Work IN Progress and EXPERIMENTS for Behavioral Cloning Task

File is a Work In progress and experimentation file

In [157]:
```
!conda list
```

```
# packages in environment at /miniconda3/envs/udrivesimul:
#
# Name                    Version                   Build  Channel
absl-py                   0.7.0                    py36_1000    conda-forge
appnope                   0.1.0                py36hf537a9a_0
asn1crypto                0.24.0                   py36_1003    conda-forge
astor                     0.7.1                      py_0    conda-forge
backcall                  0.1.0                    py36_0
blas                      1.0                         mkl    anaconda
bleach                    3.1.0                    pypi_0    pypi
bzip2                     1.0.6                 h1de35cc_1002    conda-forge
c-ares                    1.15.0                h1de35cc_1001    conda-forge
ca-certificates           2019.1.23                    0
cairo                     1.14.12               h9d4d9ac_1005    conda-forge
certifi                   2018.11.29               py36_0
cffi                      1.11.5               py36h342bebf_1001    conda-forge
click                     7.0                        py_0    conda-forge
cloudpickle               0.7.0                    pypi_0    pypi
cryptography              2.5                  py36ha12b0ac_0
cycler                    0.10.0                     py_1    conda-forge
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-forge
expat                     2.2.6                   h0a44026_0
ffmpeg                    4.1                   heb45b42_1000    conda-forge
flask                     1.0.2                      py_2    conda-forge
fontconfig                2.13.1                h1e4e890_1000    conda-forge
freetype                  2.9.1                    hb4e5f40_0    anaconda
gast                      0.2.1.post0                py_0    conda-forge
gettext                   0.19.8.1              hcca000d_1001    conda-forge
giflib                    5.1.4                 h1de35cc_1001    conda-forge
glib                      2.56.2                h67dad55_1001    conda-forge
gmp                       6.1.2                 h0a44026_1000    conda-forge
gnutls                    3.6.5                 h53004b3_1001    conda-forge
graphite2                 1.3.13                h2098e52_1000    conda-forge
greenlet                  0.4.13                   py36_0    conda-forge
grpcio                    1.16.1               py36h044775b_1
h5py                      2.8.0                py36h878fce3_3    anaconda
harfbuzz                  1.9.0                 h9889186_1001    conda-forge
hdf5                      1.10.2                   hfa1e0ec_1    anaconda
```

```
icu                       58.2                 h0a44026_1000    conda-forge
idna                      2.8                      py36_1000    conda-forge
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-forge
jasper                    1.900.1              h636a363_1006    conda-forge
jedi                      0.13.2                   py36_0
jinja2                    2.10                       py_1    conda-forge
jpeg                      9c                   h1de35cc_1001    conda-forge
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-forge
keras-applications        1.0.4                      py_1    conda-forge
keras-preprocessing       1.0.2                      py_1    conda-forge
kiwisolver                1.0.1           py36h04f5b5a_1002    conda-forge
libcxx                    7.0.0                h2d50403_2    conda-forge
libffi                    3.2.1                h0a44026_1005    conda-forge
libgfortran               3.0.1                h93005f0_2
libgpuarray               0.7.6                h1de35cc_1003    conda-forge
libiconv                  1.15                 h1de35cc_1004    conda-forge
libopenblas               0.3.3                hdc02c5d_3
libpng                    1.6.36               ha441bb4_0    anaconda
libprotobuf               3.6.1                hd9629dc_1000    conda-forge
libsodium                 1.0.16               h3efe00b_0
libtiff                   4.0.10               hcb84e12_2    anaconda
libwebp                   1.0.2                h801f6e5_1    conda-forge
libxml2                   2.9.8                hf14e9c8_1005    conda-forge
llvm-meta                 7.0.0                        0    conda-forge
mako                      1.0.7                      py_1    conda-forge
markdown                  2.6.11                     py_0    conda-forge
markupsafe                1.1.0           py36h1de35cc_1000    conda-forge
matplotlib                3.0.2                 py36_1002    conda-forge
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-forge
mkl_random                1.0.2                 py36_0    conda-forge
nbconvert                 5.3.1                 py36_0
nbformat                  4.4.0              py36h827af21_0
ncurses                   6.1                 h0a44026_1002    conda-forge
nettle                    3.4.1               h1de35cc_1002    conda-forge
networkx                  2.2                    pypi_0    pypi
notebook                  5.7.4                 py36_0
numpy                     1.15.4             py36hacdab7b_0    anaconda
numpy-base                1.15.4             py36h6575580_0    anaconda
olefile                   0.46                  py36_0    anaconda
openblas                  0.3.3               hdc02c5d_1001    conda-forge
opencv                    3.4.1              py36h6fd60c2_1    anaconda
opencv-python             4.0.0.21               pypi_0    pypi
openh264                  1.8.0               hd9629dc_1000    conda-forge
openssl                   1.1.1a                 h1de35cc_0
pandas                    0.24.1             py36h0a44026_0    anaconda
pandoc                    2.2.3.2                       0
pandocfilters             1.4.2                 py36_1
parso                     0.3.2                 py36_0
pcre                      8.41                h0a44026_1003    conda-forge
pexpect                   4.6.0                 py36_0
pickleshare               0.7.5                 py36_0
pillow                    5.4.1              py36hb68e598_0    anaconda
pip                       19.0.1                py36_0
pixman                    0.34.0              h1de35cc_1003    conda-forge
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-forge
pygments                  2.3.1                 py36_0
pygpu                     0.7.6           py36h917ab60_1000      conda-forge
pyopenssl                 19.0.0                py36_0    conda-forge
pyparsing                 2.3.1                   py_0    conda-forge
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-forge
python-socketio           3.1.2                   py_0    conda-forge
pytz                      2018.9                py36_0    anaconda
```

```
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-forge
scikit-image              0.14.2                 pypi_0    pypi
scikit-learn              0.20.2          py36h27c97d8_0    anaconda
scipy                     1.2.0           py36h1410ff5_0    anaconda
send2trash                1.5.0                  py36_0
setuptools                40.8.0                 py36_0    conda-forge
shapely                   1.6.4.post2            pypi_0    pypi
sip                       4.19.8          py36h0a44026_0
six                       1.12.0               py36_1000    conda-forge
sqlite                    3.26.0           h1765d9f_1000    conda-forge
tensorboard               1.10.0                 py36_0    conda-forge
tensorflow                1.10.0                 py36_0    conda-forge
termcolor                 1.1.0                    py_2    conda-forge
terminado                 0.8.1                  py36_1
testpath                  0.4.2                  py36_0
theano                    1.0.3                  py36_0    conda-forge
tk                        8.6.9            ha441bb4_1000    conda-forge
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-forge
wheel                     0.32.3                 py36_0    conda-forge
widgetsnbextension        3.4.2                  py36_0
x264                      1!152.20180717    h1de35cc_1001    conda-forge
xz                        5.2.4             h1de35cc_1001    conda-forge
yaml                      0.1.7             h1de35cc_1001    conda-forge
zeromq                    4.3.1                h0a44026_3
zlib                      1.2.11            h1de35cc_1004    conda-forge
zstd                      1.3.7                h5bba6e5_0    anaconda
```

In [158]:
```
import numpy as np
import matplotlib.pyplot as plt
import keras
from keras.models import Sequential
from keras.optimizers import Adam
from keras.layers import Convolution2D, MaxPooling2D,Dropout,Flatten,Dense
import cv2
import pandas as pd
import random
import os
import ntpath
from sklearn.utils import shuffle
from sklearn.model_selection import train_test_split
import matplotlib.image as mpimg
import sys
from imgaug import augmenters as iaa
```

In [159]:
```
datadir = 'IMG'
columns =['center','left','right','steering','throttle','reverse','speed']
data=pd.read_csv('driving_log.csv',names=columns)
data.head
```

```
Out[159]: <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	throttle	\
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]>
```

In [160]:
```
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[160]:

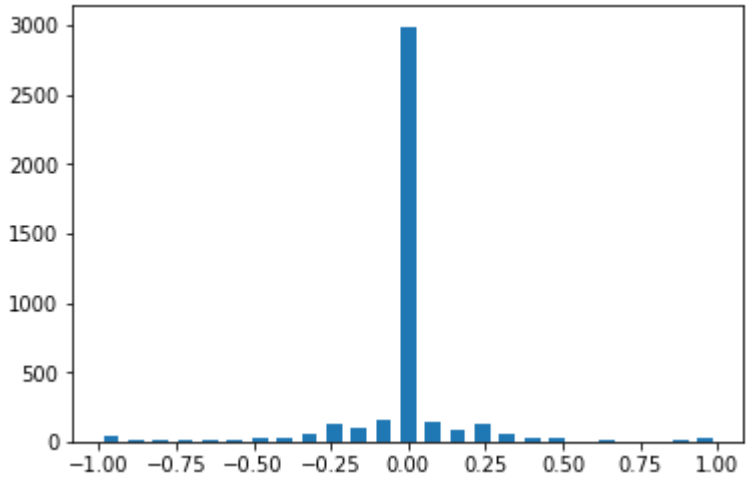
	center	left	right	steering	throttle	revers
0	center_2018_07_16_17_11_43_382.jpg	left_2018_07_16_17_11_43_382.jpg	right_2018_07_16_17_11_43_382.jpg	0.0	0.0	0.0
1	center_2018_07_16_17_11_43_670.jpg	left_2018_07_16_17_11_43_670.jpg	right_2018_07_16_17_11_43_670.jpg	0.0	0.0	0.0
2	center_2018_07_16_17_11_43_724.jpg	left_2018_07_16_17_11_43_724.jpg	right_2018_07_16_17_11_43_724.jpg	0.0	0.0	0.0
3	center_2018_07_16_17_11_43_792.jpg	left_2018_07_16_17_11_43_792.jpg	right_2018_07_16_17_11_43_792.jpg	0.0	0.0	0.0
4	center_2018_07_16_17_11_43_860.jpg	left_2018_07_16_17_11_43_860.jpg	right_2018_07_16_17_11_43_860.jpg	0.0	0.0	0.0

In [161]:
```
#which steering angles are most frequent?, let us visualize with a histogram
num_bins = 25
hist,bins = np.histogram(data['steering'],num_bins)
print(bins)
```

```
[-1.   -0.92 -0.84 -0.76 -0.68 -0.6  -0.52 -0.44 -0.36 -0.28 -0.2  -0.12
 -0.04  0.04  0.12  0.2   0.28  0.36  0.44  0.52  0.6   0.68  0.76  0.84
  0.92  1.  ]
```

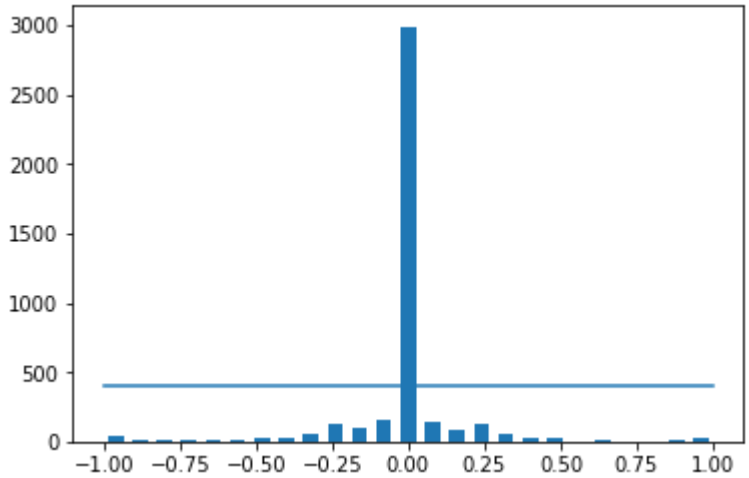
In [162]:
```
#Not centered around 0 ie middle of the road so let us do it.
num_bins = 25

hist,bins = np.histogram(data['steering'],num_bins)
center = (bins[:-1]+ bins[1:]) *0.5
plt.bar(center,hist,width=0.05)
#freq of each steering angle alongthe track
#left and right is balanced since we drove on both directions
#Nature of given uda track is more left turns, it could lead to biased cloning
```

Out[162]: <BarContainer object of 25 artists>



In [163]:
```
#Let us avoid too much bias towards 0 deg angle
# Every bin can have a max of 200 samples only
num_bins=25
#samples_per_bin = 200 after first test of car, account bias for straight line driving
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[163]: [<matplotlib.lines.Line2D at 0x1034bfe9e8>]



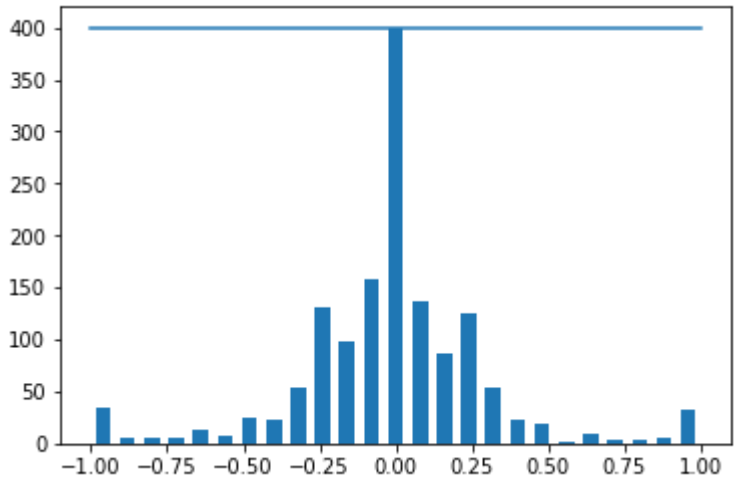
In [164]:
```
# Let us balance the data, anyway data is skewed to middle,
# it is good to drive in straight line in straight way but got to make sure its not too much biased

#When we set threshold for data for particular steering angle bin, we will have to reject data beyond the thr
eshold, but how ?
#If we list and delete last ramaining data after threshold, it may not work, because it will correspond to pa
rticular set of data from
#specific part of the track. Doing it may leave out with confusion to drive in that part ie scenario edge con
ditions.
#so we randomise, shuffle and handle it through a cut, so info is preserved and we are not biased.
print('total data before removal', len(data))
```

```
total data before removal 4053
```

In [165]:
```
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))

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))
```

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

Out[165]:
```
[<matplotlib.lines.Line2D at 0x1034c652e8>]
```



In [166]:
```
#Now We got much more uniform driving angle in samples
#So we now understand the need for recovery modes and driving simulation for recovery laps, such training, he
lps
```

In [167]:
```
#Now Training data and validation data split
```

In [168]:
```
print(data.iloc[1])
def load_img_steering(datadir, df):
    image_path = []
    steering = []
    for i in range (len(data)):
        indexed_data = data.iloc[i]
# Row at index 1 that corresponds a specific steering angle. Now (for that row) let us get the center,left an
d right from it
```

```
center      center_2018_07_16_17_11_44_413.jpg
left          left_2018_07_16_17_11_44_413.jpg
right        right_2018_07_16_17_11_44_413.jpg
steering                                -0.05
throttle                             0.642727
reverse                                     0
speed                                1.43401
Name: 12, dtype: object
```

In [169]:
```
print(data.iloc[1])
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


#two arrays, array for images and for steering angles
```

```
center      center_2018_07_16_17_11_44_413.jpg
left          left_2018_07_16_17_11_44_413.jpg
right        right_2018_07_16_17_11_44_413.jpg
steering                                -0.05
throttle                              0.642727
reverse                                     0
speed                                 1.43401
Name: 12, dtype: object
```

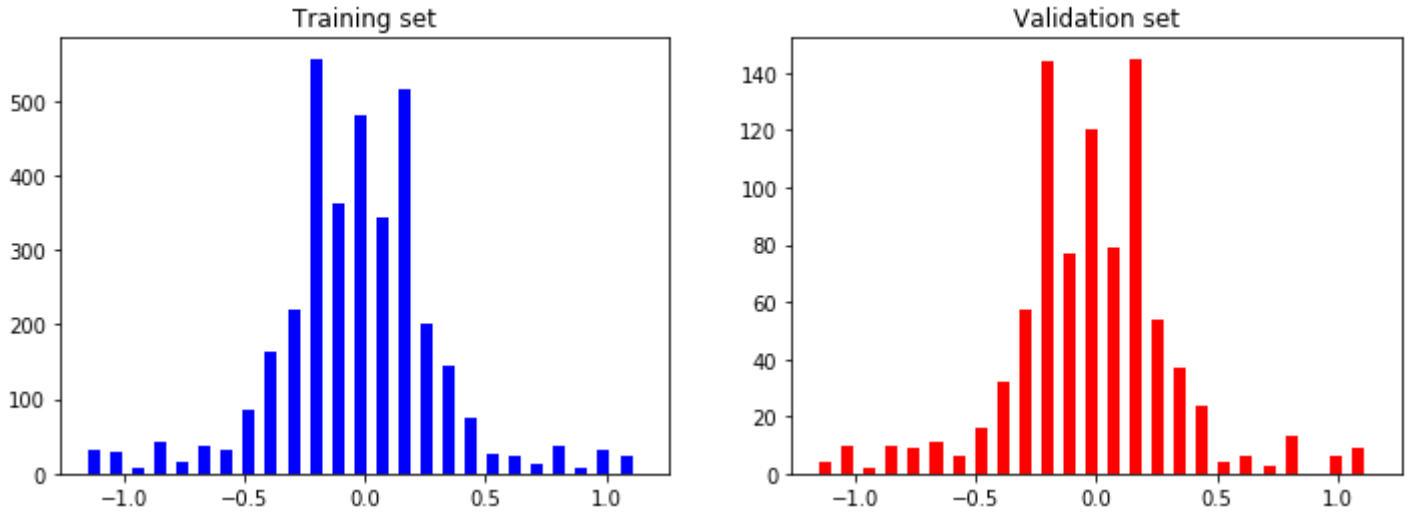
In [170]:
```
image_paths, steerings = load_img_steering(datadir,data)
print(image_paths)
```

```
['IMG/center_2018_07_16_17_11_44_342.jpg'
 'IMG/left_2018_07_16_17_11_44_342.jpg'
 'IMG/right_2018_07_16_17_11_44_342.jpg' ...
 'IMG/center_2018_07_16_17_16_30_492.jpg'
 'IMG/left_2018_07_16_17_16_30_492.jpg'
 'IMG/right_2018_07_16_17_16_30_492.jpg']
```

In [171]:
```
# art of selecting randomizer state by iterative experience

X_train, X_valid, y_train, y_valid = train_test_split(image_paths, steerings, test_size=0.2, random_state=6)
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[171]:
```
Text(0.5, 1.0, 'Validation set')
```

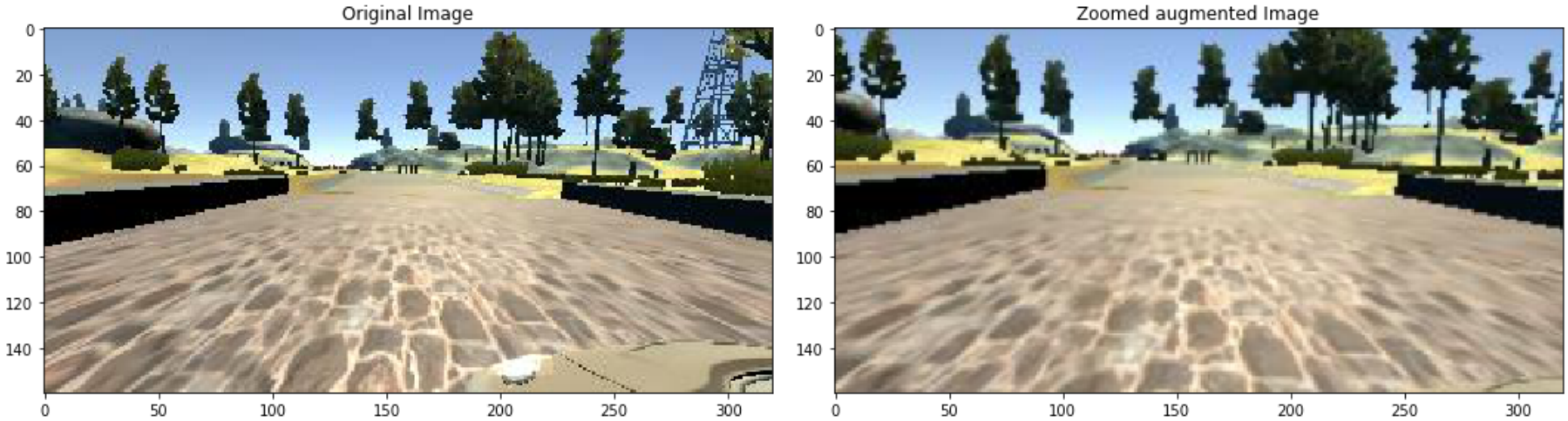


***** Insertion -Adding Augmentation after first round of testing with ELU*****

In [66]:
```
def zoom(image) :
    zoom = iaa.Affine(scale=(1,1.3))
    image = zoom.augment_image(image)
    return image
```

In [68]:
```
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 augmented Image')
```

Out[68]: Text(0.5, 1.0, 'Zoomed augmented Image')

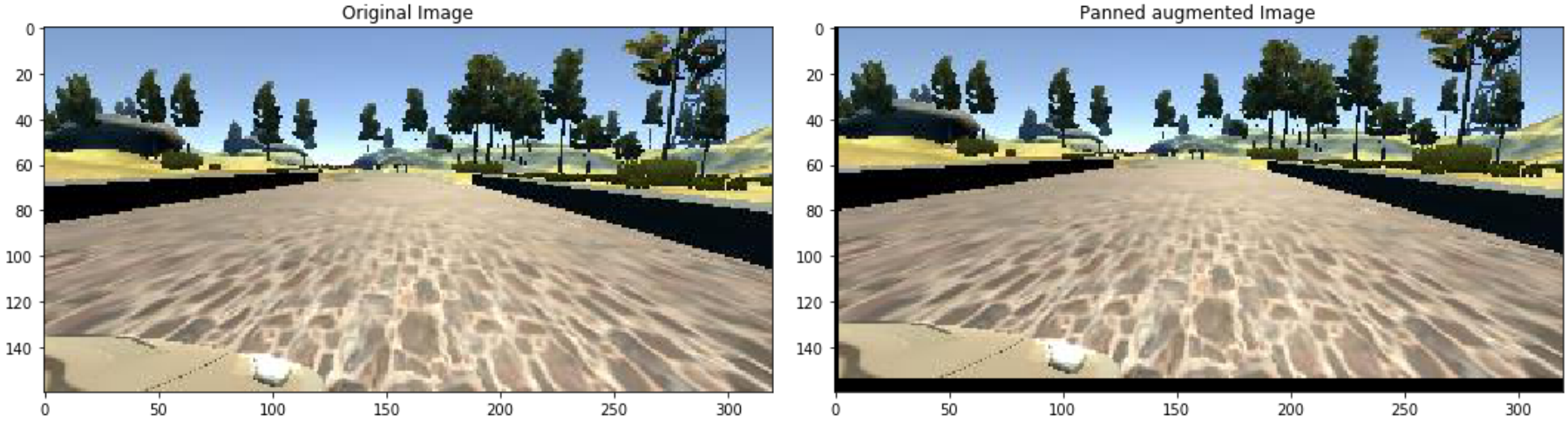


***** Insertion -Adding Augmentation PAN after first round of testing with ELU***

In [74]:
```
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
```

In [75]:
```
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 augmented Image')
```

Out[75]: Text(0.5, 1.0, 'Panned augmented Image')

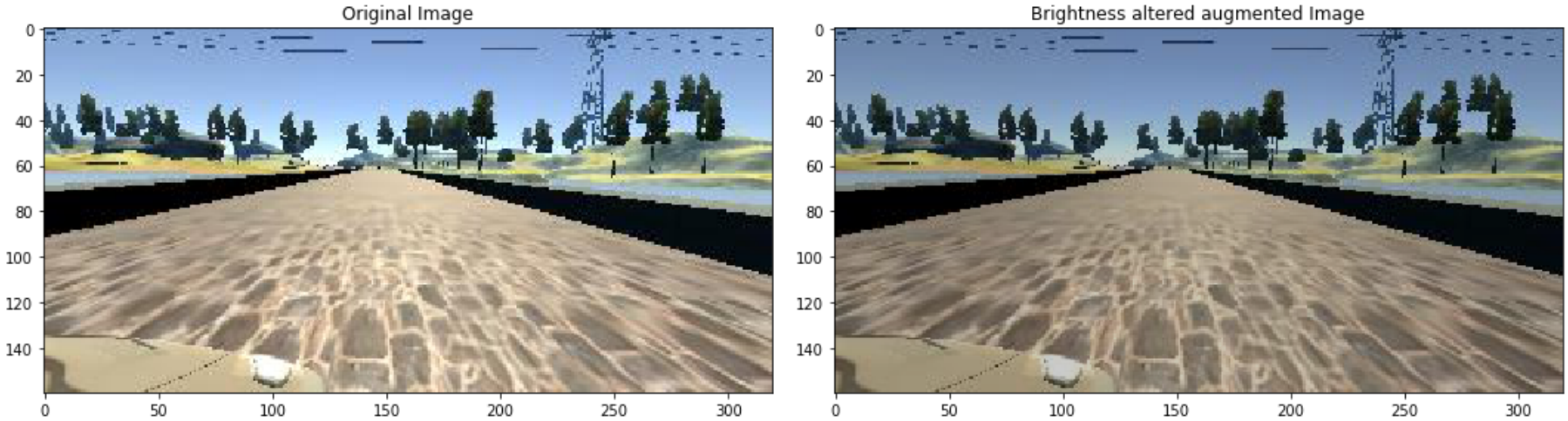


***** Insertion -Adding Augmentation Brightness Change- after first round of testing with ELU***

In [84]:
```
def img_random_brightess(image):
    brightness = iaa.Multiply((0.2,1.2))
    image = brightness.augment_image(image)
    return image
```

In [85]:
```
image = image_paths[random.randint(0,1000)]
original_image = mpimg.imread(image)
brightness_altered_image = img_random_brightess (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 augmented Image')
```

Out[85]: Text(0.5, 1.0, 'Brightness altered augmented Image')

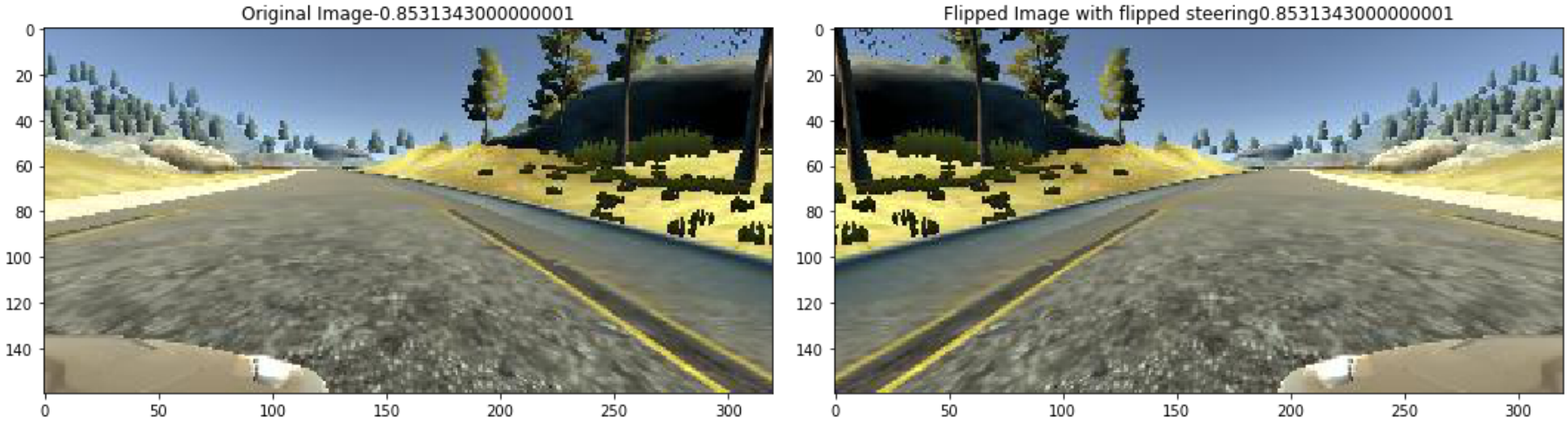


***** Insertion -Adding Augmentation Flipping Change- after first round of testing with ELU***

In [94]:
```
def img_random_flip(image , steering_angle):
    image = cv2.flip(image, 1 )
    steering_angle = -steering_angle
    return image, steering_angle
```

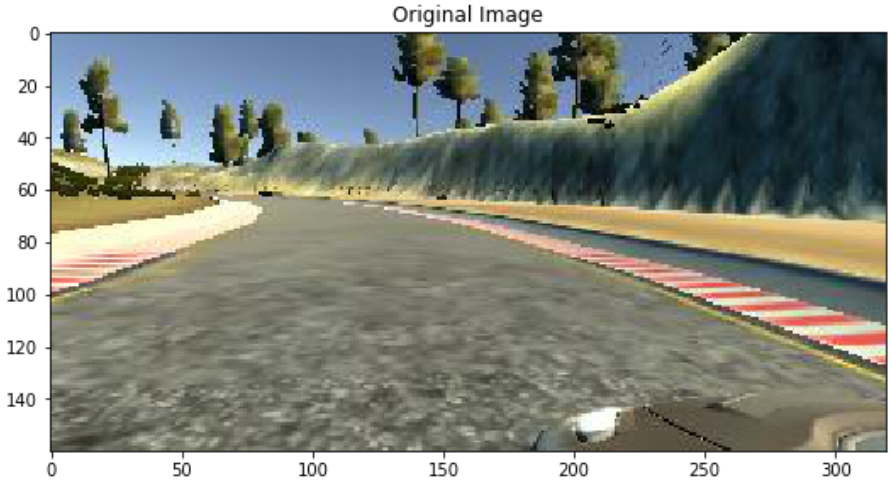
In [95]:
```
random_index = random.randint(0,1000)
image = image_paths[random_index]
steering_angle = steerings[random_index]


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'+str(steering_angle))

axs[1].imshow(flipped_image)
axs[1].set_title('Flipped Image with flipped steering'+ str(flipped_steering_angle))
```

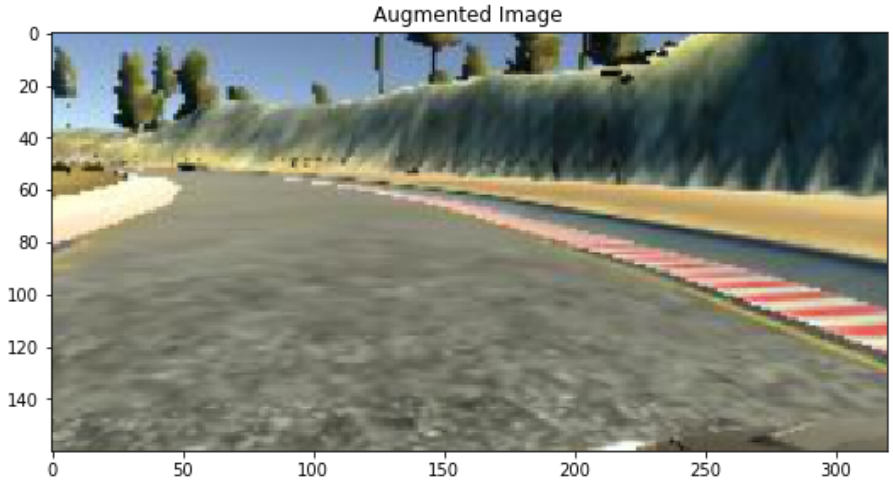
Out[95]:
```
Text(0.5, 1.0, 'Flipped Image with flipped steering0.8531343000000001')
```

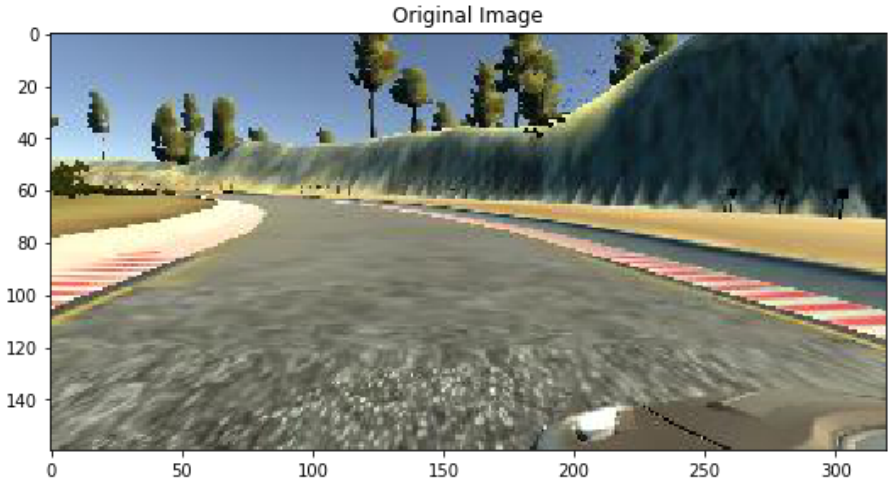


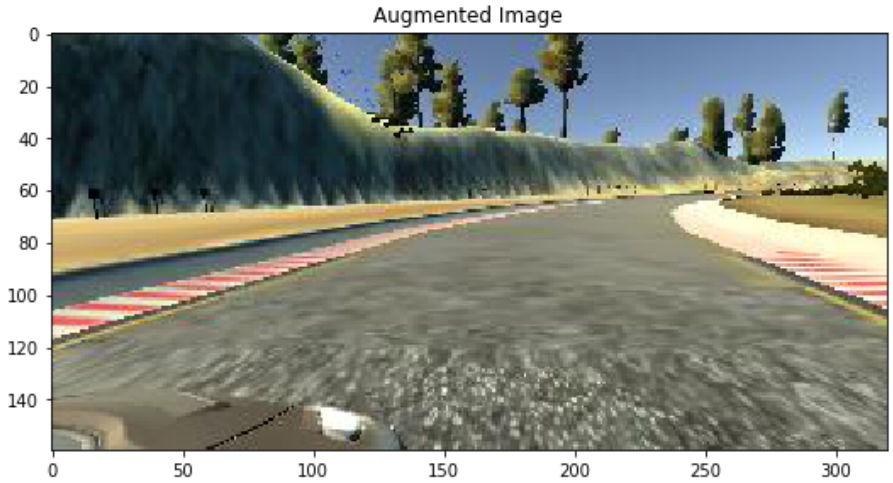
In [106]:
```
def random_augment(image, steering_angle):
    image = mpimg.imread(image)
    if np.random.rand() < 0.5:
      image = pan(image)
    if np.random.rand() < 0.5:
      image = zoom(image)
    if np.random.rand() < 0.5:
      image = img_random_brightess(image)
    if np.random.rand() < 0.5:
      image, steering_angle = img_random_flip(image, steering_angle)

    return image, steering_angle
```

In [107]:
```
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
0
20
40
60
80
100
120
140
0
50
100
150
200
250
300

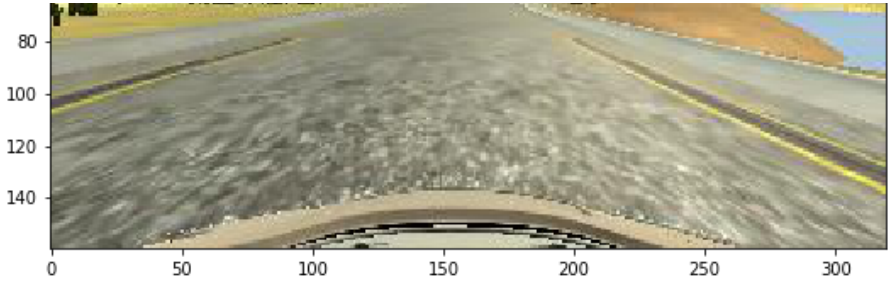

Augmented Image
0
20
40
60
80
100
120
140
0
50
100
150
200
250
300

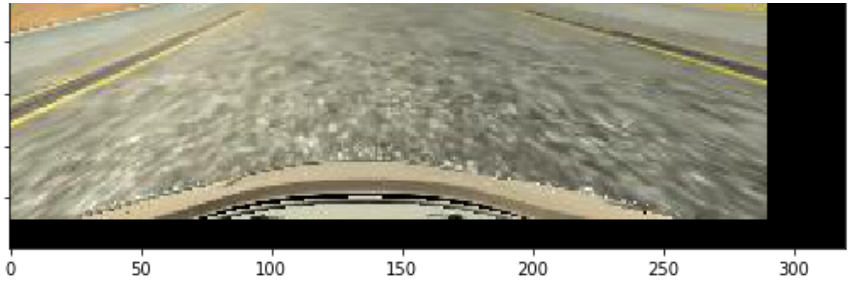

Original Image
0
20
40
60
80
100
120
140
0
50
100
150
200
250
300

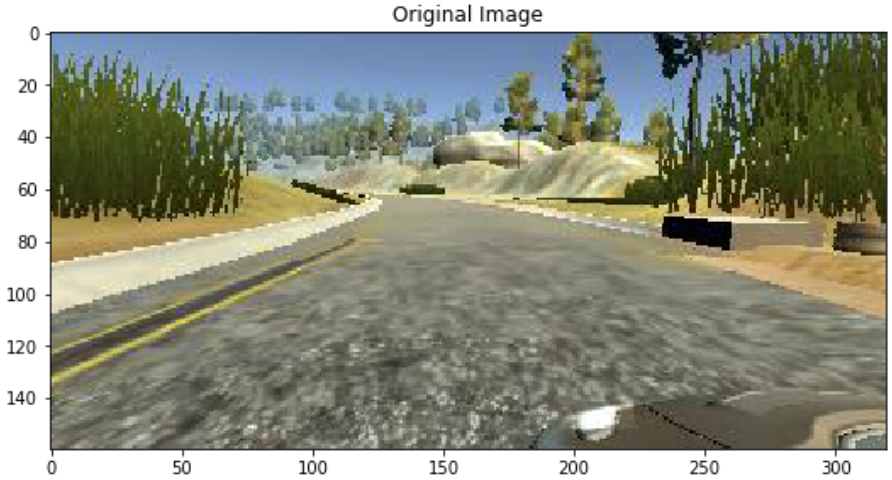

Augmented Image
0
20
40
60
80
100
120
140
0
50
100
150
200
250
300

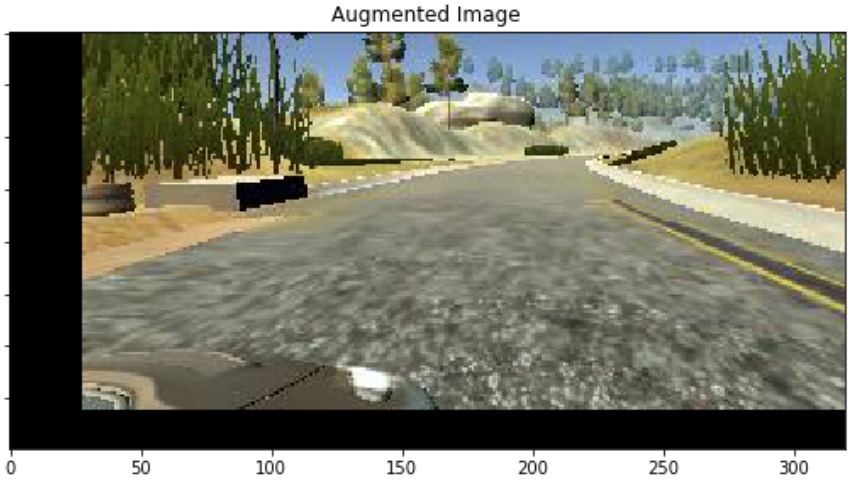

Original Image
0
20
40
60


Augmented Image
0
20
40
60



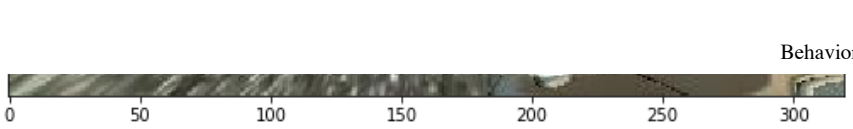


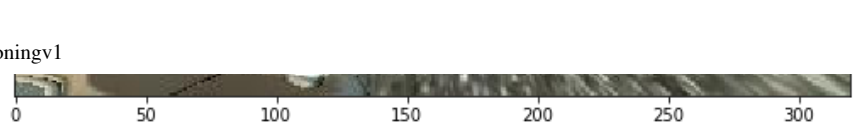

Original Image

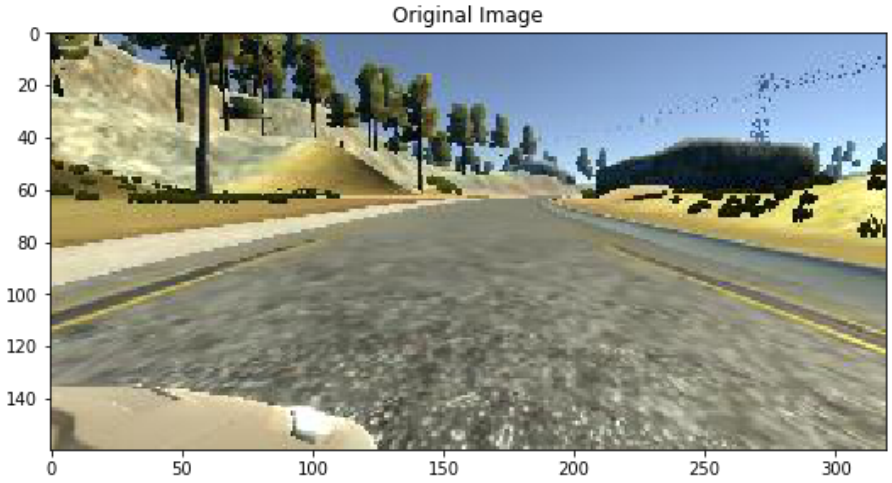

Augmented Image

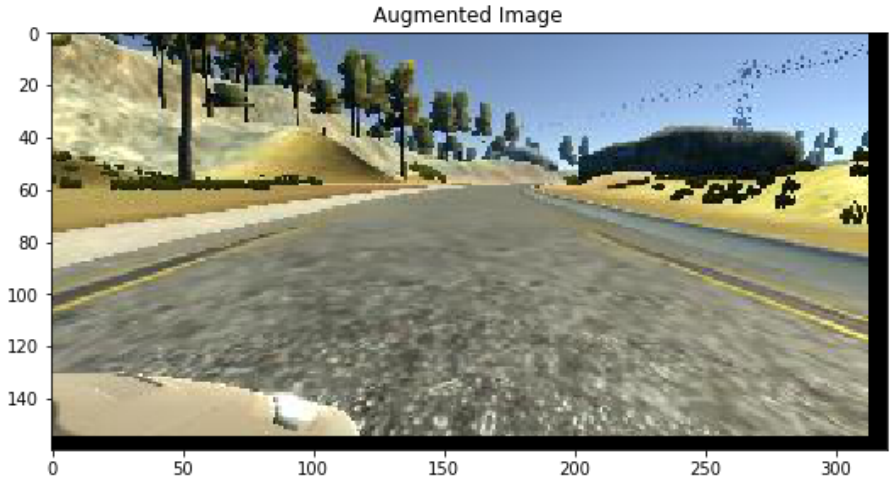

Original Image

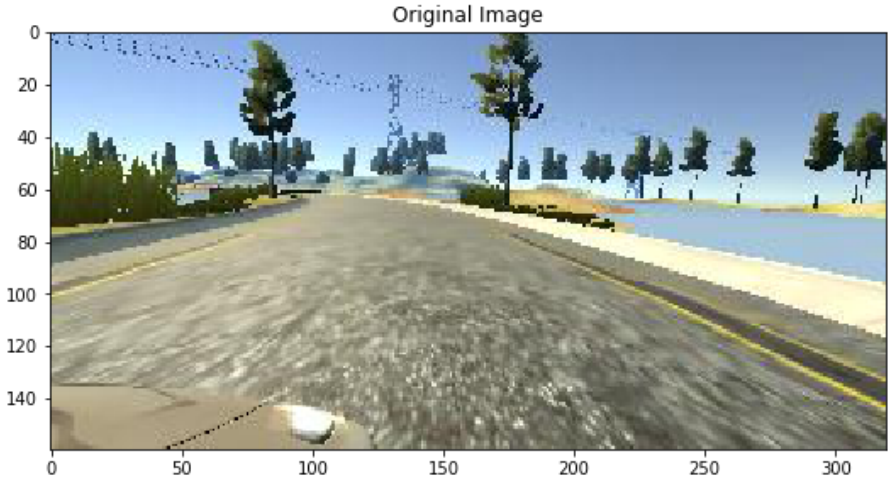

Augmented Image

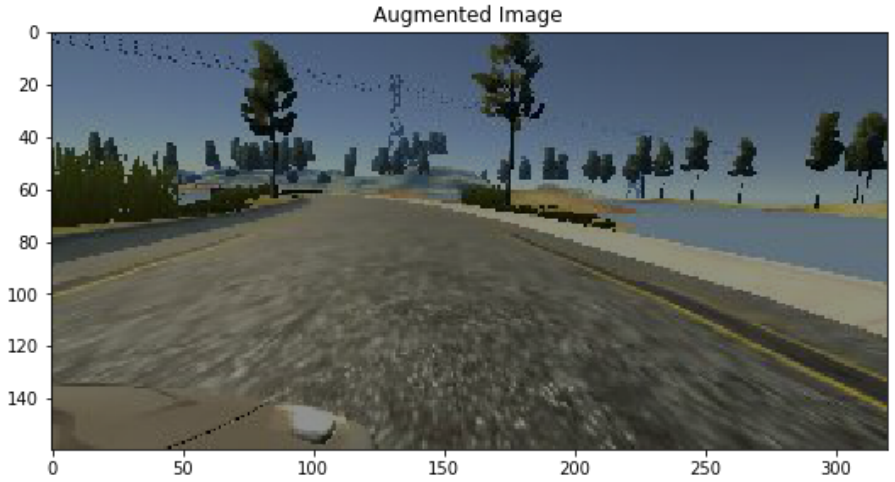

0
50
100
150
200
250
300


0
50
100
150
200
250
300


Original Image
0
20
40
60
80
100
120
140
0
50
100
150
200
250
300

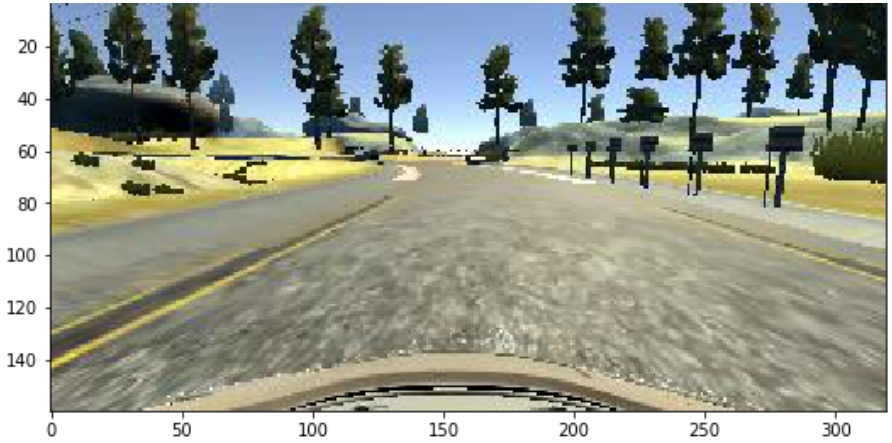

Augmented Image
0
20
40
60
80
100
120
140
0
50
100
150
200
250
300

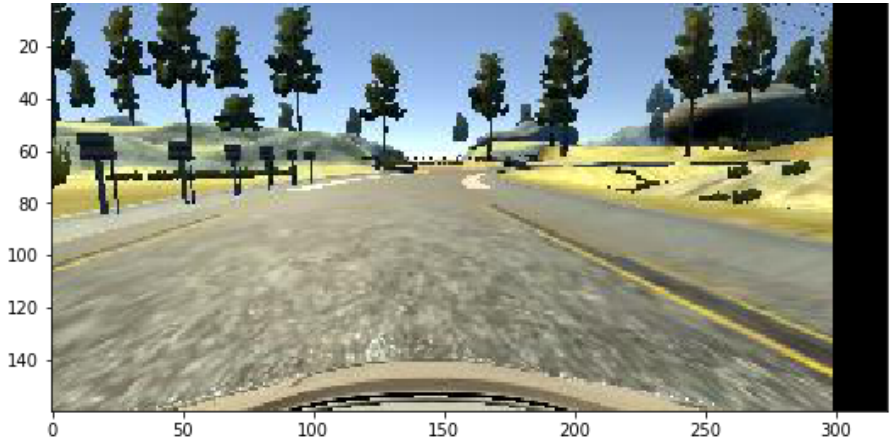

Original Image
0
20
40
60
80
100
120
140
0
50
100
150
200
250
300

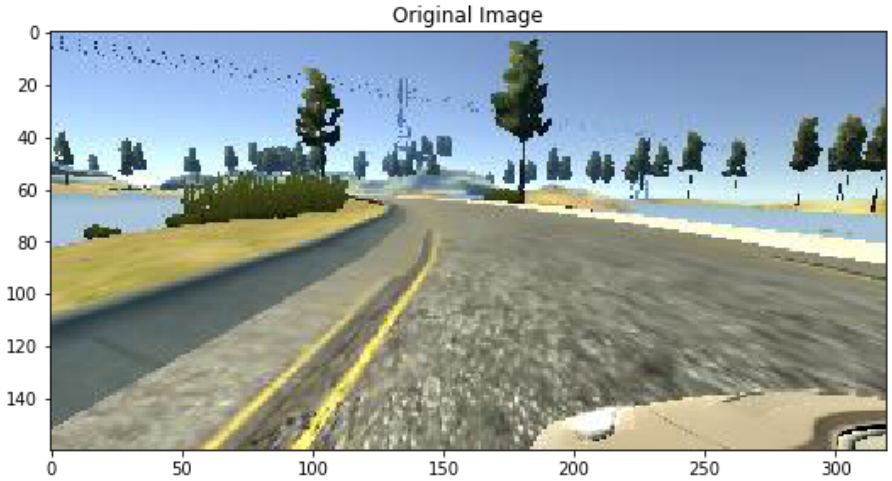

Augmented Image
0
20
40
60
80
100
120
140
0
50
100
150
200
250
300

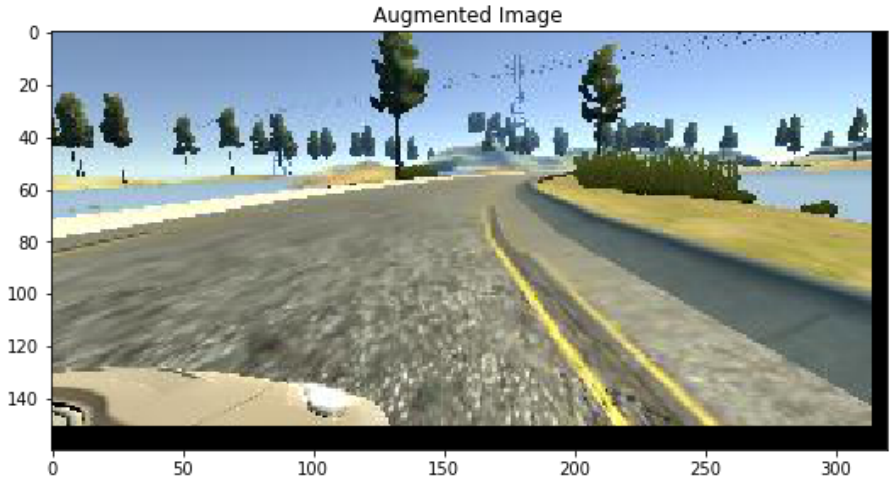

Original Image
0

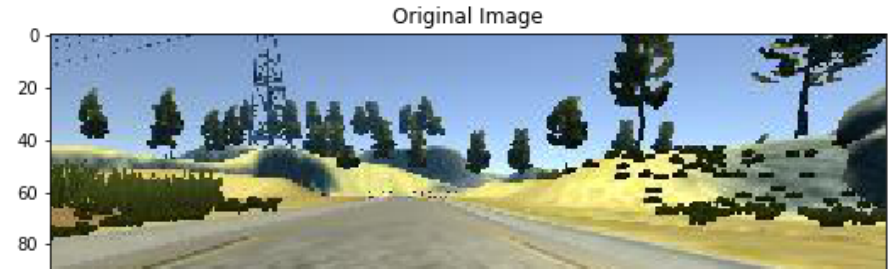

Augmented Image
0

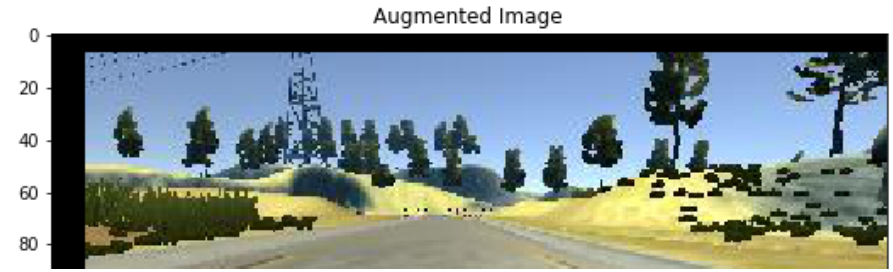


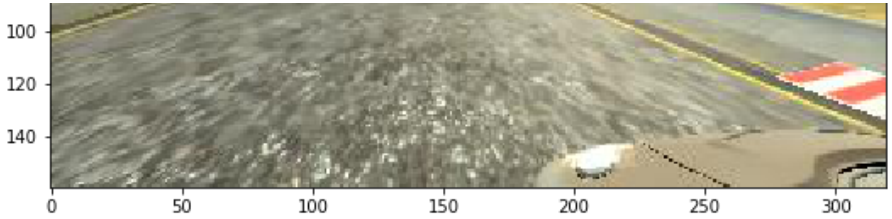


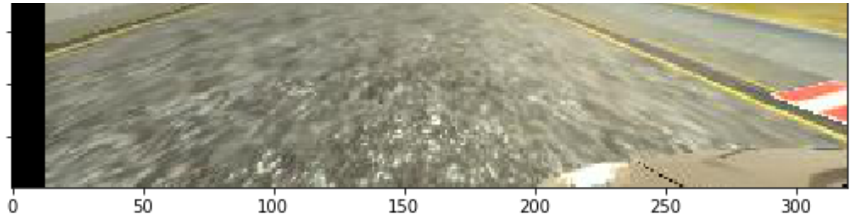

Original Image


Augmented Image


Original Image


Augmented Image





Insertion after ELU test ends

Now Let us PreProcess our data

def img_preprocess(img) : img = mpimg.imread(img) return img

In [22]:
```
print(datadir)
```

```
IMG
```

In [23]:
```
image = image_paths[100]
print(image)
print(image_paths)
original_image = mpimg.imread(image)
preprocessed_image = img_preprocess(image)
```

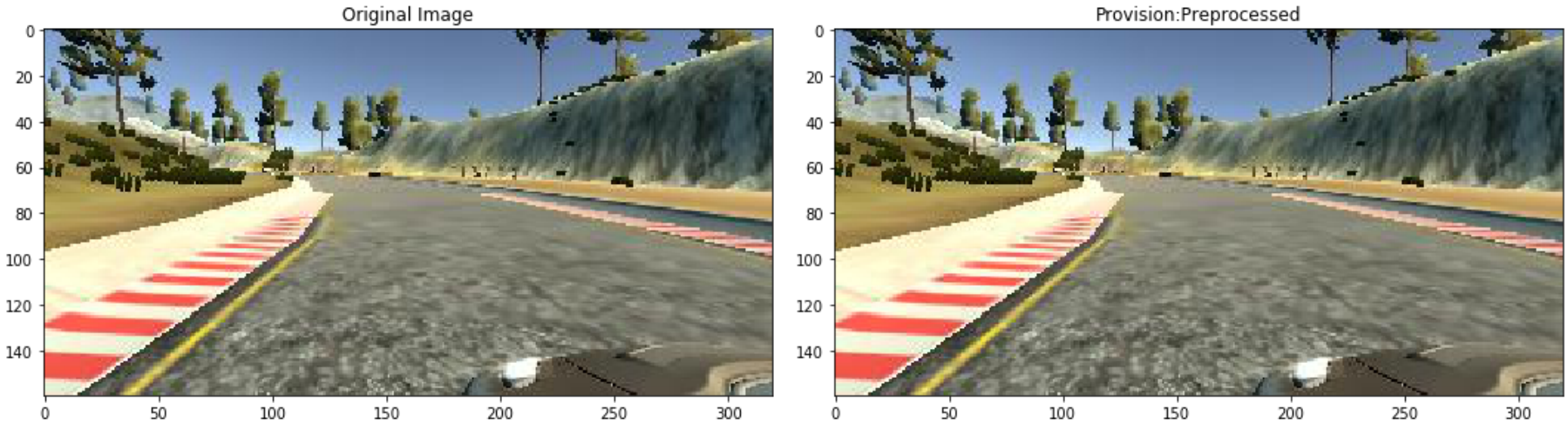
```
IMG/left_2018_07_16_17_11_47_636.jpg
['IMG/center_2018_07_16_17_11_44_413.jpg'
 'IMG/left_2018_07_16_17_11_44_413.jpg'
 'IMG/right_2018_07_16_17_11_44_413.jpg' ...
 'IMG/center_2018_07_16_17_16_30_345.jpg'
 'IMG/left_2018_07_16_17_16_30_345.jpg'
 'IMG/right_2018_07_16_17_16_30_345.jpg']
```

In [24]:
```
if os.path.isfile(image):
    img1 = cv2.imread(image, 0)
    print ("The file " + image + " is found in dir.")
else:
    print ("The file " + image + " does not exist.")
```

```
The file IMG/left_2018_07_16_17_11_47_636.jpg is found in dir.
```

In [25]:
```
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(original_image)
axs[1].set_title('Provision:Preprocessed')
```

Out[25]:
```
Text(0.5, 1.0, 'Provision:Preprocessed')
```

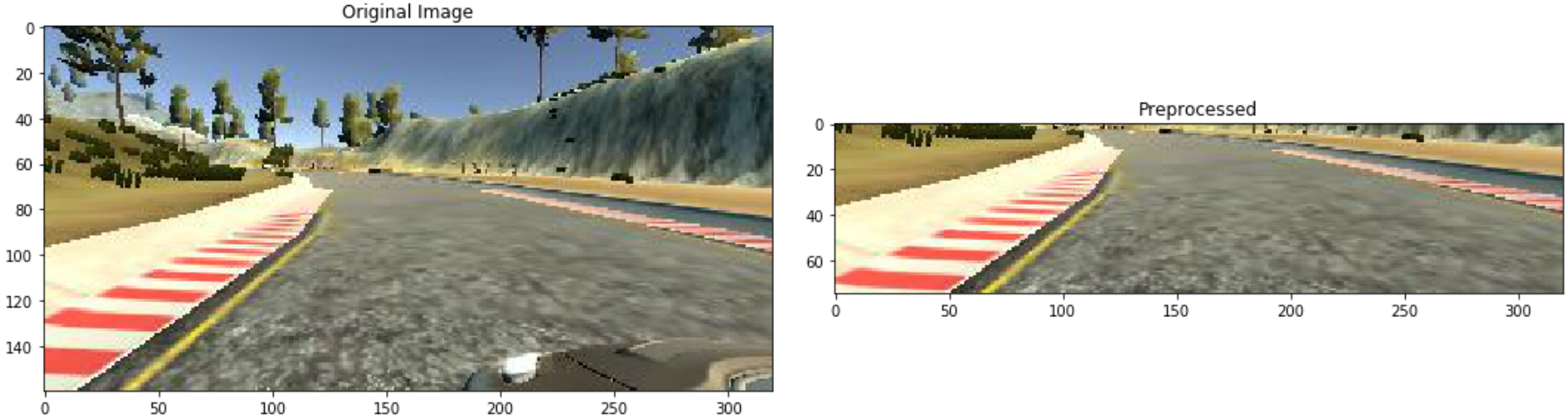


In [26]:
```
#Let us cut unnessary top and bottom, top side which shoes sky and clouds and the bonnet on the bottom
```

In [27]:
```
def img_preprocess(img) :
    img = mpimg.imread(img)
    img = img[60:135,:,:]
    return img
```

In [28]:
```
preprocessed_image = img_preprocess(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')
```

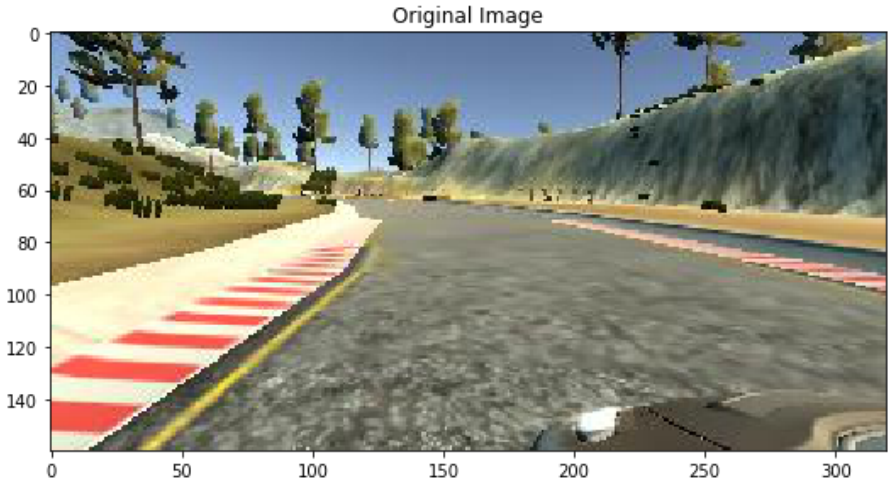
Out[28]: Text(0.5, 1.0, 'Preprocessed')

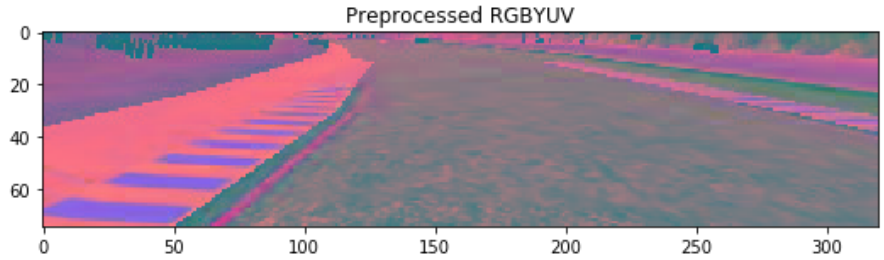


In [29]:
```
def img_preprocess(img) :
    img = mpimg.imread(img)
    img = img[60:135,:,:]
    img = cv2.cvtColor(img, cv2.COLOR_RGB2YUV)
    return img
```

In [30]:
```
preprocessed_image = img_preprocess(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 RGBYUV')
```

Out[30]: Text(0.5, 1.0, 'Preprocessed RGBYUV')

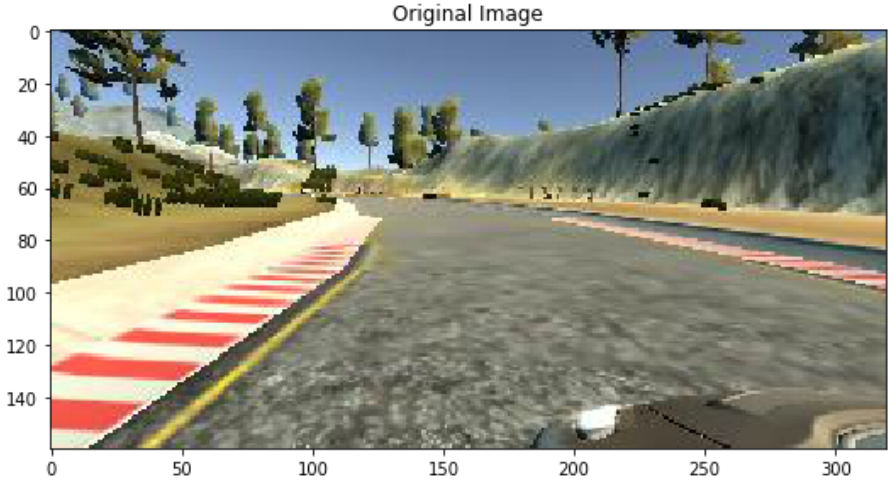


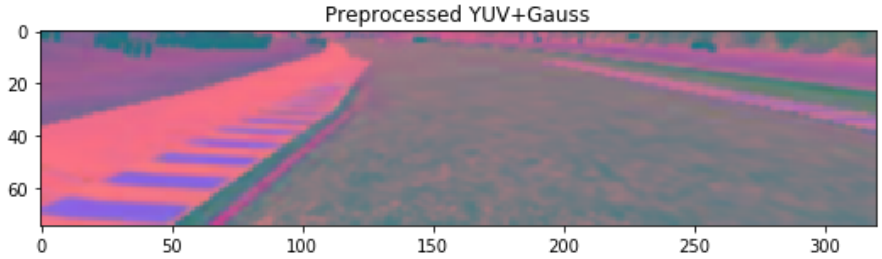


In [31]:
```
def img_preprocess(img) :
    img = mpimg.imread(img)
    img = img[60:135,:,:]
    img = cv2.cvtColor(img, cv2.COLOR_RGB2YUV)
    img = cv2.GaussianBlur(img, (3,3),0)
    return img
```

In [32]:
```
preprocessed_image = img_preprocess(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 YUV+Gauss')
```

Out[32]: Text(0.5, 1.0, 'Preprocessed YUV+Gauss')

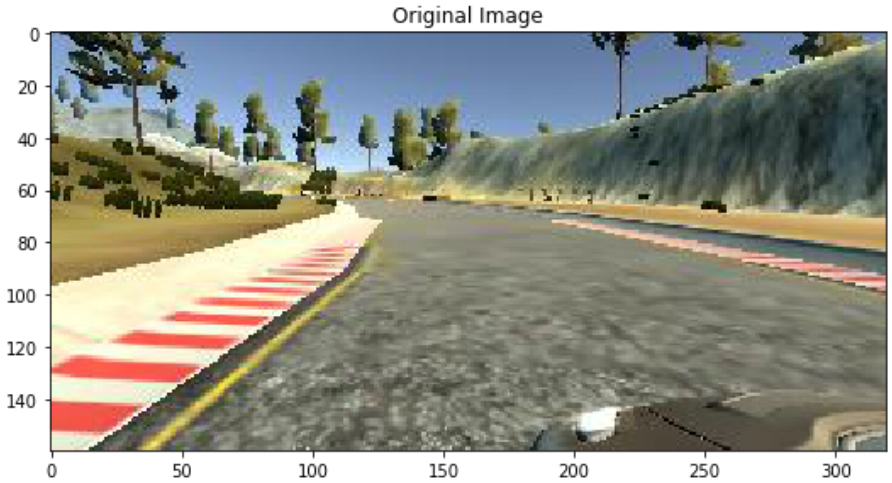


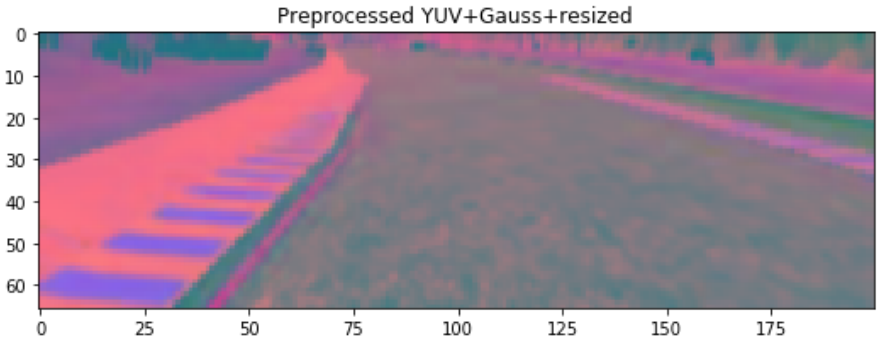


In [33]:
```
def img_preprocess(img) :
    img = mpimg.imread(img)
    img = img[60:135,:,:]
    img = cv2.cvtColor(img, cv2.COLOR_RGB2YUV)
    img = cv2.GaussianBlur(img, (3,3),0)
    img = cv2.resize(img,(200,66))
    return img
```

In [34]:
```
preprocessed_image = img_preprocess(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 YUV+Gauss+resized')
```

Out[34]: Text(0.5, 1.0, 'Preprocessed YUV+Gauss+resized')

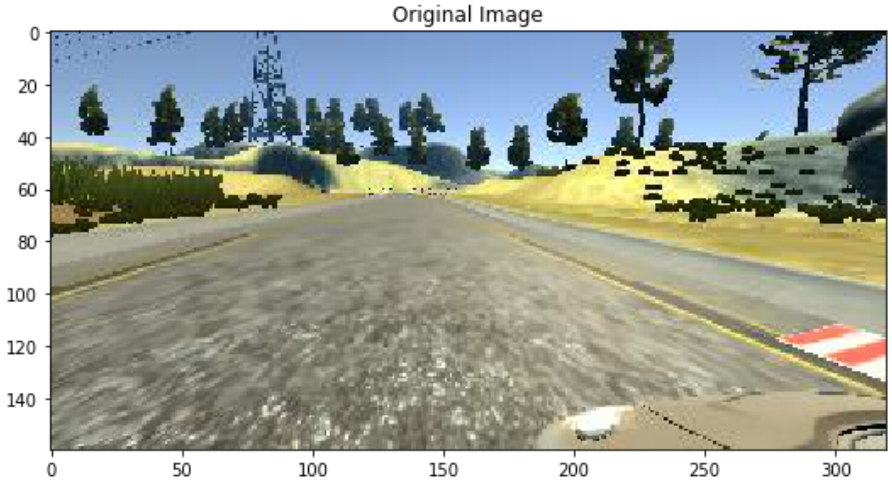


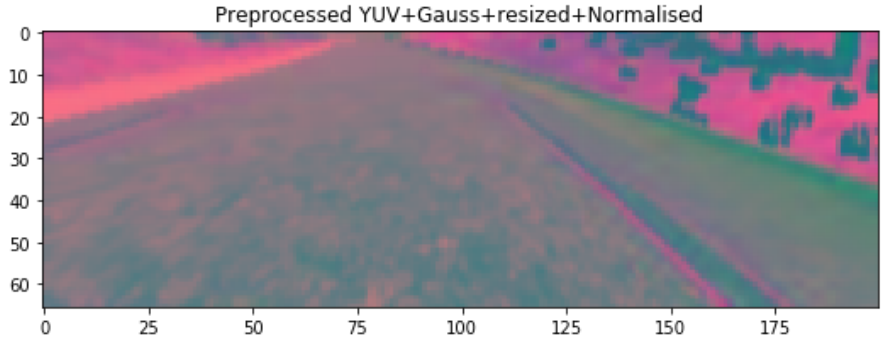


In [142]:
```
def img_preprocess_read(img) :
    img = mpimg.imread(img)
    img = img[60:135,:,:]
    img = cv2.cvtColor(img, cv2.COLOR_RGB2YUV)
    img = cv2.GaussianBlur(img, (3,3),0)
    img = cv2.resize(img,(200,66))
    img = img/255
    return img
```

In [143]:
```
preprocessed_image = img_preprocess_read(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 YUV+Gauss+resized+Normalised')
```

Out[143]: Text(0.5, 1.0, 'Preprocessed YUV+Gauss+resized+Normalised')





****Inserting after first test with ELU -Batch Generator***

to get small live batches instead of all at once

In [172]:
```
def img_preprocess(img) :
    #img = mpimg.imread(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
```

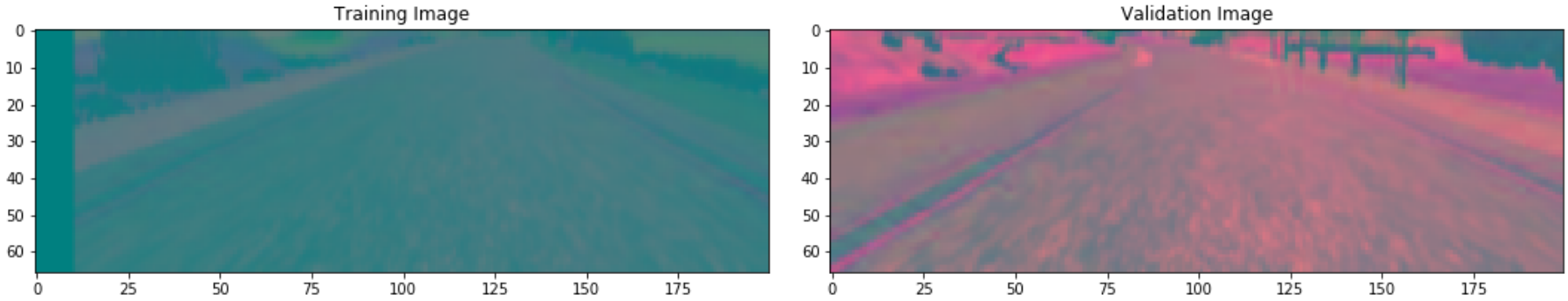
In [173]:
```
def batch_generator(image_paths, steering_ang, batch_size, istraining):

  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[random_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))
```

In [174]:
```
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))
```

Let us plot and check batch generator

In [175]:
```
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[175]: Text(0.5, 1.0, 'Validation Image')



*** Insert Ends**

In [37]:
```
X_train = np.array(list(map(img_preprocess, X_train)))
X_valid = np.array(list(map(img_preprocess, X_valid)))
```

In [38]:
```
plt.imshow(X_train[random.randint(0, len(X_train) -1)])
plt.axis('off')
print(X_train.shape)
```

```
(3031, 66, 200, 3)
```

Note its 66x200 this is the input size for of model

Let us read NVDIA Paper ref https://arxiv.org/pdf/1604.07316v1.pdf (https://arxiv.org/pdf/1604.07316v1.pdf)

Let us build NVIDIA MODEL

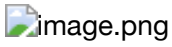
Thanks to NVIDIA for paper,figs & model are from two nvidia papers and references mentioned. Fully acknowledged, appreciated and thanked for sharing with public.

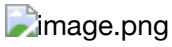
Title :End to End Learning for Self-Driving Cars

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

image.png

image.png





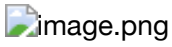
Thanks to NVIDIA Title :End to End Learning for Self-Driving Cars

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 [39]:
```
def nvidia_model():
    #Normalisation - we can skip as we already did it above in sections.
    #we remove subsampling in last two layers as skipping pixels in already well sliced and diced image featu
res in not necessary
    # Flatten layer - converts output to one dimensional layer so it can be fed to fully connected layers
    # dropouts to help generalise
    model = Sequential ()
    model.add(Convolution2D(24,5,5, subsample=(2,2), input_shape=(66,200,3), activation='relu'))
    model.add(Convolution2D(36,5,5, subsample =(2,2), activation='relu'))
    model.add(Convolution2D(48,5,5, subsample =(2,2), activation='relu'))
    model.add(Convolution2D(64,3,3,  activation='relu'))
    model.add(Convolution2D(64,3,3,  activation='relu'))
    model.add(Dropout(0.5))

    model.add (Flatten())
    model.add(Dense(100, activation = 'relu'))
    model.add(Dropout(0.5))

    model.add(Dense(50, activation = 'relu'))
    model.add(Dropout(0.5))
    model.add(Dense(10, activation = 'relu'))
    model.add(Dropout(0.5))
    model.add(Dense(1))

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

In [40]:
```
model = nvidia_model()
print(model.summary())
```

```
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_launcher.py:7: UserWarning: Update your `
Conv2D` call to the Keras 2 API: `Conv2D(24, (5, 5), input_shape=(66, 200, ..., activation="relu", strides=
(2, 2))`
  import sys
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_launcher.py:8: UserWarning: Update your `
Conv2D` call to the Keras 2 API: `Conv2D(36, (5, 5), activation="relu", strides=(2, 2))`

/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_launcher.py:9: UserWarning: Update your `
Conv2D` call to the Keras 2 API: `Conv2D(48, (5, 5), activation="relu", strides=(2, 2))`
  if __name__ == '__main__':
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_launcher.py:10: UserWarning: Update your
`Conv2D` call to the Keras 2 API: `Conv2D(64, (3, 3), activation="relu")`
  # Remove the CWD from sys.path while we load stuff.
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_launcher.py:11: UserWarning: Update your
`Conv2D` call to the Keras 2 API: `Conv2D(64, (3, 3), activation="relu")`
  # This is added back by InteractiveShellApp.init_path()
```

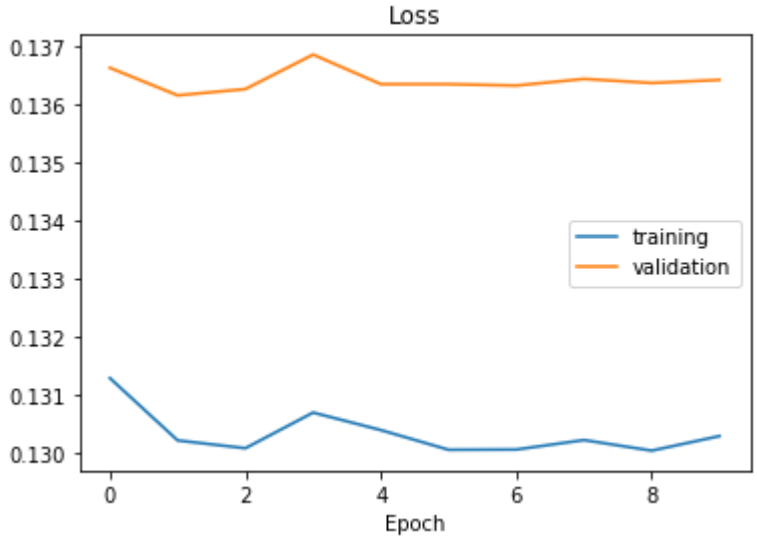
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
conv2d_1 (Conv2D)            (None, 31, 98, 24)        1824      
_________________________________________________________________
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     
_________________________________________________________________
dropout_1 (Dropout)          (None, 1, 18, 64)         0         
_________________________________________________________________
flatten_1 (Flatten)          (None, 1152)              0         
_________________________________________________________________
dense_1 (Dense)              (None, 100)               115300    
_________________________________________________________________
dropout_2 (Dropout)          (None, 100)               0         
_________________________________________________________________
dense_2 (Dense)              (None, 50)                5050      
_________________________________________________________________
dropout_3 (Dropout)          (None, 50)                0         
_________________________________________________________________
dense_3 (Dense)              (None, 10)                510       
_________________________________________________________________
dropout_4 (Dropout)          (None, 10)                0         
_________________________________________________________________
dense_4 (Dense)              (None, 1)                 11        
=================================================================
Total params: 252,219
Trainable params: 252,219
Non-trainable params: 0
_________________________________________________________________
None
```

In [41]:
```
history = model.fit(X_train,y_train, epochs=10, validation_data= (X_valid,y_valid), batch_size=100,verbose=1,
shuffle=1)
```

```
Train on 3031 samples, validate on 758 samples
Epoch 1/10
3031/3031 [==============================] - 24s 8ms/step - loss: 0.1313 - val_loss: 0.1366
Epoch 2/10
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1302 - val_loss: 0.1361
Epoch 3/10
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 4/10
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1307 - val_loss: 0.1368
Epoch 5/10
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1304 - val_loss: 0.1363
Epoch 6/10
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 7/10
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 8/10
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1302 - val_loss: 0.1364
Epoch 9/10
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1300 - val_loss: 0.1364
Epoch 10/10
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1303 - val_loss: 0.1364
```

In [42]:
```
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.legend(['training', 'validation'])
plt.title('Loss')
plt.xlabel('Epoch')
```

Out[42]: Text(0.5, 0, 'Epoch')



In [43]:
```
history = model.fit(X_train,y_train, epochs=50, validation_data= (X_valid,y_valid), batch_size=100,verbose=1,
shuffle=1)
```

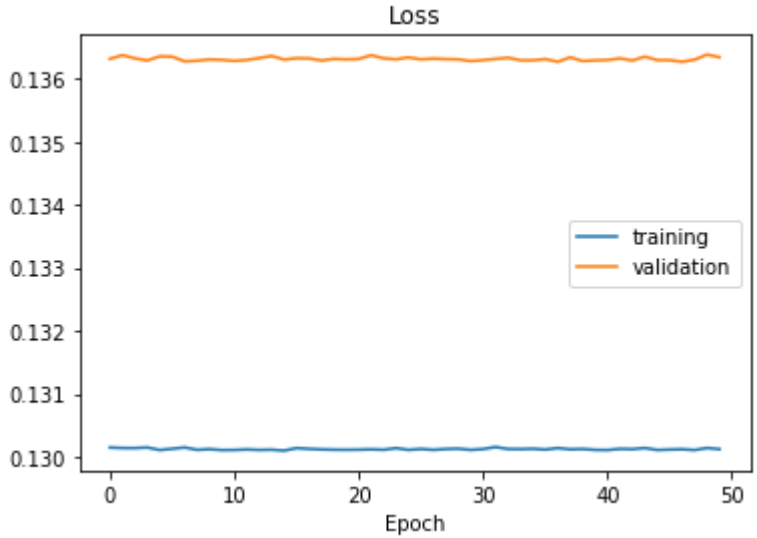
```
Train on 3031 samples, validate on 758 samples
Epoch 1/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 2/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1364
Epoch 3/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 4/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 5/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1364
Epoch 6/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 7/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 8/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 9/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 10/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 11/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 12/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 13/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 14/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1364
Epoch 15/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 16/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 17/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 18/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 19/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 20/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 21/50
```

```
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 22/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1364
Epoch 23/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 24/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 25/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 26/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 27/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 28/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 29/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 30/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 31/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 32/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1302 - val_loss: 0.1363
Epoch 33/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 34/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 35/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 36/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 37/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 38/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 39/50
3031/3031 [==============================] - 22s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 40/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 41/50
3031/3031 [==============================] - 24s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 42/50
```

```
3031/3031 [==============================] - 24s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 43/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 44/50
3031/3031 [==============================] - 23s 7ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 45/50
3031/3031 [==============================] - 28s 9ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 46/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 47/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 48/50
3031/3031 [==============================] - 24s 8ms/step - loss: 0.1301 - val_loss: 0.1363
Epoch 49/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1364
Epoch 50/50
3031/3031 [==============================] - 23s 8ms/step - loss: 0.1301 - val_loss: 0.1363
```

In [44]:
```
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.legend(['training', 'validation'])
plt.title('Loss')
plt.xlabel('Epoch')
```

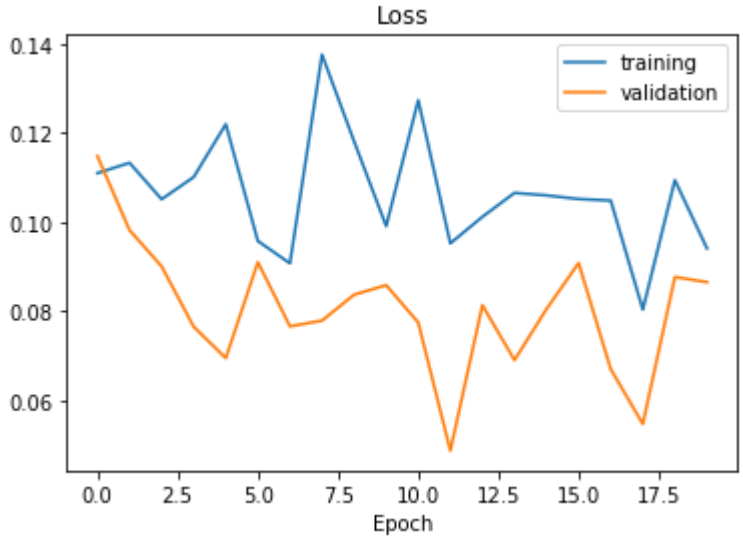
Out[44]: Text(0.5, 0, 'Epoch')



After Elu - Insertion for new augumentation

In [176]:
```
history_elu_augmented = model_elu.fit_generator(batch_generator(X_train, y_train, 100, 1),
                                    steps_per_epoch=3,
                                    epochs=20,
                                    validation_data=batch_generator(X_valid, y_valid, 100, 0),
                                    validation_steps=2,
                                    verbose=1,
                                    shuffle = 1)
plt.plot(history_elu_augmented.history['loss'])
plt.plot(history_elu_augmented.history['val_loss'])
plt.legend(['training', 'validation'])
plt.title('Loss')
plt.xlabel('Epoch')
model.save('model_Final.h5')
```

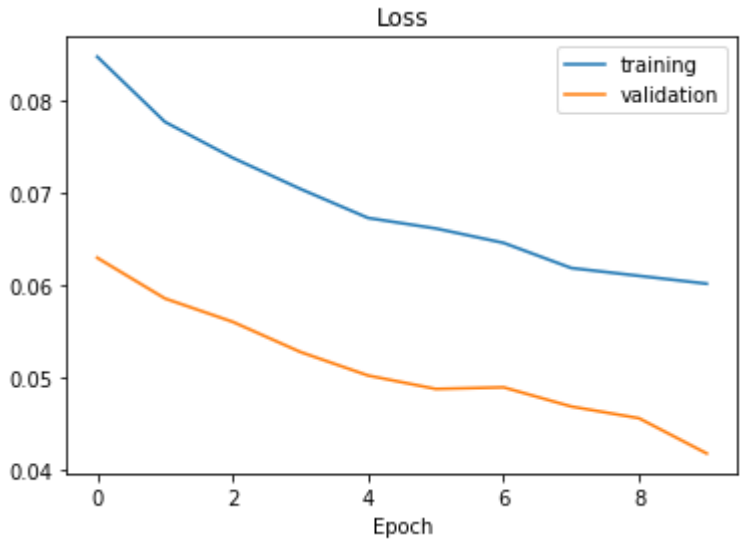
```
Epoch 1/20
3/3 [==============================] - 5s 2s/step - loss: 0.1110 - val_loss: 0.1148
Epoch 2/20
3/3 [==============================] - 5s 2s/step - loss: 0.1133 - val_loss: 0.0982
Epoch 3/20
3/3 [==============================] - 4s 1s/step - loss: 0.1051 - val_loss: 0.0901
Epoch 4/20
3/3 [==============================] - 4s 1s/step - loss: 0.1101 - val_loss: 0.0765
Epoch 5/20
3/3 [==============================] - 4s 1s/step - loss: 0.1219 - val_loss: 0.0695
Epoch 6/20
3/3 [==============================] - 4s 1s/step - loss: 0.0957 - val_loss: 0.0910
Epoch 7/20
3/3 [==============================] - 4s 1s/step - loss: 0.0907 - val_loss: 0.0766
Epoch 8/20
3/3 [==============================] - 4s 1s/step - loss: 0.1375 - val_loss: 0.0779
Epoch 9/20
3/3 [==============================] - 4s 1s/step - loss: 0.1181 - val_loss: 0.0837
Epoch 10/20
3/3 [==============================] - 4s 1s/step - loss: 0.0991 - val_loss: 0.0858
Epoch 11/20
3/3 [==============================] - 4s 1s/step - loss: 0.1273 - val_loss: 0.0775
Epoch 12/20
3/3 [==============================] - 4s 1s/step - loss: 0.0952 - val_loss: 0.0487
Epoch 13/20
3/3 [==============================] - 4s 1s/step - loss: 0.1012 - val_loss: 0.0813
Epoch 14/20
3/3 [==============================] - 4s 1s/step - loss: 0.1065 - val_loss: 0.0690
Epoch 15/20
3/3 [==============================] - 4s 1s/step - loss: 0.1060 - val_loss: 0.0805
Epoch 16/20
3/3 [==============================] - 4s 1s/step - loss: 0.1052 - val_loss: 0.0908
Epoch 17/20
3/3 [==============================] - 4s 1s/step - loss: 0.1048 - val_loss: 0.0670
Epoch 18/20
3/3 [==============================] - 4s 1s/step - loss: 0.0804 - val_loss: 0.0547
Epoch 19/20
3/3 [==============================] - 4s 1s/step - loss: 0.1094 - val_loss: 0.0877
Epoch 20/20
3/3 [==============================] - 4s 1s/step - loss: 0.0941 - val_loss: 0.0865
```



Looks not too bad, so to continue, after learning rate changed to e-4 from e-3

In [177]:
```
history_elu_augmented = model_elu.fit_generator(batch_generator(X_train, y_train, 100, 1),
                                    steps_per_epoch=300,
                                    epochs=10,
                                    validation_data=batch_generator(X_valid, y_valid, 100, 0),
                                    validation_steps=200,
                                    verbose=1,
                                    shuffle = 1)
plt.plot(history_elu_augmented.history['loss'])
plt.plot(history_elu_augmented.history['val_loss'])
plt.legend(['training', 'validation'])
plt.title('Loss')
plt.xlabel('Epoch')
model.save('model_Final.h5')
```

```
Epoch 1/10
300/300 [==============================] - 331s 1s/step - loss: 0.0847 - val_loss: 0.0629
Epoch 2/10
300/300 [==============================] - 333s 1s/step - loss: 0.0776 - val_loss: 0.0585
Epoch 3/10
300/300 [==============================] - 322s 1s/step - loss: 0.0737 - val_loss: 0.0560
Epoch 4/10
300/300 [==============================] - 321s 1s/step - loss: 0.0704 - val_loss: 0.0527
Epoch 5/10
300/300 [==============================] - 323s 1s/step - loss: 0.0672 - val_loss: 0.0502
Epoch 6/10
300/300 [==============================] - 322s 1s/step - loss: 0.0661 - val_loss: 0.0487
Epoch 7/10
300/300 [==============================] - 321s 1s/step - loss: 0.0645 - val_loss: 0.0489
Epoch 8/10
300/300 [==============================] - 323s 1s/step - loss: 0.0618 - val_loss: 0.0468
Epoch 9/10
300/300 [==============================] - 323s 1s/step - loss: 0.0610 - val_loss: 0.0455
Epoch 10/10
300/300 [==============================] - 323s 1s/step - loss: 0.0601 - val_loss: 0.0417
```



In [ ]:
```
model.save('model_augmentedelu.h5')
```

MODEL DEFINITION SECTION

In [152]:
```
def nvidia_model_elu():
    #Normalisation - we can skip as we already did it above in sections.
    #we remove subsampling in last two layers as skipping pixels in already well sliced and diced image featu
res in not necessary
    # Flatten layer - converts output to one dimensional layer so it can be fed to fully connected layers
    # dropouts to help generalise
    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)
    optimizer = Adam(lr=1e-4)
    model.compile(loss='mse',optimizer = optimizer)
    return model
```

In [153]:
```
model_elu = nvidia_model_elu()
print(model_elu.summary())
```

```
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_launcher.py:7: UserWarning: Update your `
Conv2D` call to the Keras 2 API: `Conv2D(24, (5, 5), input_shape=(66, 200, ..., activation="elu", strides=
(2, 2))`
  import sys
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_launcher.py:8: UserWarning: Update your `
Conv2D` call to the Keras 2 API: `Conv2D(36, (5, 5), activation="elu", strides=(2, 2))`

/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_launcher.py:9: UserWarning: Update your `
Conv2D` call to the Keras 2 API: `Conv2D(48, (5, 5), activation="elu", strides=(2, 2))`
  if __name__ == '__main__':
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_launcher.py:10: UserWarning: Update your
`Conv2D` call to the Keras 2 API: `Conv2D(64, (3, 3), activation="elu")`
  # Remove the CWD from sys.path while we load stuff.
/miniconda3/envs/udrivesimul/lib/python3.6/site-packages/ipykernel_launcher.py:11: UserWarning: Update your
`Conv2D` call to the Keras 2 API: `Conv2D(64, (3, 3), activation="elu")`
  # This is added back by InteractiveShellApp.init_path()
```

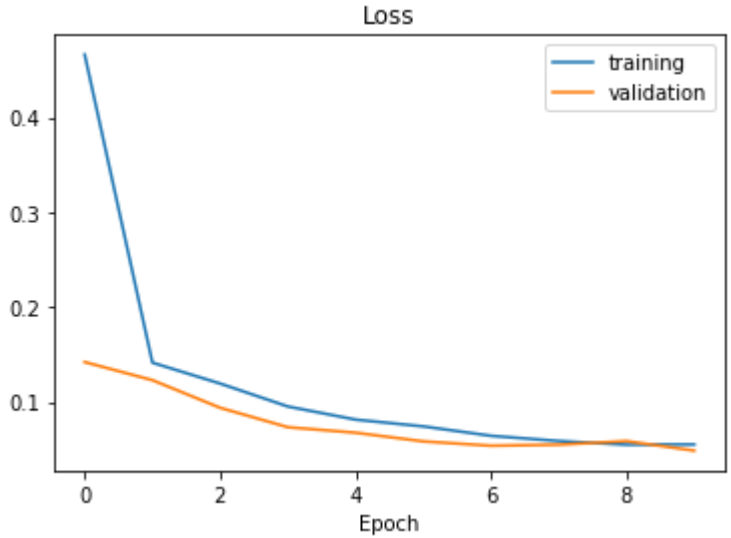
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
conv2d_11 (Conv2D)           (None, 31, 98, 24)        1824      
_________________________________________________________________
conv2d_12 (Conv2D)           (None, 14, 47, 36)        21636     
_________________________________________________________________
conv2d_13 (Conv2D)           (None, 5, 22, 48)         43248     
_________________________________________________________________
conv2d_14 (Conv2D)           (None, 3, 20, 64)         27712     
_________________________________________________________________
conv2d_15 (Conv2D)           (None, 1, 18, 64)         36928     
_________________________________________________________________
dropout_7 (Dropout)          (None, 1, 18, 64)         0         
_________________________________________________________________
flatten_3 (Flatten)          (None, 1152)              0         
_________________________________________________________________
dense_9 (Dense)              (None, 100)               115300    
_________________________________________________________________
dropout_8 (Dropout)          (None, 100)               0         
_________________________________________________________________
dense_10 (Dense)             (None, 50)                5050      
_________________________________________________________________
dense_11 (Dense)             (None, 10)                510       
_________________________________________________________________
dense_12 (Dense)             (None, 1)                 11        
=================================================================
Total params: 252,219
Trainable params: 252,219
Non-trainable params: 0
_________________________________________________________________
None
```

In [47]:
```
history_elu = model_elu.fit(X_train,y_train, epochs=10, validation_data= (X_valid,y_valid), batch_size=100,ve
rbose=1,shuffle=1)
```

```
Train on 3031 samples, validate on 758 samples
Epoch 1/10
3031/3031 [==============================] - 25s 8ms/step - loss: 0.4661 - val_loss: 0.1419
Epoch 2/10
3031/3031 [==============================] - 24s 8ms/step - loss: 0.1413 - val_loss: 0.1229
Epoch 3/10
3031/3031 [==============================] - 24s 8ms/step - loss: 0.1193 - val_loss: 0.0937
Epoch 4/10
3031/3031 [==============================] - 29s 10ms/step - loss: 0.0951 - val_loss: 0.0733
Epoch 5/10
3031/3031 [==============================] - 27s 9ms/step - loss: 0.0813 - val_loss: 0.0675
Epoch 6/10
3031/3031 [==============================] - 26s 9ms/step - loss: 0.0741 - val_loss: 0.0583
Epoch 7/10
3031/3031 [==============================] - 27s 9ms/step - loss: 0.0642 - val_loss: 0.0538
Epoch 8/10
3031/3031 [==============================] - 24s 8ms/step - loss: 0.0587 - val_loss: 0.0549
Epoch 9/10
3031/3031 [==============================] - 24s 8ms/step - loss: 0.0549 - val_loss: 0.0586
Epoch 10/10
3031/3031 [==============================] - 24s 8ms/step - loss: 0.0551 - val_loss: 0.0486
```

In [48]:
```
plt.plot(history_elu.history['loss'])
plt.plot(history_elu.history['val_loss'])
plt.legend(['training', 'validation'])
plt.title('Loss')
plt.xlabel('Epoch')
```

Out[48]: Text(0.5, 0, 'Epoch')



So We have eliminated dead ReLu issues ie that negative gradient behind negative region in activation function

image.png

In [50]:
```
model.save('model_elu.h5')
```

In [51]:
```
!pwd
```

```
/Other_Apps/HomeLab/Udacity/SelfDrivingCars/Behavioral_cloning/project_area
```

-Ref to Simulator Videos and screenshots

Now Our Model drives first turn and leaves ! so it learnt better than first time but not enough

Let us find ways to improve our model by augmentation. Let us use imgaug library, it seems better than keras and more refreshing to try something new

ref: imgaug.readthedocs.io

In [ ]:
```
!conda install
```