA: 0s - loss: 0.0060 - acc: 0.0082Serving from  
..../data/proccessed_and_pickled/right_turn_valid.npz  
Serving from ..../data/proccessed_and_pickled/gentle_swerving_valid.npz  
Serving from ..../data/proccessed_and_pickled/focused_center_on_turns_valid.npz  
Serving from ..../data/proccessed_and_pickled/lap_with_mouse_valid.npz  
Serving from ..../data/proccessed_and_pickled/lap2_with_mouse_valid.npz  
Serving from ..../data/proccessed_and_pickled/cc_lap_with_mouse_valid.npz  
Serving from ..../data/proccessed_and_pickled/bridge_valid.npz  
Epoch 00004: val_acc did not improve  
1457/1457 [=====] - 41s 28ms/step - loss: 0.0060 - acc: 0.0082 -  
val_loss: 0.0176 - val_acc: 0.0081  
Epoch 5/5  
 10/1457 [.....] - ETA: 29s - loss: 0.0057 - acc: 0.0063Serving from  
..../data/proccessed_and_pickled/bridge_train.npz  
 207/1457 [==>.....] - ETA: 29s - loss: 0.0067 - acc: 0.0011Serving from  
..../data/proccessed_and_pickled/right_turn_train.npz  
 247/1457 [==>.....] - ETA: 27s - loss: 0.0067 - acc: 0.0111Serving from  
..../data/proccessed_and_pickled/gentle_swerving_train.npz  
 414/1457 [=====>.....] - ETA: 24s - loss: 0.0088 - acc: 0.0121Serving from  
..../data/proccessed_and_pickled/focused_center_on_turns_train.npz  
 917/1457 [=====>.....] - ETA: 12s - loss: 0.0053 - acc: 0.0076Serving from  
..../data/proccessed_and_pickled/lap_with_mouse_train.npz  
1114/1457 [=====>.....] - ETA: 8s - loss: 0.0052 - acc: 0.0081Serving from  
..../data/proccessed_and_pickled/lap2_with_mouse_train.npz  
1302/1457 [=====>....] - ETA: 3s - loss: 0.0049 - acc: 0.0081Serving from  
..../data/proccessed_and_pickled/cc_lap_with_mouse_train.npz  
1456/1457 [=====>.] - ETA: 0s - loss: 0.0053 - acc: 0.0082Serving from  
..../data/proccessed_and_pickled/right_turn_valid.npz  
Serving from ..../data/proccessed_and_pickled/gentle_swerving_valid.npz  
Serving from ..../data/proccessed_and_pickled/focused_center_on_turns_valid.npz  
Serving from ..../data/proccessed_and_pickled/lap_with_mouse_valid.npz  
Serving from ..../data/proccessed_and_pickled/lap2_with_mouse_valid.npz  
Serving from ..../data/proccessed_and_pickled/cc_lap_with_mouse_valid.npz  
Serving from ..../data/proccessed_and_pickled/bridge_valid.npz  
Epoch 00005: val_acc did not improve  
1457/1457 [=====] - 42s 29ms/step - loss: 0.0053 - acc: 0.0082 -  
val_loss: 0.0148 - val_acc: 0.0081
```

