# Behavioral Cloning for Udacity Drive Simulator

## This Project was prepared for Udacity Self Driving Car ND - Term one. 12 Feb 2019

Notes:

Project Uses Nvidia model published in NVIDIA End To End Driving Paper - Thanks and Acknowledgements

There are two environments. One for Drive simulator Model building Environment (this file) and second environment is for simulation interface bridge for realtime simulation (drive.py)

I have used two different workstations, a mac and a tensorflow gpu on windows 10 pro Xeon W workstation on GPU 1080, unfortunately this environment broke down with pip issues when installing pip (12.0 - relased Feb 2019 )along with eventlet and socket io which we use for drive.py. I switched between mac dual core mojave tensorflow cpu (conda based). I finished off in mac. Final movie is edited in apple imove. Music track is americal naitive pipe.

There are two ipnyb jupyter notebooks. One shows all the experiments from the start and this one is the second notebook where i finalised a model for final testing and submission

Date:12 Feb 2019

Change request for release : 1

Release no:

Prepared for : Udacity ND Term 1 BC

Prepared by : ARV

Design doc ref:

Release doc ref:

Tech doc ref:

Testing doc ref:

User Read me ref :

Admin and Dev read me ref:

In [45]:

```
!conda list
```

```
# packages in environment at /miniconda3/envs/udrivesimul:
#
# Name                    Version                   Build  Channel
absl-py                   0.7.0                    py36_1000    conda-f
orge
appnope                   0.1.0            py36hf537a9a_0
asn1crypto                0.24.0                py36_1003    conda-f
orge
astor                     0.7.1                     py_0    conda-f
orge
backcall                  0.1.0                    py36_0
blas                      1.0                         mkl    anacond
a
bleach                    3.1.0                    pypi_0    pypi
bzip2                     1.0.6             h1de35cc_1002    conda-f
orge
c-ares                    1.15.0            h1de35cc_1001    conda-f
orge
ca-certificates           2019.1.23                    0
cairo                     1.14.12           h9d4d9ac_1005    conda-f
orge
certifi                   2018.11.29               py36_0
cffi                      1.11.5          py36h342bebf_1001    conda
-forge
click                     7.0                       py_0    conda-f
orge
cloudpickle               0.7.0                    pypi_0    pypi
cryptography              2.5              py36ha12b0ac_0
cycler                    0.10.0                    py_1    conda-f
orge
dask                      1.1.1                    pypi_0    pypi
dbus                      1.13.6              h90a0687_0
decorator                 4.3.2                    py36_0
entrypoints               0.3                      py36_0
eventlet                  0.23.0                py36_1000    conda-f
orge
expat                     2.2.6               h0a44026_0
ffmpeg                    4.1                heb45b42_1000    conda-f
orge
flask                     1.0.2                     py_2    conda-f
orge
fontconfig                2.13.1            h1e4e890_1000    conda-f
orge
freetype                  2.9.1               hb4e5f40_0    anacond
a
gast                      0.2.1.post0               py_0    conda-f
orge
gettext                   0.19.8.1          hcca000d_1001    conda-f
orge
giflib                    5.1.4             h1de35cc_1001    conda-f
orge
glib                      2.56.2            h67dad55_1001    conda-f
orge
gmp                       6.1.2             h0a44026_1000    conda-f
orge
gnutls                    3.6.5             h53004b3_1001    conda-f
orge
graphite2                 1.3.13            h2098e52_1000    conda-f
orge
greenlet                  0.4.13                   py36_0    conda-f
orge
```

```
grpcio                    1.16.1           py36h044775b_1
h5py                      2.8.0            py36h878fce3_3    anacond
a
harfbuzz                  1.9.0             h9889186_1001    conda-f
orge
hdf5                      1.10.2               hfa1e0ec_1    anacond
a
icu                       58.2              h0a44026_1000    conda-f
orge
idna                      2.8                   py36_1000    conda-f
orge
imageio                   2.5.0                    pypi_0    pypi
imgaug                    0.2.8                    pypi_0    pypi
intel-openmp              2019.1                      144
ipykernel                 5.1.0            py36h39e3cac_0
ipython                   7.2.0            py36h39e3cac_0
ipython_genutils          0.2.0            py36h241746c_0
ipywidgets                7.4.2                    py36_0
itsdangerous              1.1.0                      py_0    conda-f
orge
jasper                    1.900.1           h636a363_1006    conda-f
orge
jedi                      0.13.2                   py36_0
jinja2                    2.10                       py_1    conda-f
orge
jpeg                      9c                h1de35cc_1001    conda-f
orge
jsonschema                2.6.0            py36hb385e00_0
jupyter                   1.0.0                    py36_7
jupyter_client            5.2.4                    py36_0
jupyter_console           6.0.0                    py36_0
jupyter_core              4.4.0                    py36_0
keras                     2.2.4                    py36_0    conda-f
orge
keras-applications        1.0.4                      py_1    conda-f
orge
keras-preprocessing       1.0.2                      py_1    conda-f
orge
kiwisolver                1.0.1           py36h04f5b5a_1002    conda
-forge
libcxx                    7.0.0                h2d50403_2    conda-f
orge
libffi                    3.2.1             h0a44026_1005    conda-f
orge
libgfortran               3.0.1                h93005f0_2
libgpuarray               0.7.6             h1de35cc_1003    conda-f
orge
libiconv                  1.15              h1de35cc_1004    conda-f
orge
libopenblas               0.3.3                hdc02c5d_3
libpng                    1.6.36               ha441bb4_0    anacond
a
libprotobuf               3.6.1             hd9629dc_1000    conda-f
orge
libsodium                 1.0.16               h3efe00b_0
libtiff                   4.0.10               hcb84e12_2    anacond
a
libwebp                   1.0.2                h801f6e5_1    conda-f
orge
libxml2                   2.9.8             hf14e9c8_1005    conda-f
orge
```

```
llvm-meta                 7.0.0                         0    conda-f
orge
mako                      1.0.7                       py_1    conda-f
orge
markdown                  2.6.11                      py_0    conda-f
orge
markupsafe                1.1.0           py36h1de35cc_1000    conda
-forge
matplotlib                3.0.2                 py36_1002    conda-f
orge
matplotlib-base           3.0.2           py36hf043ca5_1002    conda
-forge
mistune                   0.8.4            py36h1de35cc_0
mkl                       2019.1                      144
mkl_fft                   1.0.10                  py36_0    conda-f
orge
mkl_random                1.0.2                   py36_0    conda-f
orge
nbconvert                 5.3.1                   py36_0
nbformat                  4.4.0            py36h827af21_0
ncurses                   6.1               h0a44026_1002    conda-f
orge
nettle                    3.4.1             h1de35cc_1002    conda-f
orge
networkx                  2.2                     pypi_0    pypi
notebook                  5.7.4                   py36_0
numpy                     1.15.4           py36hacdab7b_0    anacond
a
numpy-base                1.15.4           py36h6575580_0    anacond
a
olefile                   0.46                    py36_0    anacond
a
openblas                  0.3.3             hdc02c5d_1001    conda-f
orge
opencv                    3.4.1            py36h6fd60c2_1    anacond
a
opencv-python             4.0.0.21                pypi_0    pypi
openh264                  1.8.0              hd9629dc_1000    conda-f
orge
openssl                   1.1.1a                h1de35cc_0
pandas                    0.24.1           py36h0a44026_0    anacond
a
pandoc                    2.2.3.2                       0
pandocfilters             1.4.2                   py36_1
parso                     0.3.2                   py36_0
pcre                      8.41              h0a44026_1003    conda-f
orge
pexpect                   4.6.0                   py36_0
pickleshare               0.7.5                   py36_0
pillow                    5.4.1            py36hb68e598_0    anacond
a
pip                       19.0.1                  py36_0
pixman                    0.34.0            h1de35cc_1003    conda-f
orge
prometheus_client         0.5.0                   py36_0
prompt_toolkit            2.0.8                     py_0
protobuf                  3.6.1           py36h0a44026_1001    conda
-forge
ptyprocess                0.6.0                   py36_0
pycparser                 2.19                      py_0    conda-f
orge
```

```
pygments                  2.3.1                    py36_0
pygpu                     0.7.6           py36h917ab60_1000    conda
-forge
pyopenssl                 19.0.0                   py36_0    conda-f
orge
pyparsing                 2.3.1                      py_0    conda-f
orge
pyqt                      5.9.2            py36h655552a_2
python                    3.6.8                haf84260_0
python-dateutil           2.7.5                    py36_0
python-engineio           3.0.0                      py_0    conda-f
orge
python-socketio           3.1.2                      py_0    conda-f
orge
pytz                      2018.9                   py36_0    anacond
a
pywavelets                1.0.1                    pypi_0    pypi
pyyaml                    3.13            py36h1de35cc_1001    conda
-forge
pyzmq                     17.1.2           py36h0a44026_2
qt                        5.9.7                h468cd18_1
qtconsole                 4.4.3                    py36_0
readline                  7.0               hcfe32e1_1001    conda-f
orge
scikit-image              0.14.2                   pypi_0    pypi
scikit-learn              0.20.2           py36h27c97d8_0    anacond
a
scipy                     1.2.0            py36h1410ff5_0    anacond
a
send2trash                1.5.0                    py36_0
setuptools                40.8.0                   py36_0    conda-f
orge
shapely                   1.6.4.post2              pypi_0    pypi
sip                       4.19.8           py36h0a44026_0
six                       1.12.0                py36_1000    conda-f
orge
sqlite                    3.26.0            h1765d9f_1000    conda-f
orge
tensorboard               1.10.0                   py36_0    conda-f
orge
tensorflow                1.10.0                   py36_0    conda-f
orge
termcolor                 1.1.0                      py_2    conda-f
orge
terminado                 0.8.1                    py36_1
testpath                  0.4.2                    py36_0
theano                    1.0.3                    py36_0    conda-f
orge
tk                        8.6.9             ha441bb4_1000    conda-f
orge
toolz                     0.9.0                    pypi_0    pypi
tornado                   5.1.1            py36h1de35cc_0
traitlets                 4.3.2            py36h65bd3ce_0
wcwidth                   0.1.7            py36h8c6ec74_0
webencodings              0.5.1                    py36_1
werkzeug                  0.14.1                     py_0    conda-f
orge
wheel                     0.32.3                   py36_0    conda-f
orge
widgetsnbextension        3.4.2                    py36_0
x264                      1!152.20180717    h1de35cc_1001    conda-f
```

```
orge
xz                        5.2.4            h1de35cc_1001    conda-f
orge
yaml                      0.1.7            h1de35cc_1001    conda-f
orge
zeromq                    4.3.1               h0a44026_3
zlib                      1.2.11           h1de35cc_1004    conda-f
orge
zstd                      1.3.7               h5bba6e5_0    anacond
a
```

Let us do the imports. All the imports done were collected and collated here

In [17]:

```
import os
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
import keras
from keras.models import Sequential
from keras.optimizers import Adam
from keras.layers import Convolution2D, MaxPooling2D, Dropout, Flatten, Dense
from sklearn.utils import shuffle
from sklearn.model_selection import train_test_split
from imgaug import augmenters as iaa
import cv2
import pandas as pd
import ntpath
import random
```

Handle the CSV file containing ref to snapshot of manual driving files simulating real cameras on a car

In [18]:

```
datadir = 'new_track'
columns =['center','left','right','steering','throttle','reverse','speed']
data=pd.read_csv('driving_log.csv',names=columns)
data.head
```

Out[18]:

```
<bound method NDFrame.head of
           center  \
0     C:\Users\Vijy\Desktop\new_track\IMG\center_201...
1     C:\Users\Vijy\Desktop\new_track\IMG\center_201...
2     C:\Users\Vijy\Desktop\new_track\IMG\center_201...
3     C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4     C:\Users\Vijy\Desktop\new_track\IMG\center_201...
5     C:\Users\Vijy\Desktop\new_track\IMG\center_201...
6     C:\Users\Vijy\Desktop\new_track\IMG\center_201...
7     C:\Users\Vijy\Desktop\new_track\IMG\center_201...
8     C:\Users\Vijy\Desktop\new_track\IMG\center_201...
9     C:\Users\Vijy\Desktop\new_track\IMG\center_201...
10    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
11    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
12    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
13    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
14    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
15    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
16    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
17    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
18    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
19    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
20    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
21    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
22    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
23    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
24    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
25    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
26    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
27    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
28    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
29    C:\Users\Vijy\Desktop\new_track\IMG\center_201...
...                                                 ...
4023  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4024  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4025  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4026  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4027  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4028  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4029  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4030  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4031  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4032  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4033  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4034  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4035  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4036  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4037  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4038  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4039  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4040  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4041  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4042  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4043  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4044  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4045  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4046  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4047  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
4048  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
```

```
      4049  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
      4050  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
      4051  C:\Users\Vijy\Desktop\new_track\IMG\center_201...
      4052  C:\Users\Vijy\Desktop\new_track\IMG\center_201...

                                                      left  \
      0     C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      1     C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      2     C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      3     C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4     C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      5     C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      6     C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      7     C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      8     C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      9     C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      10    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      11    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      12    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      13    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      14    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      15    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      16    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      17    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      18    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      19    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      20    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      21    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      22    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      23    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      24    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      25    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      26    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      27    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      28    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      29    C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      ...                                                 ...
      4023  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4024  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4025  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4026  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4027  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4028  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4029  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4030  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4031  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4032  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4033  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4034  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4035  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4036  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4037  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4038  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4039  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4040  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4041  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4042  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4043  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4044  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4045  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
      4046  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
```

```
4047  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
4048  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
4049  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
4050  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
4051  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...
4052  C:\Users\Vijy\Desktop\new_track\IMG\left_2018_...

                                                  right  steering  t
hrottle  \
0     C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
1     C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
2     C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
3     C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4     C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
5     C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
6     C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
7     C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
8     C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
9     C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.024006
10    C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.202773
11    C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.382694
12    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.050000
0.642727
13    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.250000
0.863326
14    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.400000
1.000000
15    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.535431
1.000000
16    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.431962
1.000000
17    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.190957
1.000000
18    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.006140
1.000000
19    C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
1.000000
20    C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
1.000000
21    C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
1.000000
22    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.050000
1.000000
23    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.250000
1.000000
24    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.417654
1.000000
25    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.148133
1.000000
```

```
26    C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
1.000000
27    C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
1.000000
28    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.150000
1.000000
29    C:\Users\Vijy\Desktop\new_track\IMG\right_2018... -0.203962
1.000000
...                                                 ...       ...
    ...
4023  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.150000
1.000000
4024  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.300000
1.000000
4025  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.097232
0.811835
4026  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.566915
4027  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.380225
4028  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.132538
4029  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4030  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4031  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4032  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4033  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4034  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4035  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4036  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4037  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4038  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4039  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4040  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4041  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4042  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4043  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4044  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4045  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4046  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4047  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4048  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
```

```
0.000000
4049  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4050  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4051  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
4052  C:\Users\Vijy\Desktop\new_track\IMG\right_2018...  0.000000
0.000000
```

| | reverse | speed |
|---|---|---|
| 0 | 0.0 | 0.649786 |
| 1 | 0.0 | 0.627942 |
| 2 | 0.0 | 0.622910 |
| 3 | 0.0 | 0.619162 |
| 4 | 0.0 | 0.615438 |
| 5 | 0.0 | 0.610506 |
| 6 | 0.0 | 0.606834 |
| 7 | 0.0 | 0.601971 |
| 8 | 0.0 | 0.598350 |
| 9 | 0.0 | 0.620654 |
| 10 | 0.0 | 0.707000 |
| 11 | 0.0 | 0.946799 |
| 12 | 0.0 | 1.434013 |
| 13 | 0.0 | 2.173052 |
| 14 | 0.0 | 2.864847 |
| 15 | 0.0 | 3.791584 |
| 16 | 0.0 | 4.489107 |
| 17 | 0.0 | 5.422441 |
| 18 | 0.0 | 6.111838 |
| 19 | 0.0 | 7.026899 |
| 20 | 0.0 | 7.709986 |
| 21 | 0.0 | 8.614678 |
| 22 | 0.0 | 9.288085 |
| 23 | 0.0 | 10.162890 |
| 24 | 0.0 | 10.792000 |
| 25 | 0.0 | 11.502830 |
| 26 | 0.0 | 12.418390 |
| 27 | 0.0 | 13.250700 |
| 28 | 0.0 | 14.023000 |
| 29 | 0.0 | 14.644770 |
| ... | ... | ... |
| 4023 | 0.0 | 30.184000 |
| 4024 | 0.0 | 30.169180 |
| 4025 | 0.0 | 30.153680 |
| 4026 | 0.0 | 30.089560 |
| 4027 | 0.0 | 30.040620 |
| 4028 | 0.0 | 29.987340 |
| 4029 | 0.0 | 29.788930 |
| 4030 | 0.0 | 29.608500 |
| 4031 | 0.0 | 29.369460 |
| 4032 | 0.0 | 29.132300 |
| 4033 | 0.0 | 28.955670 |
| 4034 | 0.0 | 28.721800 |
| 4035 | 0.0 | 28.490560 |
| 4036 | 0.0 | 28.318310 |
| 4037 | 0.0 | 28.090210 |
| 4038 | 0.0 | 27.920720 |
| 4039 | 0.0 | 27.752260 |
| 4040 | 0.0 | 27.529250 |
| 4041 | 0.0 | 27.363170 |

```
4042      0.0  27.143300
4043      0.0  26.979560
4044      0.0  26.762780
4045      0.0  26.601340
4046      0.0  26.387600
4047      0.0  26.228420
4048      0.0  26.017680
4049      0.0  25.808640
4050      0.0  25.652960
4051      0.0  25.446850
4052      0.0  17.007810

[4053 rows x 7 columns]>
```

Handle path data and verifing first few lines with 'head'

In [19]:

```
def path_leaf(path):
    head,tail = ntpath.split(path)
    return tail
data['center'] = data['center'].apply(path_leaf)
data['left'] = data['left'].apply(path_leaf)
data['right'] = data['right'].apply(path_leaf)
data.head()
```

Out[19]:

| | center | left | |
|---|---|---|---|
| **0** | center_2018_07_16_17_11_43_382.jpg | left_2018_07_16_17_11_43_382.jpg | right_2018 |
| **1** | center_2018_07_16_17_11_43_670.jpg | left_2018_07_16_17_11_43_670.jpg | right_2018 |
| **2** | center_2018_07_16_17_11_43_724.jpg | left_2018_07_16_17_11_43_724.jpg | right_2018 |
| **3** | center_2018_07_16_17_11_43_792.jpg | left_2018_07_16_17_11_43_792.jpg | right_2018 |
| **4** | center_2018_07_16_17_11_43_860.jpg | left_2018_07_16_17_11_43_860.jpg | right_2018 |

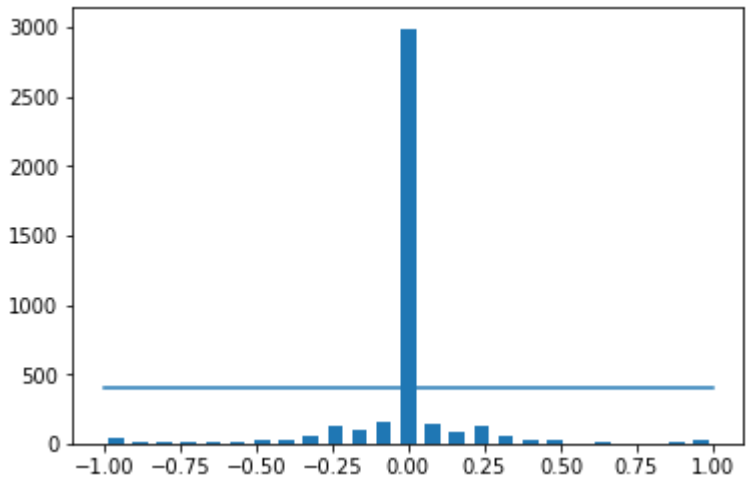
Plotting steering angle as bins between values of -1 and +1, against number of occurances in the training

In [20]:

```
num_bins = 25
samples_per_bin = 400
hist, bins = np.histogram(data['steering'], num_bins)
center = (bins[:-1]+ bins[1:]) * 0.5
plt.bar(center, hist, width=0.05)
plt.plot((np.min(data['steering']), np.max(data['steering'])), (samples_per_bin,
 samples_per_bin))
```

Out[20]:

```
[<matplotlib.lines.Line2D at 0x1c2f1f7390>]
```




Data study for feature enineering and data wrangling. Removing too much bias for 0 deg straight angle and attemting at generalising beyond this predominatly straight track road

In [21]:

```
print('total data:', len(data))
remove_list = []
for j in range(num_bins):
  list_ = []
  for i in range(len(data['steering'])):
    if data['steering'][i] >= bins[j] and data['steering'][i] <= bins[j+1]:
      list_.append(i)
  list_ = shuffle(list_)
  list_ = list_[samples_per_bin:]
  remove_list.extend(list_)

print('removed:', len(remove_list))
data.drop(data.index[remove_list], inplace=True)
print('remaining:', len(data))
```

```
total data: 4053
removed: 2590
remaining: 1463
```

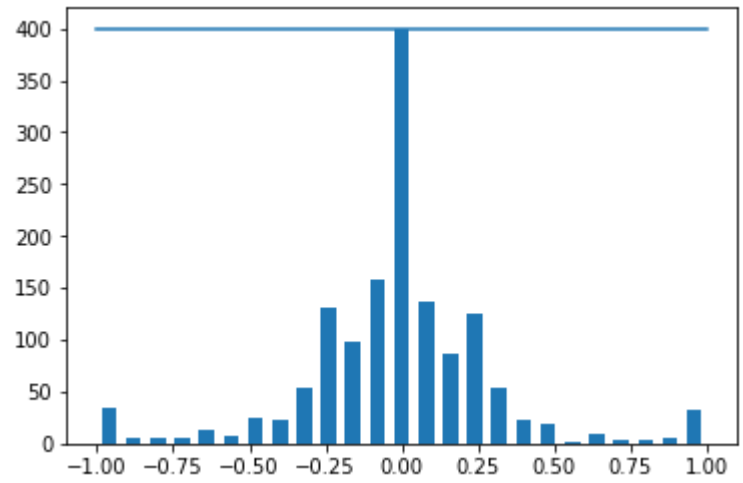
Histogramming steering after removal

In [22]:

```
hist, _ = np.histogram(data['steering'], (num_bins))
plt.bar(center, hist, width=0.05)
plt.plot((np.min(data['steering']), np.max(data['steering'])), (samples_per_bin,
 samples_per_bin))
```

Out[22]:

```
[<matplotlib.lines.Line2D at 0x1c2e5140b8>]
```




Printing a sample and verifying data pattern

In [23]:

```
print(data.iloc[1])
```

```
center      center_2018_07_16_17_11_44_485.jpg
left          left_2018_07_16_17_11_44_485.jpg
right        right_2018_07_16_17_11_44_485.jpg
steering                                 -0.25
throttle                               0.863326
reverse                                       0
speed                                   2.17305
Name: 13, dtype: object
```

Function for picking up images and returning image paths and steering angle value from csv file

In [24]:

```
def load_img_steering(datadir, df):
  image_path = []
  steering = []
  for i in range(len(data)):
    indexed_data = data.iloc[i]
    center, left, right = indexed_data[0], indexed_data[1], indexed_data[2]
    image_path.append(os.path.join(datadir, center.strip()))
    steering.append(float(indexed_data[3]))
    # left image append
    image_path.append(os.path.join(datadir,left.strip()))
    steering.append(float(indexed_data[3])+0.15)
    # right image append
    image_path.append(os.path.join(datadir,right.strip()))
    steering.append(float(indexed_data[3])-0.15)
  image_paths = np.asarray(image_path)
  steerings = np.asarray(steering)
  return image_paths, steerings
```

Adding real file system point for dataset. Spliting training and validation set from collected data

In [25]:

```
image_paths, steerings = load_img_steering(datadir + '/IMG', data)
X_train, X_valid, y_train, y_valid = train_test_split(image_paths, steerings, te
st_size=0.2, random_state=6)
```

Printing out our training and validation split set, plotting histogram to visualise steering angles

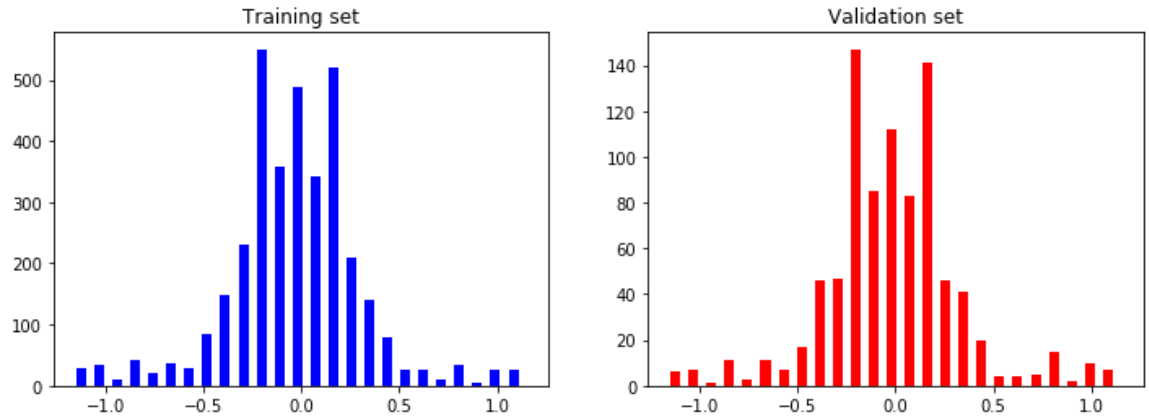
In [26]:

```
print('Training Samples: {}\nValid Samples: {}'.format(len(X_train), len(X_valid
)))
fig, axes = plt.subplots(1, 2, figsize=(12, 4))
axes[0].hist(y_train, bins=num_bins, width=0.05, color='blue')
axes[0].set_title('Training set')
axes[1].hist(y_valid, bins=num_bins, width=0.05, color='red')
axes[1].set_title('Validation set')
```

```
Training Samples: 3511
Valid Samples: 878
```

Out[26]:

```
Text(0.5, 1.0, 'Validation set')
```




***Augmentation Techniques***

We attempt various augmentation methods to try and generalise trained model from the collected training data.

Zoom affine augmentation

In [27]:

```
def zoom(image):
  zoom = iaa.Affine(scale=(1, 1.3))
  image = zoom.augment_image(image)
  return image
```

In [28]:

```
print(image)
```

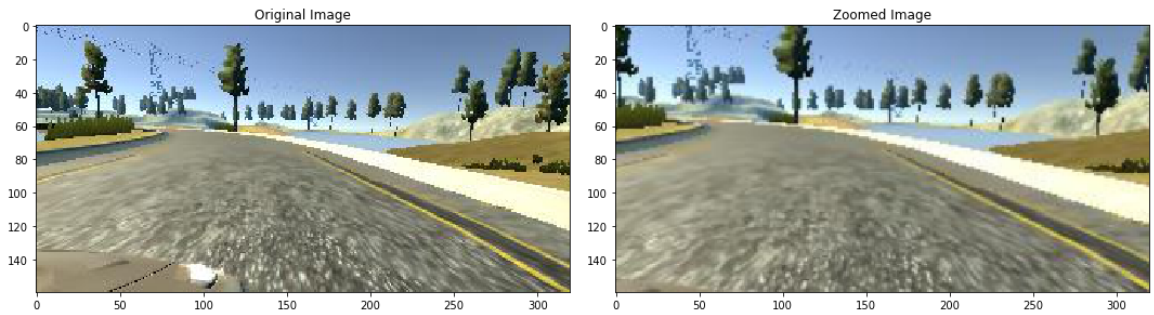
```
IMG/IMG/right_2018_07_16_17_11_47_988.jpg
```

In [29]:

```
image = image_paths[random.randint(0, 1000)]
original_image = mpimg.imread(image)
zoomed_image = zoom(original_image)

fig, axs = plt.subplots(1, 2, figsize=(15, 10))
fig.tight_layout()

axs[0].imshow(original_image)
axs[0].set_title('Original Image')

axs[1].imshow(zoomed_image)
axs[1].set_title('Zoomed Image')
```

Out[29]:

```
Text(0.5, 1.0, 'Zoomed Image')
```



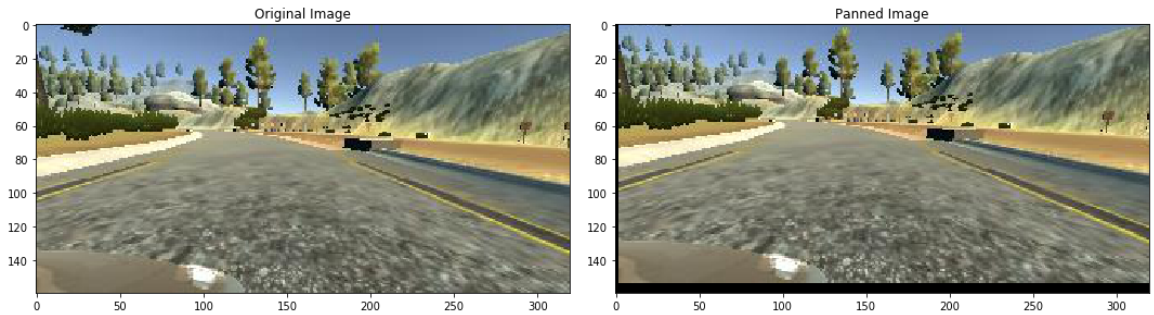

Pan augmentation

In [30]:

```
def pan(image):
  pan = iaa.Affine(translate_percent= {"x" : (-0.1, 0.1), "y": (-0.1, 0.1)})
  image = pan.augment_image(image)
  return image
image = image_paths[random.randint(0, 1000)]
original_image = mpimg.imread(image)
panned_image = pan(original_image)

fig, axs = plt.subplots(1, 2, figsize=(15, 10))
fig.tight_layout()

axs[0].imshow(original_image)
axs[0].set_title('Original Image')

axs[1].imshow(panned_image)
axs[1].set_title('Panned Image')
```

Out[30]:

```
Text(0.5, 1.0, 'Panned Image')
```



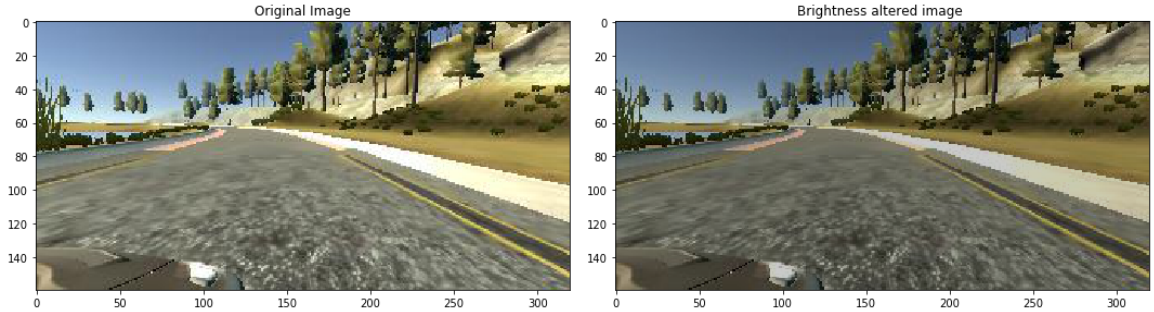

Altered Brightess augmentation

In [31]:

```
def img_random_brightness(image):
    brightness = iaa.Multiply((0.2, 1.2))
    image = brightness.augment_image(image)
    return image
image = image_paths[random.randint(0, 1000)]
original_image = mpimg.imread(image)
brightness_altered_image = img_random_brightness(original_image)

fig, axs = plt.subplots(1, 2, figsize=(15, 10))
fig.tight_layout()

axs[0].imshow(original_image)
axs[0].set_title('Original Image')

axs[1].imshow(brightness_altered_image)
axs[1].set_title('Brightness altered image ')
```

Out[31]:

```
Text(0.5, 1.0, 'Brightness altered image ')
```



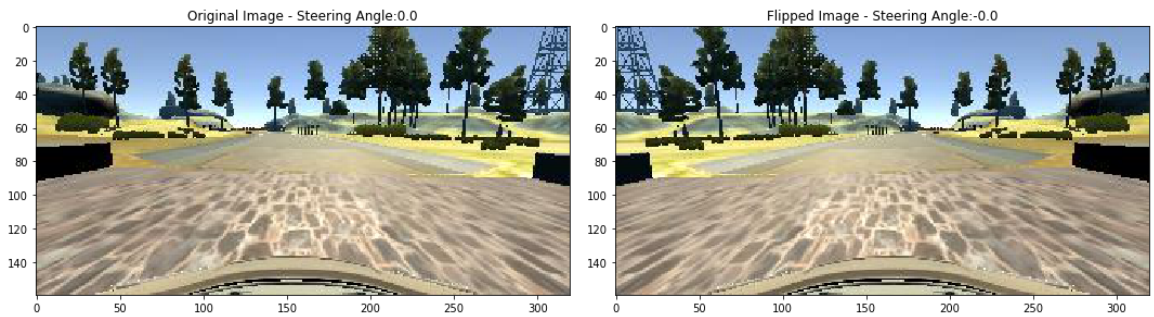

Random Flip Augmentation

In [32]:

```
def img_random_flip(image, steering_angle):
    image = cv2.flip(image,1)
    steering_angle = -steering_angle
    return image, steering_angle
random_index = random.randint(0, 1000)
image = image_paths[random_index]
steering_angle = steerings[random_index]


original_image = mpimg.imread(image)
flipped_image, flipped_steering_angle = img_random_flip(original_image, steering
_angle)

fig, axs = plt.subplots(1, 2, figsize=(15, 10))
fig.tight_layout()

axs[0].imshow(original_image)
axs[0].set_title('Original Image - ' + 'Steering Angle:' + str(steering_angle))

axs[1].imshow(flipped_image)
axs[1].set_title('Flipped Image - ' + 'Steering Angle:' + str(flipped_steering_a
ngle))
```

Out[32]:

```
Text(0.5, 1.0, 'Flipped Image - Steering Angle:-0.0')
```




We do a 50% probablity for selective augmentation on various images

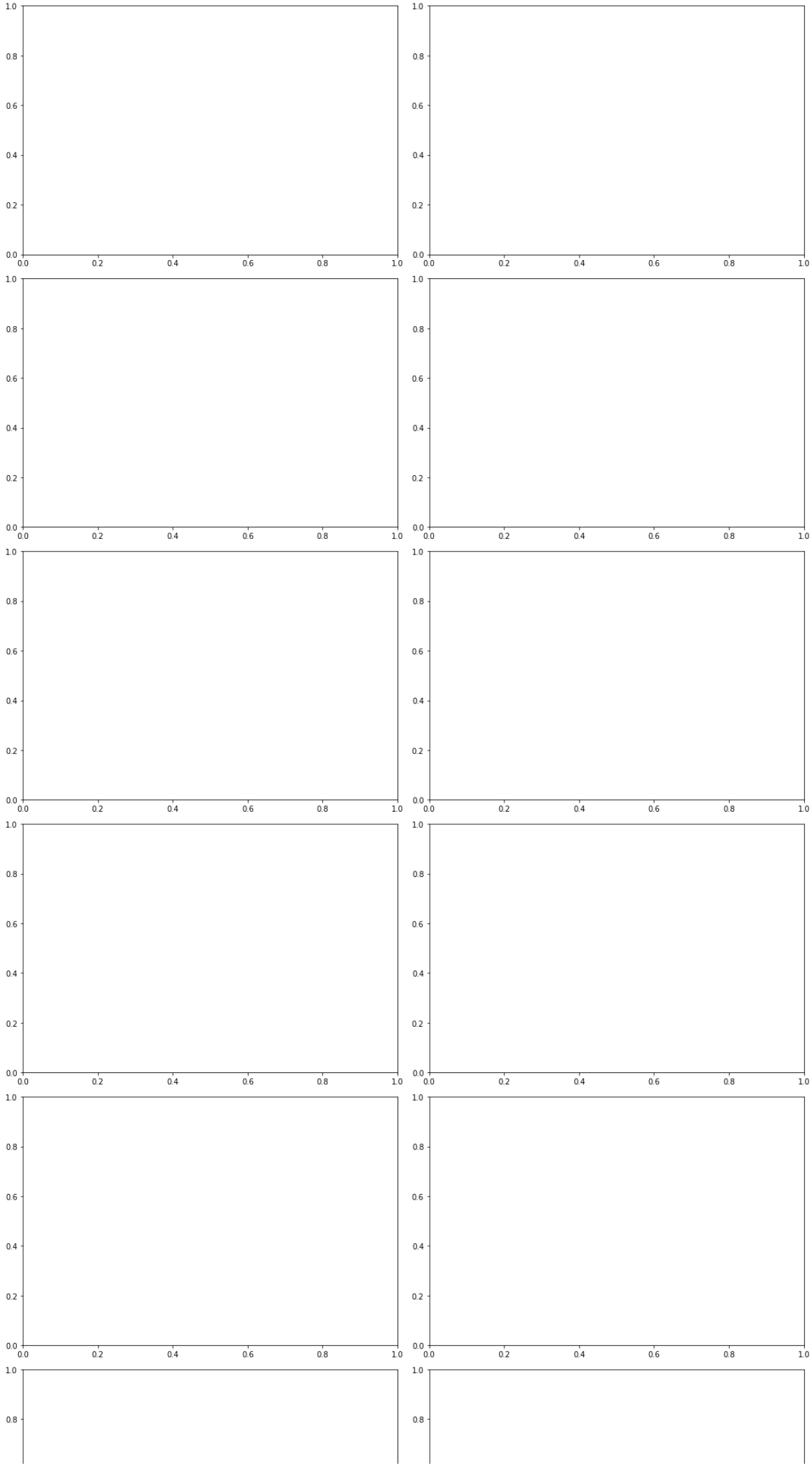
In [33]:

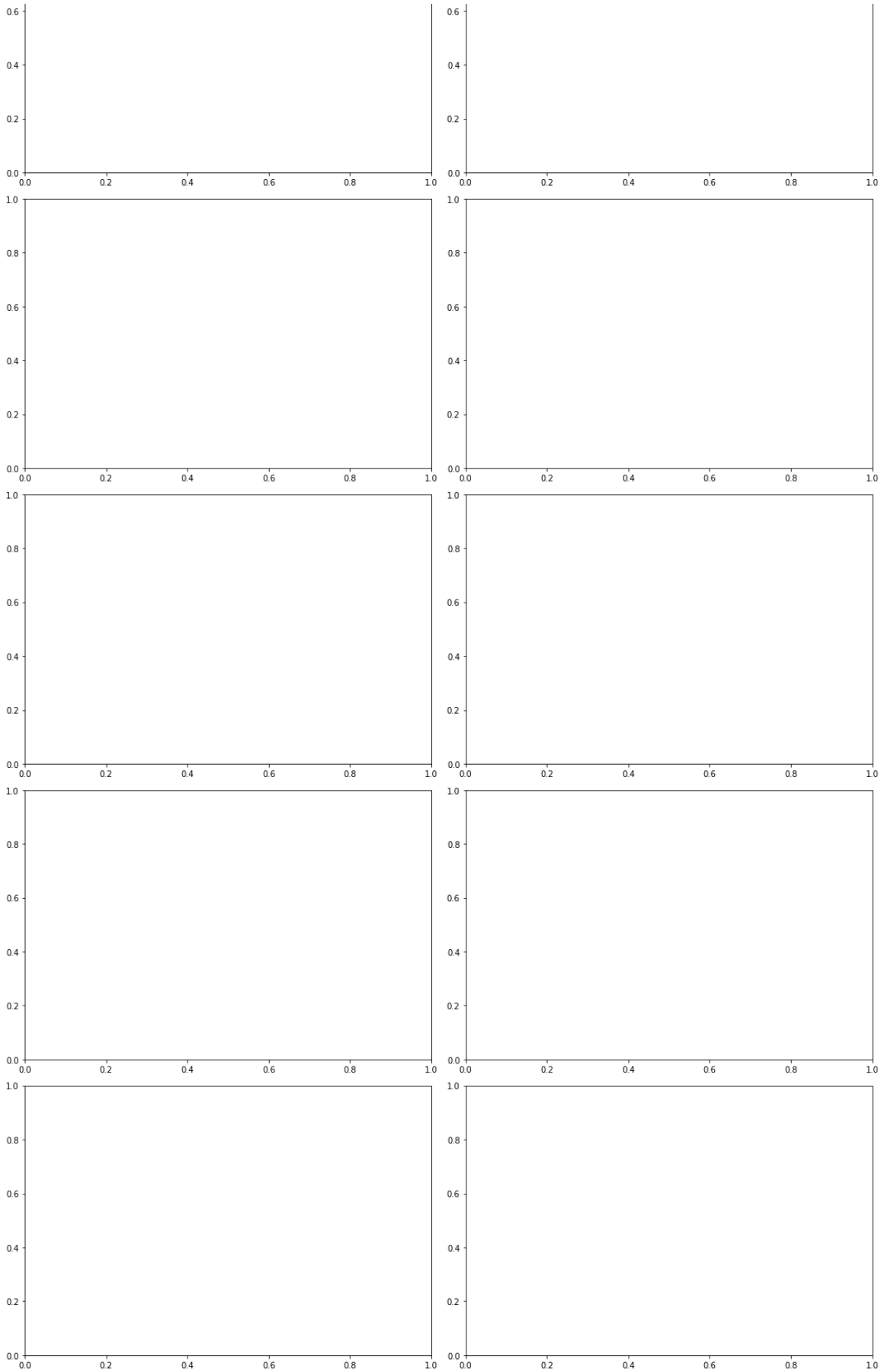
```
def random_augment(image, steering_angle):
    image = mpimg.imread(image)
    if np.random.rand() < 0.5:
      image = pan(image)
    if np.random.rand() < 0.5:
      image = zoom(image)
    if np.random.rand() < 0.5:
      image = img_random_brightness(image)
    if np.random.rand() < 0.5:
      image, steering_angle = img_random_flip(image, steering_angle)

    return image, steering_angle
```

Preparing for plot areas to visualise augmentations

In [34]:

```
ncol = 2
nrow = 10

fig, axs = plt.subplots(nrow, ncol, figsize=(15, 50))
fig.tight_layout()
```

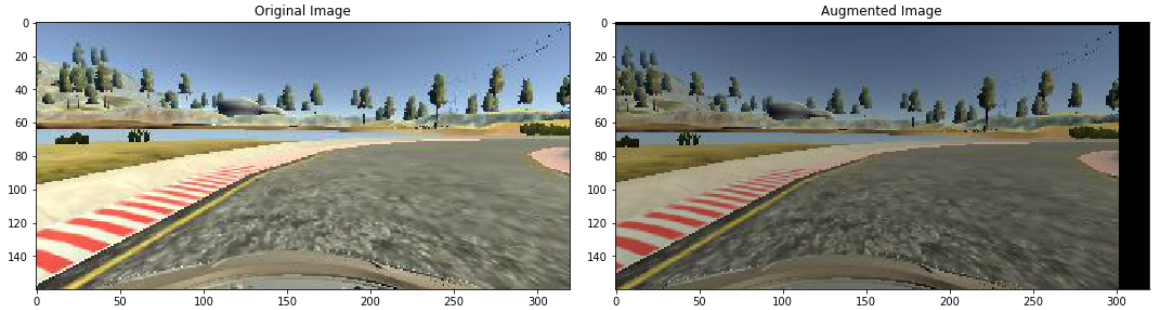

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

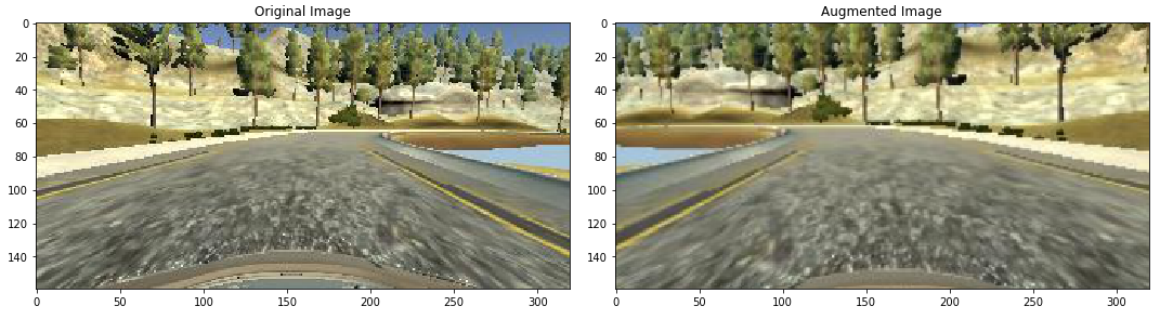


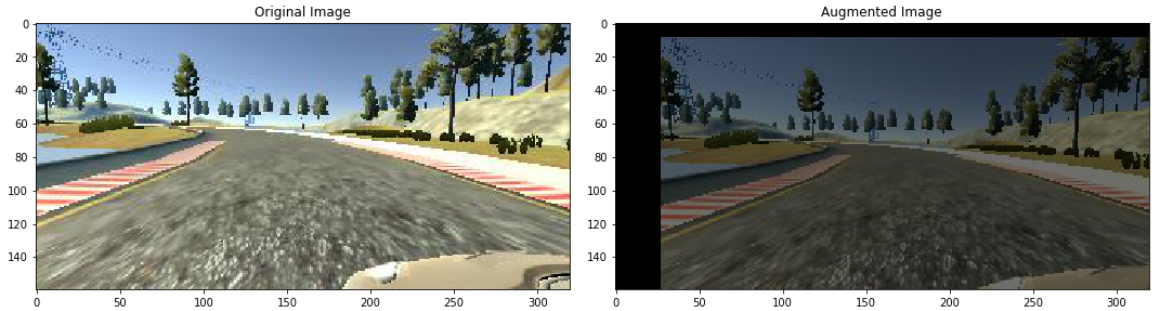
Augmentations visualised

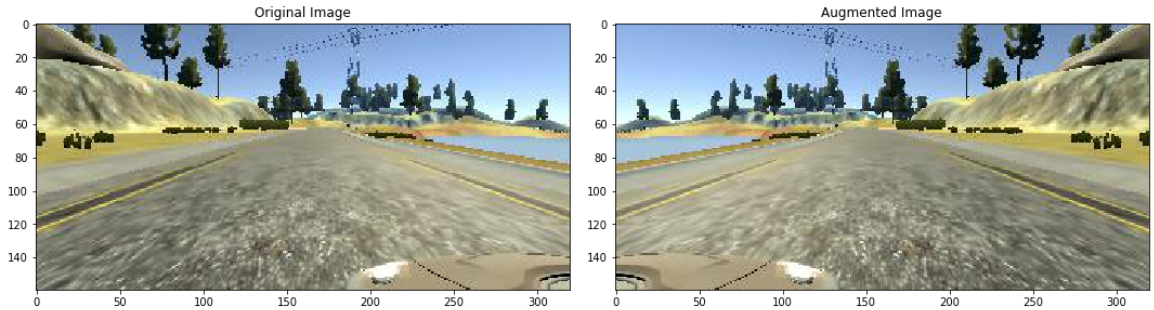
In [35]:

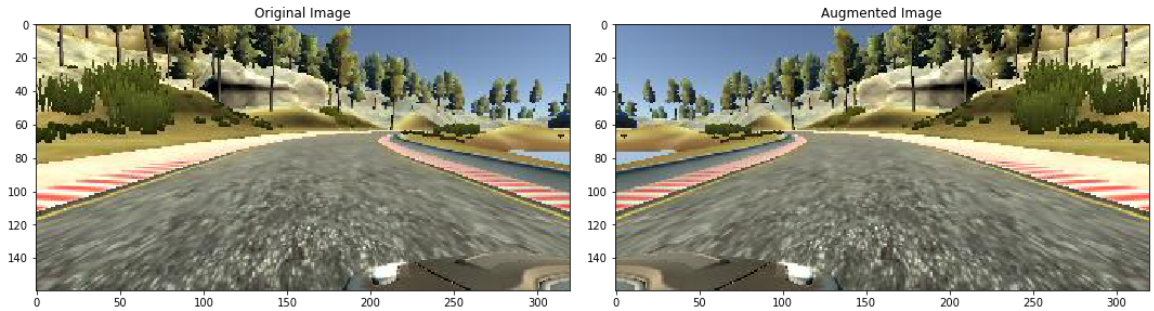
```
ncol = 2
nrow = 10

fig, axs = plt.subplots(nrow, ncol, figsize=(15, 50))
fig.tight_layout()

for i in range(10):
  randnum = random.randint(0, len(image_paths) - 1)
  random_image = image_paths[randnum]
  random_steering = steerings[randnum]

  original_image = mpimg.imread(random_image)
  augmented_image, steering = random_augment(random_image, random_steering)

  axs[i][0].imshow(original_image)
  axs[i][0].set_title("Original Image")

  axs[i][1].imshow(augmented_image)
  axs[i][1].set_title("Augmented Image")
```

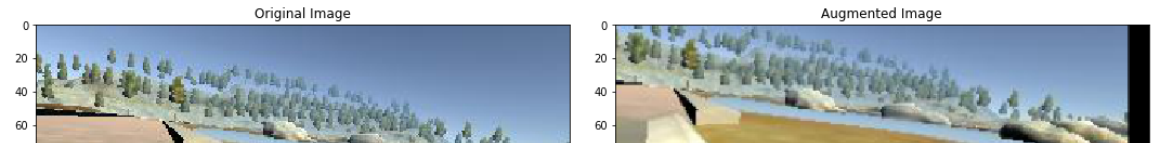

Original Image
Augmented Image

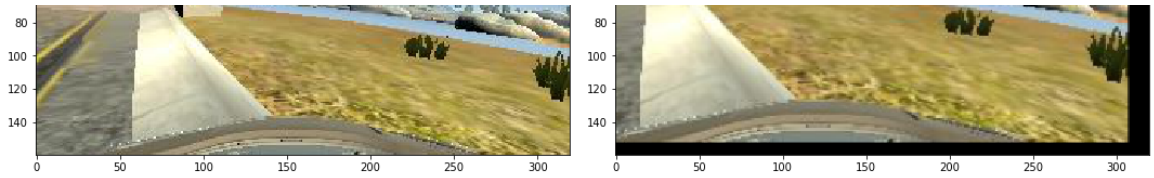

Original Image
Augmented Image

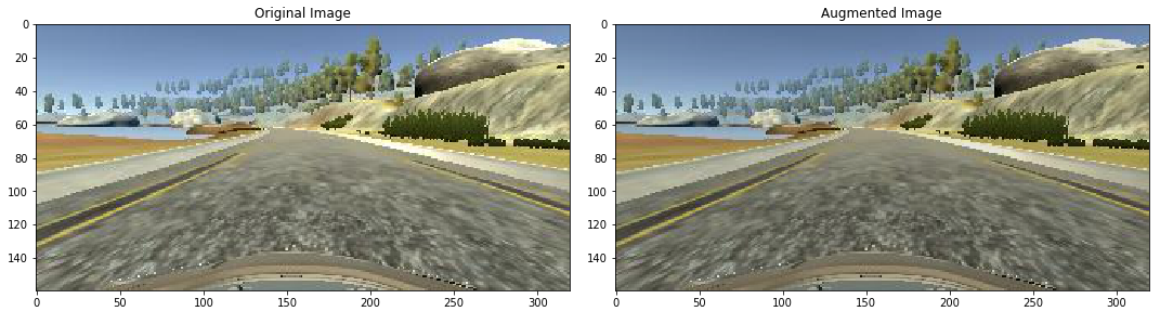

Original Image
Augmented Image

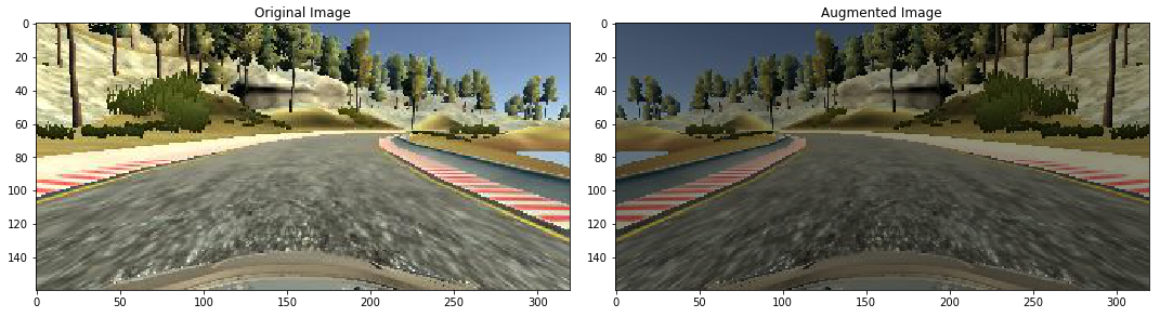

Original Image
Augmented Image

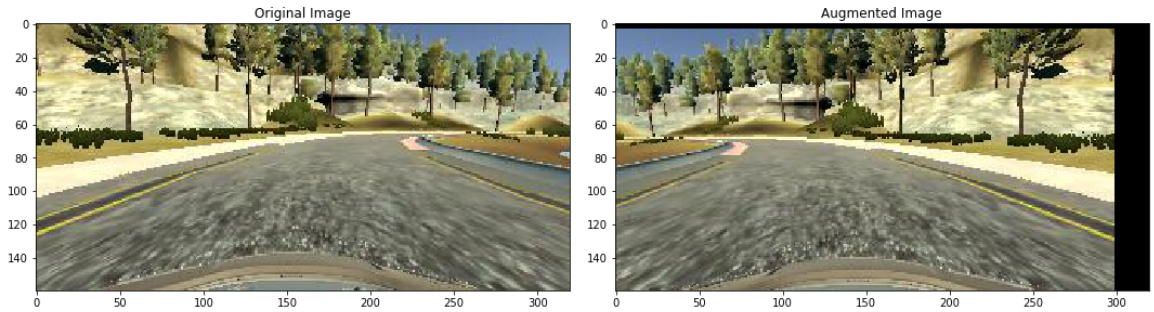

Original Image
Augmented Image

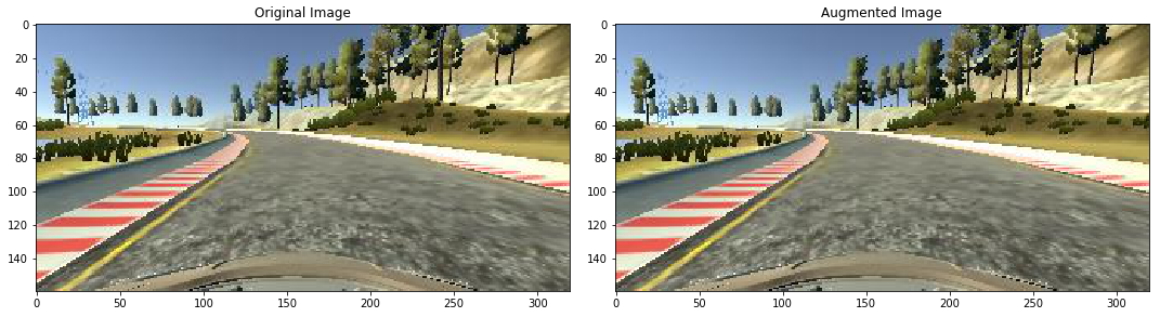

Original Image
Augmented Image






Usual regular Preprocessing of RGB to YUV,Gauss Blurr, and resizing

In [36]:

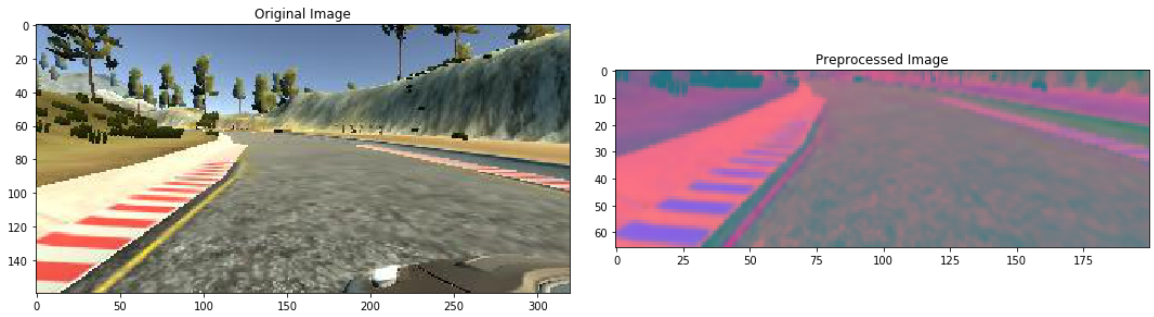
```
def img_preprocess(img):
    img = img[60:135,:,:]
    img = cv2.cvtColor(img, cv2.COLOR_RGB2YUV)
    img = cv2.GaussianBlur(img,  (3, 3), 0)
    img = cv2.resize(img, (200, 66))
    img = img/255
    return img
```

In [37]:

```
image = image_paths[100]
original_image = mpimg.imread(image)
preprocessed_image = img_preprocess(original_image)

fig, axs = plt.subplots(1, 2, figsize=(15, 10))
fig.tight_layout()
axs[0].imshow(original_image)
axs[0].set_title('Original Image')
axs[1].imshow(preprocessed_image)
axs[1].set_title('Preprocessed Image')
```

Out[37]:

```
Text(0.5, 1.0, 'Preprocessed Image')
```




A Realtime batch processor with 'Yield' as return to handle realtime simulation

In [38]:

```
def batch_generator(image_paths, steering_ang, batch_size, istraining):

  while True:
    batch_img = []
    batch_steering = []

    for i in range(batch_size):
      random_index = random.randint(0, len(image_paths) - 1)

      if istraining:
        im, steering = random_augment(image_paths[random_index], steering_ang[ra
ndom_index])

      else:
        im = mpimg.imread(image_paths[random_index])
        steering = steering_ang[random_index]

      im = img_preprocess(im)
      batch_img.append(im)
      batch_steering.append(steering)
    yield (np.asarray(batch_img), np.asarray(batch_steering))
```

Training in batches using batch generator

In [39]:

```
x_train_gen, y_train_gen = next(batch_generator(X_train, y_train, 1, 1))
x_valid_gen, y_valid_gen = next(batch_generator(X_valid, y_valid, 1, 0))
```

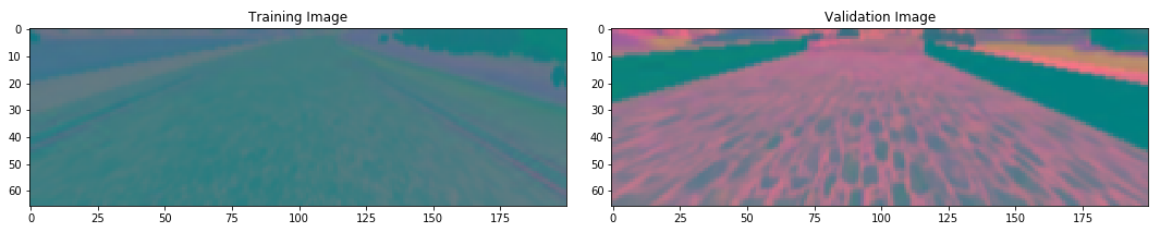
Attemting visual verification of training and validation image from batch generator output

In [40]:

```
fig, axs = plt.subplots(1, 2, figsize=(15, 10))
fig.tight_layout()

axs[0].imshow(x_train_gen[0])
axs[0].set_title('Training Image')

axs[1].imshow(x_valid_gen[0])
axs[1].set_title('Validation Image')
```

Out[40]:

```
Text(0.5, 1.0, 'Validation Image')
```



# Building NVIDIA Model

Thanks to NVIDIA for paper,figs & model are from two nvidia papers and references mentioned. Fully acknowledged, appreciated and thanked for sharing with public.

Title :End to End Learning for Self-Driving Cars

REF :https://arxiv.org/pdf/1604.07316v1.pdf¶ (https://arxiv.org/pdf/1604.07316v1.pdf¶)

***I also found another interesting but advanced paper from NVIDIA***

TItle : Explaining How a Deep Neural Network Trained with End-to-End Learning Steers a Car

https://arxiv.org/pdf/1704.07911.pdf (https://arxiv.org/pdf/1704.07911.pdf)

In [41]:

```
def nvidia_model():
  model = Sequential()
  model.add(Convolution2D(24, 5, 5, subsample=(2, 2), input_shape=(66, 200, 3),
activation='elu'))
  model.add(Convolution2D(36, 5, 5, subsample=(2, 2), activation='elu'))
  model.add(Convolution2D(48, 5, 5, subsample=(2, 2), activation='elu'))
  model.add(Convolution2D(64, 3, 3, activation='elu'))

  model.add(Convolution2D(64, 3, 3, activation='elu'))
#   model.add(Dropout(0.5))


  model.add(Flatten())

  model.add(Dense(100, activation = 'elu'))
#   model.add(Dropout(0.5))

  model.add(Dense(50, activation = 'elu'))
#   model.add(Dropout(0.5))

  model.add(Dense(10, activation = 'elu'))
#   model.add(Dropout(0.5))

  model.add(Dense(1))

  optimizer = Adam(lr=1e-3)
  model.compile(loss='mse', optimizer=optimizer)
  return model
model = nvidia_model()
print(model.summary())
```

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 31, 98, 24)        1824
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 14, 47, 36)        21636
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 5, 22, 48)         43248
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 3, 20, 64)         27712
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 1, 18, 64)         36928
_________________________________________________________________
flatten_1 (Flatten)          (None, 1152)              0
_________________________________________________________________
dense_1 (Dense)              (None, 100)               115300
_________________________________________________________________
dense_2 (Dense)              (None, 50)                5050
_________________________________________________________________
dense_3 (Dense)              (None, 10)                510
_________________________________________________________________
dense_4 (Dense)              (None, 1)                 11
=================================================================
Total params: 252,219
Trainable params: 252,219
Non-trainable params: 0
_________________________________________________________________
None
```

```
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_l
auncher.py:3: UserWarning: Update your `Conv2D` call to the Keras 2
API: `Conv2D(24, (5, 5), input_shape=(66, 200, ..., activation="el
u", strides=(2, 2))`
  This is separate from the ipykernel package so we can avoid doing
imports until
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_l
auncher.py:4: UserWarning: Update your `Conv2D` call to the Keras 2
API: `Conv2D(36, (5, 5), activation="elu", strides=(2, 2))`
  after removing the cwd from sys.path.
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_l
auncher.py:5: UserWarning: Update your `Conv2D` call to the Keras 2
API: `Conv2D(48, (5, 5), activation="elu", strides=(2, 2))`
  """
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_l
auncher.py:6: UserWarning: Update your `Conv2D` call to the Keras 2
API: `Conv2D(64, (3, 3), activation="elu")`

/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_l
auncher.py:8: UserWarning: Update your `Conv2D` call to the Keras 2
API: `Conv2D(64, (3, 3), activation="elu")`
```

Training Time

In [42]:

```
history = model.fit_generator(batch_generator(X_train, y_train, 100, 1),
                                    steps_per_epoch=300,
                                    epochs=10,
                                    validation_data=batch_generator(X_valid, y_val
id, 100, 0),
                                    validation_steps=200,
                                    verbose=1,
                                    shuffle = 1)
```

```
Epoch 1/10
300/300 [==============================] - 343s 1s/step - loss: 0.23
25 - val_loss: 0.1145
Epoch 2/10
300/300 [==============================] - 334s 1s/step - loss: 0.10
38 - val_loss: 0.0730
Epoch 3/10
300/300 [==============================] - 329s 1s/step - loss: 0.07
58 - val_loss: 0.0586
Epoch 4/10
300/300 [==============================] - 330s 1s/step - loss: 0.08
37 - val_loss: 0.0595
Epoch 5/10
300/300 [==============================] - 322s 1s/step - loss: 0.06
75 - val_loss: 0.0577
Epoch 6/10
300/300 [==============================] - 330s 1s/step - loss: 0.05
86 - val_loss: 0.0420
Epoch 7/10
300/300 [==============================] - 325s 1s/step - loss: 0.05
31 - val_loss: 0.0393
Epoch 8/10
300/300 [==============================] - 333s 1s/step - loss: 0.05
01 - val_loss: 0.0357
Epoch 9/10
300/300 [==============================] - 323s 1s/step - loss: 0.04
79 - val_loss: 0.0356
Epoch 10/10
300/300 [==============================] - 332s 1s/step - loss: 0.04
69 - val_loss: 0.0435
```

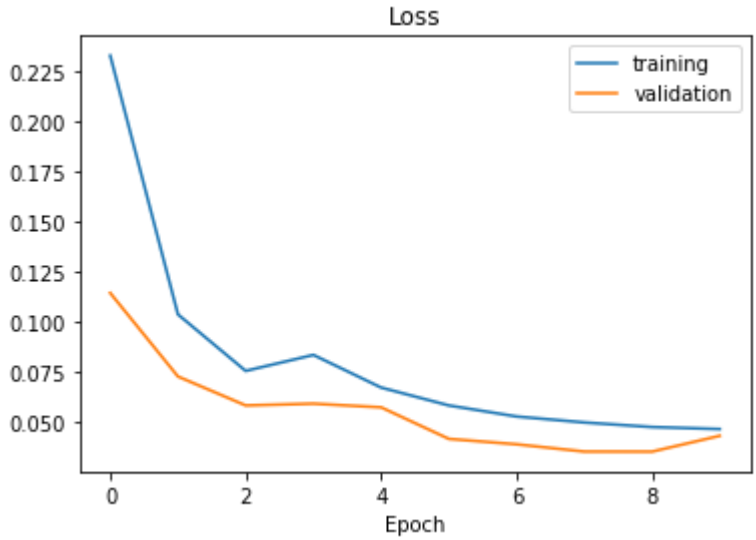
Plotting Model loss

In [43]:

```
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.legend(['training', 'validation'])
plt.title('Loss')
plt.xlabel('Epoch')
```

Out[43]:

```
Text(0.5, 0, 'Epoch')
```




Saving Model to File system.

In [44]:

```
model.save('selected_model_BC1.h5')
```

End