1: IMPORTING LIBRARIES AND DATASET

In [37]:

```
import os
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import zipfile
import cv2
from skimage import io

import tensorflow as tf
from tensorflow.python.keras import Sequential
from tensorflow.keras import layers, optimizers
from tensorflow.keras.layers import *
from tensorflow.keras.models import Model
from tensorflow.keras.initializers import glorot_uniform
from tensorflow.keras.utils import plot_model
from tensorflow.keras.callbacks import ReduceLROnPlateau, EarlyStopping, ModelCheckpoint,
LearningRateScheduler
import tensorflow.keras.backend as K

import random
import glob
from sklearn.preprocessing import StandardScaler, normalize
from IPython.display import display
```

In [38]:

```
from zipfile import ZipFile

filename='archive (5).zip'
with ZipFile(filename, 'r') as zipObj:
   # Extract all the contents of zip file in current directory
   zipObj.extractall()
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
<ipython-input-38-f5a7f832c88d> in <module>
      2
      3 filename='archive (5).zip'
----> 4 with ZipFile(filename, 'r') as zipObj:
      5    # Extract all the contents of zip file in current directory
      6    zipObj.extractall()

~/anaconda3/envs/tensorflow_p36/lib/python3.6/zipfile.py in __init__(self, fi
le, mode, compression, allowZip64)
   1111             while True:
   1112                 try:
-> 1113                     self.fp = io.open(file, filemode)
   1114                 except OSError:
   1115                     if filemode in modeDict:

FileNotFoundError: [Errno 2] No such file or directory: 'archive (5).zip'
```

In [39]:

```
data = pd.read_csv('./lgg-mri-segmentation/kaggle_3m/data.csv')
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 110 entries, 0 to 109
Data columns (total 18 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Patient                    110 non-null    object
 1   RNASeqCluster              92 non-null     float64
 2   MethylationCluster         109 non-null    float64
 3   miRNACluster               110 non-null    int64
 4   CNCluster                  108 non-null    float64
 5   RPPACluster                98 non-null     float64
 6   OncosignCluster            105 non-null    float64
 7   COCCluster                 110 non-null    int64
 8   histological_type          109 non-null    float64
 9   neoplasm_histologic_grade  109 non-null    float64
 10  tumor_tissue_site          109 non-null    float64
 11  laterality                 109 non-null    float64
 12  tumor_location             109 non-null    float64
 13  gender                     109 non-null    float64
 14  age_at_initial_pathologic  109 non-null    float64
 15  race                       108 non-null    float64
 16  ethnicity                  102 non-null    float64
 17  death01                    109 non-null    float64
dtypes: float64(15), int64(2), object(1)
memory usage: 15.6+ KB
```

In [40]:

```
data.head(10)
```

Out[40]:

	Patient	RNASeqCluster	MethylationCluster	miRNACluster	CNCluster	RPPACluster
0	TCGA_CS_4941	2.0	4.0	2	2.0	NaN
1	TCGA_CS_4942	1.0	5.0	2	1.0	1.0
2	TCGA_CS_4943	1.0	5.0	2	1.0	2.0
3	TCGA_CS_4944	NaN	5.0	2	1.0	2.0
4	TCGA_CS_5393	4.0	5.0	2	1.0	2.0
5	TCGA_CS_5395	2.0	4.0	2	2.0	NaN
6	TCGA_CS_5396	3.0	3.0	2	3.0	2.0
7	TCGA_CS_5397	NaN	4.0	1	2.0	3.0
8	TCGA_CS_6186	2.0	4.0	1	2.0	1.0
9	TCGA_CS_6188	2.0	4.0	3	2.0	3.0

In [41]:

```
data_map = []
for sub_dir_path in glob.glob("./lgg-mri-segmentation/kaggle_3m/"+"*"):
    #if os.path.isdir(sub_path_dir):
    try:
        dir_name = sub_dir_path.split('/')[-1]
        for filename in os.listdir(sub_dir_path):
            image_path = sub_dir_path + '/' + filename
            data_map.extend([dir_name, image_path])
    except Exception as e:
        print(e)
```

```
[Errno 20] Not a directory: './lgg-mri-segmentation/kaggle_3m/README.md'
[Errno 20] Not a directory: './lgg-mri-segmentation/kaggle_3m/data.csv'
```

In [42]:

```
df = pd.DataFrame({"patient_id" : data_map[::2],
                   "path" : data_map[1::2]})
df.head()
# print(df.path)
```

Out[42]:

	patient_id	path
0	TCGA_HT_7473_19970826	./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473_...
1	TCGA_HT_7473_19970826	./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473_...
2	TCGA_HT_7473_19970826	./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473_...
3	TCGA_HT_7473_19970826	./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473_...
4	TCGA_HT_7473_19970826	./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473_...

In [43]:

```
df_imgs = df[~df['path'].str.contains("mask")]
df_masks = df[df['path'].str.contains("mask")]

# File path line length images for later sorting
BASE_LEN = 89 # len(/kaggle/input/lgg-mri-segmentation/kaggle_3m/TCGA_DU_6404_19850629/TCG
A_DU_6404_19850629_ <-!!!43.tif)
END_IMG_LEN = 4 # len(/kaggle/input/lgg-mri-segmentation/kaggle_3m/TCGA_DU_6404_19850629/T
CGA_DU_6404_19850629_43 !!!->.tif)
END_MASK_LEN = 9 # (/kaggle/input/lgg-mri-segmentation/kaggle_3m/TCGA_DU_6404_19850629/TCG
A_DU_6404_19850629_43 !!!->_mask.tif)

# Data sorting
imgs = sorted(df_imgs["path"].values, key=lambda x : x[BASE_LEN:-END_IMG_LEN])
masks = sorted(df_masks["path"].values, key=lambda x : x[BASE_LEN:-END_MASK_LEN])

# Sorting check
idx = random.randint(0, len(imgs)-1)
print("Path to the Image:", imgs[idx], "\nPath to the Mask:", masks[idx])
```

```
Path to the Image: ./lgg-mri-segmentation/kaggle_3m/TCGA_DU_7019_19940908/TCG
A_DU_7019_19940908_20.tif
Path to the Mask: ./lgg-mri-segmentation/kaggle_3m/TCGA_DU_7019_19940908/TCGA
_DU_7019_19940908_15_mask.tif
```

In [44]:

```
# print(len(patient_id))
print(len(imgs))
print(len(masks))
```

```
3929
3929
```

In [45]:

```
print(df_masks)
```

```
                patient_id                                                pat
h
3     TCGA_HT_7473_19970826  ./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473
_...
4     TCGA_HT_7473_19970826  ./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473
_...
5     TCGA_HT_7473_19970826  ./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473
_...
6     TCGA_HT_7473_19970826  ./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473
_...
9     TCGA_HT_7473_19970826  ./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473
_...
...                     ...
...
7848  TCGA_DU_A5TR_19970726  ./lgg-mri-segmentation/kaggle_3m/TCGA_DU_A5TR
_...
7851  TCGA_DU_A5TR_19970726  ./lgg-mri-segmentation/kaggle_3m/TCGA_DU_A5TR
_...
7852  TCGA_DU_A5TR_19970726  ./lgg-mri-segmentation/kaggle_3m/TCGA_DU_A5TR
_...
7855  TCGA_DU_A5TR_19970726  ./lgg-mri-segmentation/kaggle_3m/TCGA_DU_A5TR
_...
7856  TCGA_DU_A5TR_19970726  ./lgg-mri-segmentation/kaggle_3m/TCGA_DU_A5TR
_...

[3929 rows x 2 columns]
```

creating final datset

In [47]:

```
# Final dataframe
a = {"patient_id": df_imgs.patient_id.values,"image_path": imgs,"mask_path": masks}
brain_df = pd.DataFrame(a)
def pos_neg_diagnosis(mask_path):
    value = np.max(cv2.imread(mask_path))
    if value > 0 :
        return 1
    else:
        return 0

brain_df['mask'] = brain_df['mask_path'].apply(lambda x: pos_neg_diagnosis(x))
brain_df
```

Out[47]:

	patient_id	image_path	
0	TCGA_HT_7473_19970826	./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473_...	segmentation/kaggle_3r
1	TCGA_HT_7473_19970826	./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473_...	segmentation/kaggle_3r
2	TCGA_HT_7473_19970826	./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473_...	segmentation/kaggle_3r
3	TCGA_HT_7473_19970826	./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473_...	segmentation/kaggle_3r
4	TCGA_HT_7473_19970826	./lgg-mri-segmentation/kaggle_3m/TCGA_HT_7473_...	segmentation/kaggle_3r
...	...	...	
3924	TCGA_DU_A5TR_19970726	./lgg-mri-segmentation/kaggle_3m/TCGA_DU_A5TR_...	segmentation/kaggle_3m
3925	TCGA_DU_A5TR_19970726	./lgg-mri-segmentation/kaggle_3m/TCGA_DU_A5TR_...	segmentation/kaggle_3m
3926	TCGA_DU_A5TR_19970726	./lgg-mri-segmentation/kaggle_3m/TCGA_DU_A5TR_...	segmentation/kaggle_3m
3927	TCGA_DU_A5TR_19970726	./lgg-mri-segmentation/kaggle_3m/TCGA_DU_A5TR_...	segmentation/kaggle_3m
3928	TCGA_DU_A5TR_19970726	./lgg-mri-segmentation/kaggle_3m/TCGA_DU_A5TR_...	segmentation/kaggle_3m

3929 rows × 4 columns

2: DATA VISUALIZATION

In [48]:

```
brain_df['mask'].value_counts()
```

Out[48]:

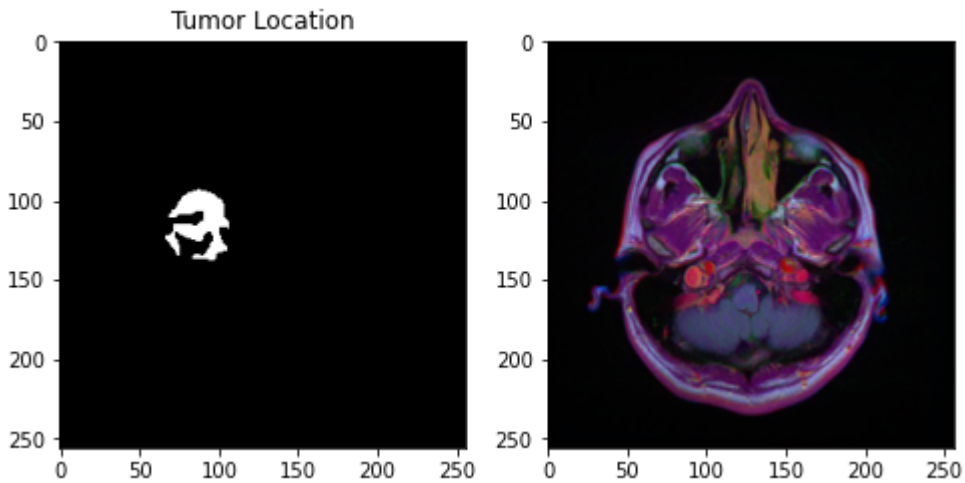
```
0    2556
1    1373
Name: mask, dtype: int64
```

In [11]:

```
import plotly.graph_objects as go  # using plotly to create interactive plots

fig = go.Figure([go.Bar(x=brain_df['mask'].value_counts().index,
                        y=brain_df['mask'].value_counts(),
                        width=[.4, .4]
                       )
                ])
fig.update_traces(marker_color='rgb(158,202,225)', marker_line_color='rgb(8,48,107)',
                  marker_line_width=4, opacity=0.4
                 )
fig.update_layout(title_text="Mask Count Plot",
                  width=700,
                  height=550,
                  yaxis=dict(
                             title_text="Count",
                             tickmode="array",
                             titlefont=dict(size=20)
                            )
                 )
fig.update_yaxes(automargin=True)
fig.show()
```

In [12]:

```
for i in range(len(brain_df)):
    if cv2.imread(brain_df.mask_path[i]).max() > 0:
        break

plt.figure(figsize=(8,8))
plt.subplot(1,2,1)
plt.imshow(cv2.imread(brain_df.mask_path[i]));
plt.title('Tumor Location')

plt.subplot(1,2,2)
plt.imshow(cv2.imread(brain_df.image_path[i]));
```



In [13]:

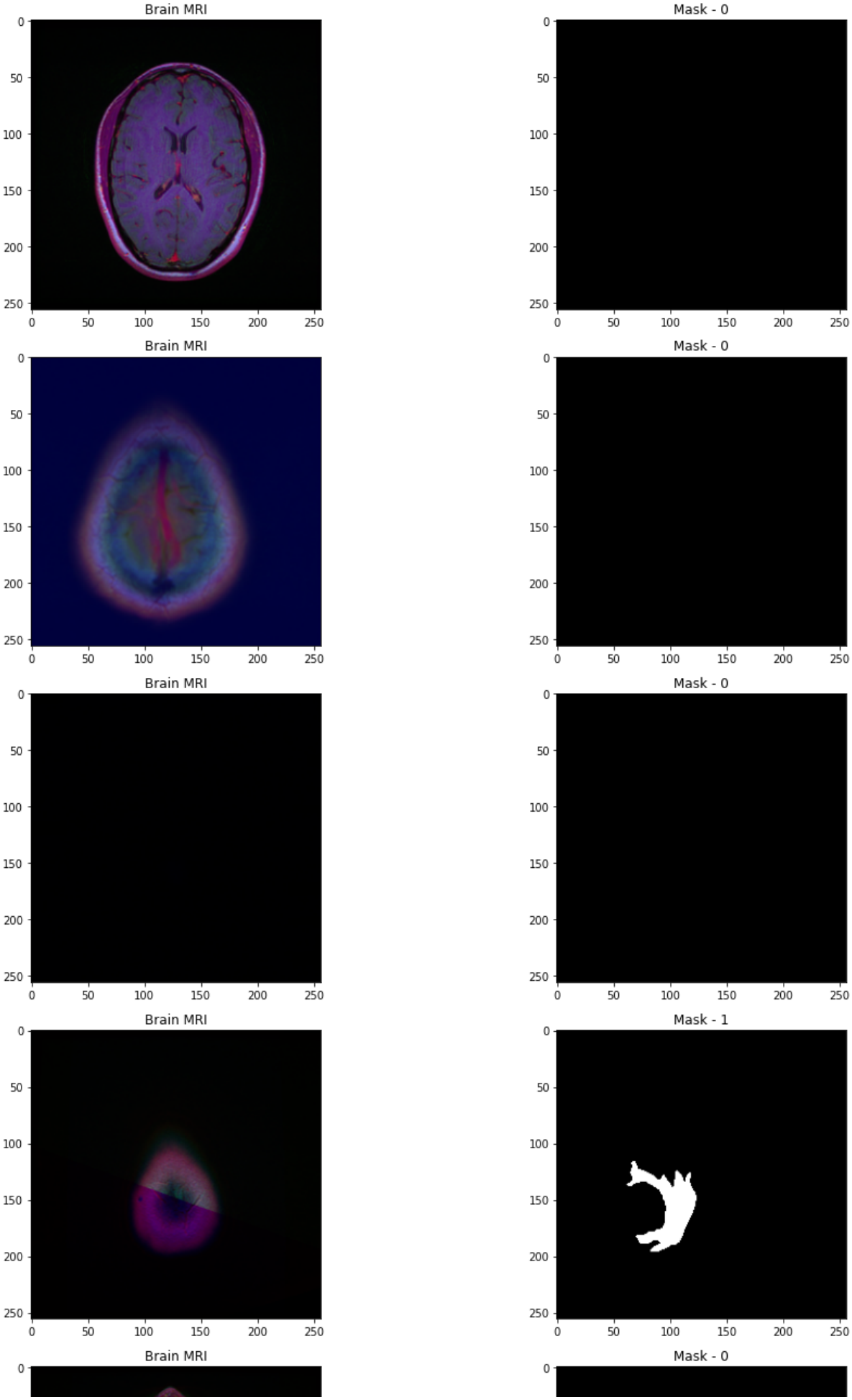
```
cv2.imread(brain_df.mask_path[i]).max(), cv2.imread(brain_df.mask_path[i]).min()
```

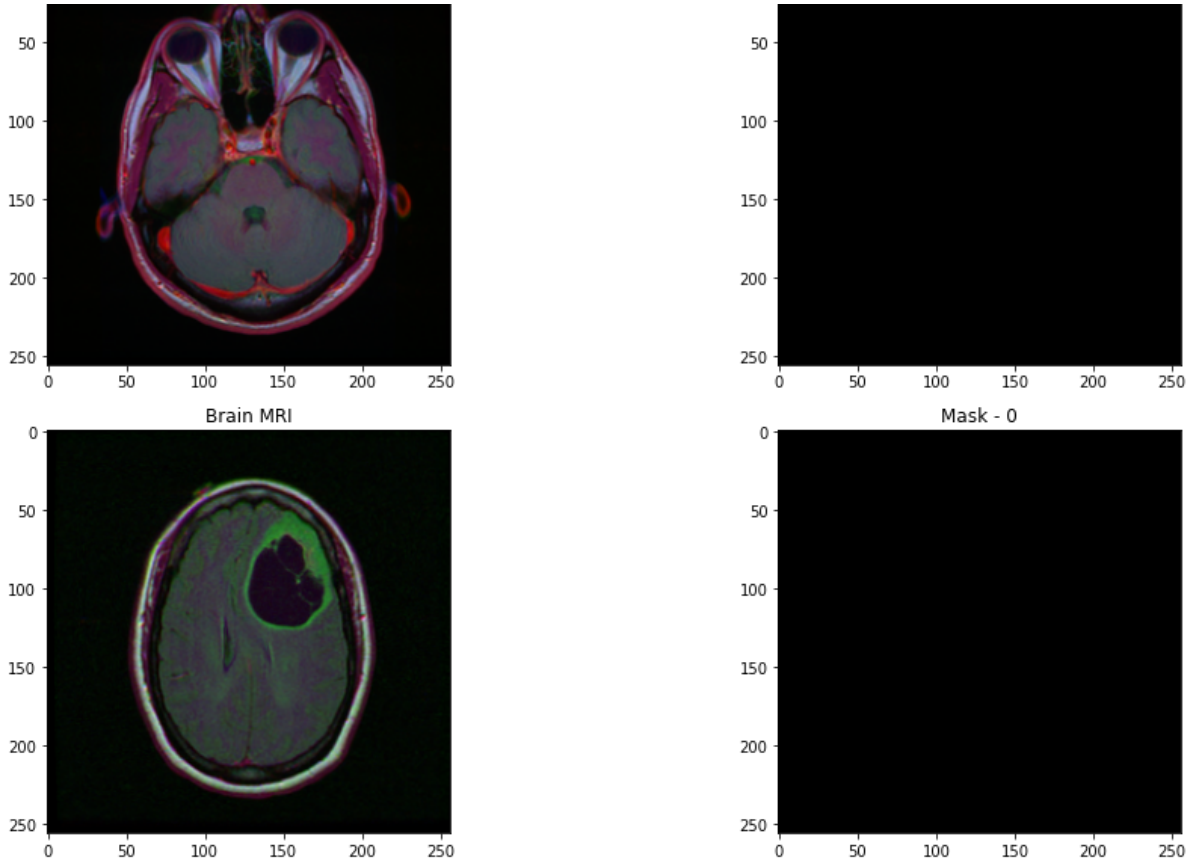
Out[13]:

```
(255, 0)
```

In [14]:

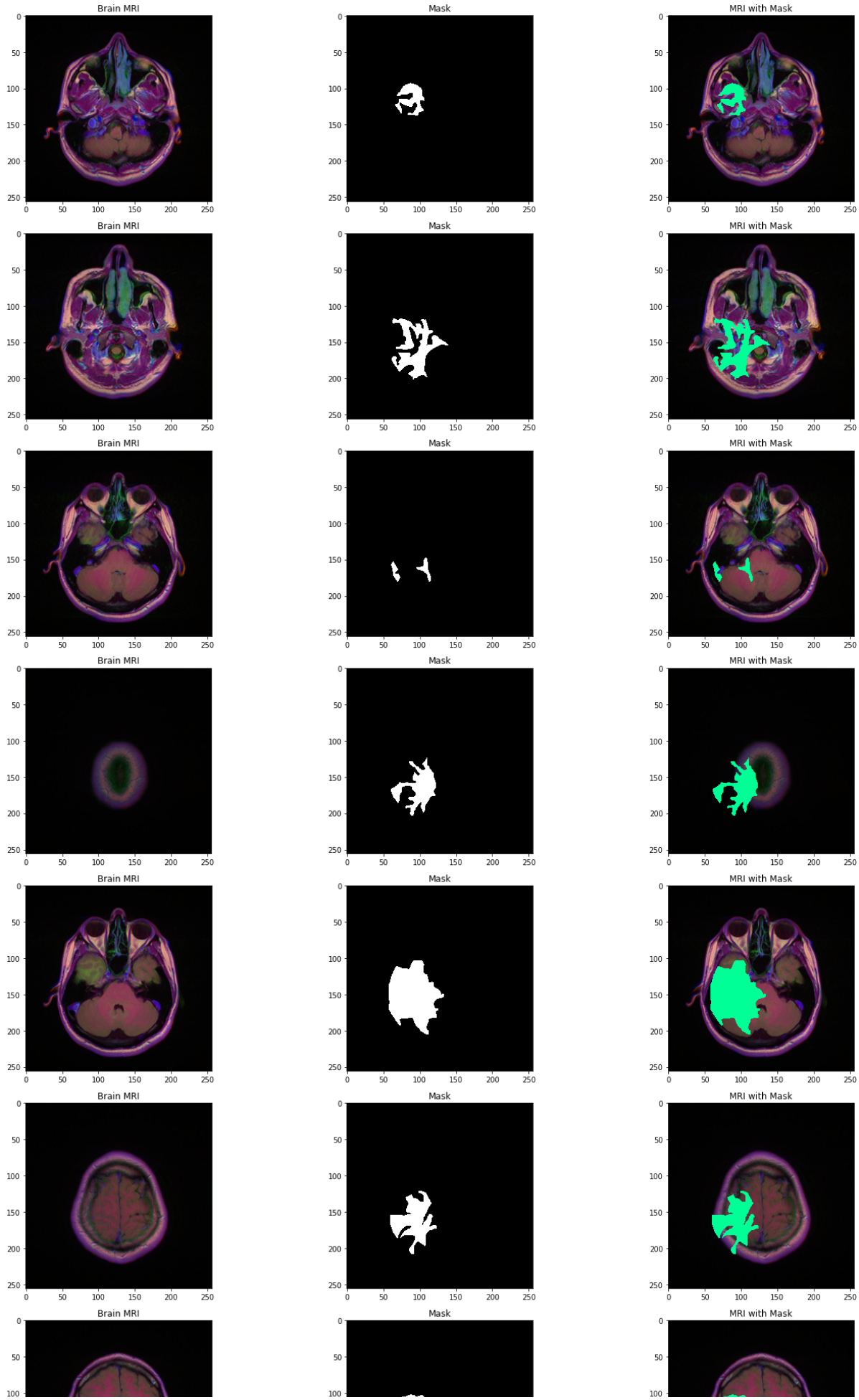
```
# Basic visualizations: Visualize the images (MRI and Mask) in the dataset separately

fig, axs = plt.subplots(6,2, figsize=(16,26))
count = 0
for x in range(6):
  i = random.randint(0, len(brain_df)) # select a random index
  axs[count][0].title.set_text("Brain MRI") # set title
  axs[count][0].imshow(cv2.imread(brain_df.image_path[i])) # show MRI
  axs[count][1].title.set_text("Mask - " + str(brain_df['mask'][i])) # plot title on the m
ask (0 or 1)
  axs[count][1].imshow(cv2.imread(brain_df.mask_path[i])) # Show corresponding mask
  count += 1

fig.tight_layout()
```

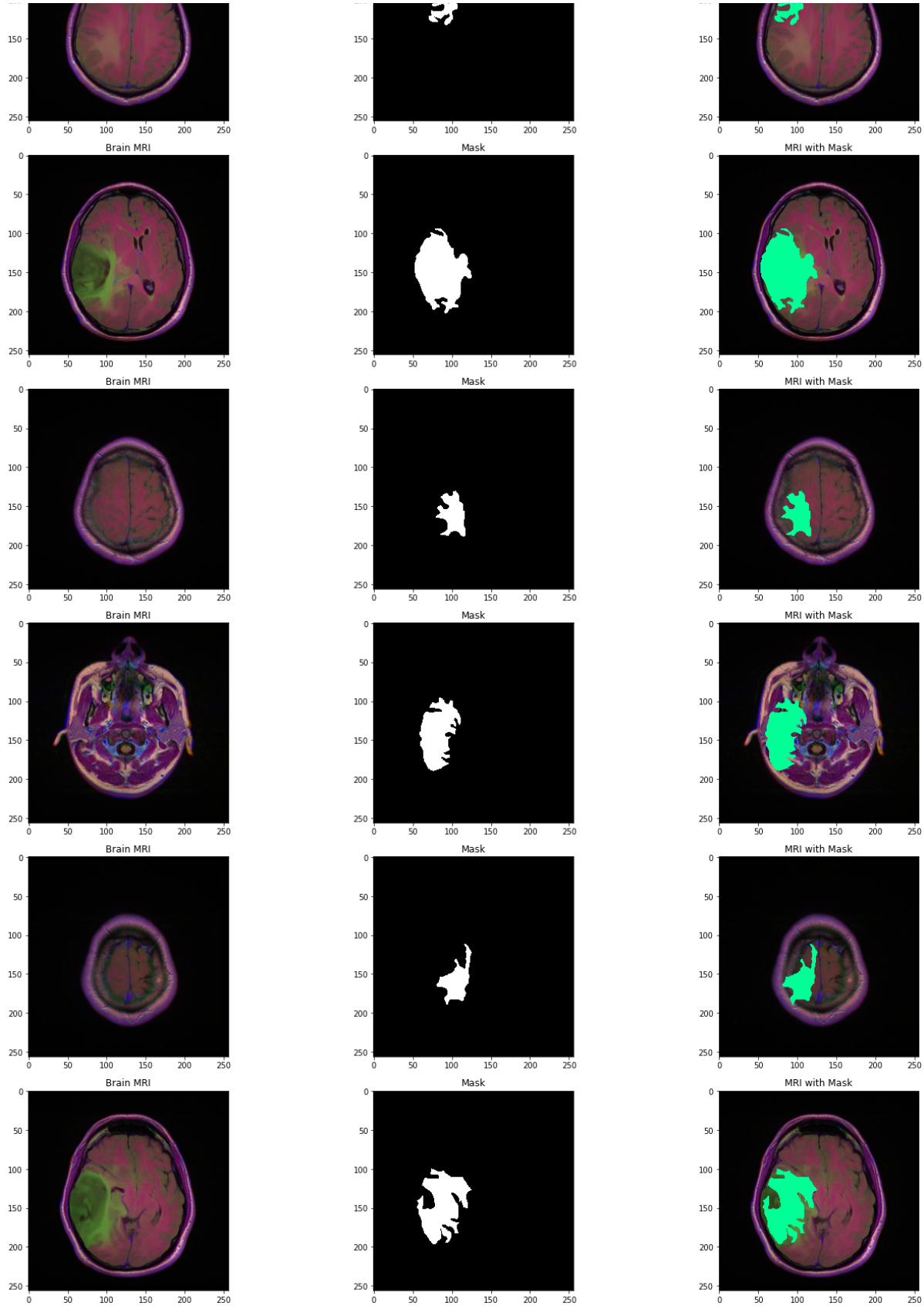

Brain MRI
Mask - 0
Brain MRI
Mask - 0
Brain MRI
Mask - 0
Brain MRI
Mask - 1
Brain MRI
Mask - 0
0
50
100
150
200
250


Brain MRI
Mask - 0

In [15]:

```
count = 0
i = 0
fig,axs = plt.subplots(12,3, figsize=(20,50))
for mask in brain_df['mask']:
    if (mask==1):
        img = io.imread(brain_df.image_path[i])
        axs[count][0].title.set_text("Brain MRI")
        axs[count][0].imshow(img)

        mask = io.imread(brain_df.mask_path[i])
        axs[count][1].title.set_text("Mask")
        axs[count][1].imshow(mask, cmap='gray')

        img[mask==255] = (0,255,150)  # change pixel color at the position of mask
        axs[count][2].title.set_text("MRI with Mask")
        axs[count][2].imshow(img)
        count +=1
    i += 1
    if (count==12):
        break

fig.tight_layout()
```


Brain MRI
Mask
MRI with Mask



5: CRETING TEST, TRAIN & VAL SET

In [49]:

```
brain_df_train = brain_df.drop(columns=['patient_id'])
# Convert the data in mask column to string format, to use categorical mode in flow_from_d
ataframe
brain_df_train['mask'] = brain_df_train['mask'].apply(lambda x: str(x))
brain_df_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3929 entries, 0 to 3928
Data columns (total 3 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   image_path  3929 non-null   object
 1   mask_path   3929 non-null   object
 2   mask        3929 non-null   object
dtypes: object(3)
memory usage: 92.2+ KB
```

In [50]:

```
brain_df_train.to_csv("brain-data1.csv")
```

In [13]:

```
from sklearn.model_selection import train_test_split

train, test = train_test_split(brain_df_train, test_size=0.15)
```

In [14]:

```
from keras_preprocessing.image import ImageDataGenerator

datagen = ImageDataGenerator(rescale=1./255., validation_split=0.1)

train_generator = datagen.flow_from_dataframe(train,
                                              directory='./',
                                              x_col='image_path',
                                              y_col='mask',
                                              subset='training',
                                              class_mode='categorical',
                                              batch_size=16,
                                              shuffle=True,
                                              target_size=(256,256)
                                             )
valid_generator = datagen.flow_from_dataframe(train,
                                              directory='./',
                                              x_col='image_path',
                                              y_col='mask',
                                              subset='validation',
                                              class_mode='categorical',
                                              batch_size=16,
                                              shuffle=True,
                                              target_size=(256,256)
                                             )
test_datagen = ImageDataGenerator(rescale=1./255.)
test_generator = test_datagen.flow_from_dataframe(test,
                                                  directory='./',
                                                  x_col='image_path',
                                                  y_col='mask',
                                                  class_mode='categorical',
                                                  batch_size=16,
                                                  shuffle=False,
                                                  target_size=(256,256)
                                                 )
```

```
Found 3006 validated image filenames belonging to 2 classes.
Found 333 validated image filenames belonging to 2 classes.
Found 590 validated image filenames belonging to 2 classes.
```

6: TRAIN A CLASSIFIER MODEL TO DETECT IF TUMOR EXISTS OR NOT

In [15]:

```
from tensorflow.keras.applications.resnet50 import ResNet50
clf_model = ResNet50(weights='imagenet', include_top=False, input_tensor=Input(shape=(256,
256,3)))
clf_model.summary()
```

```
WARNING:tensorflow:From /home/ec2-user/anaconda3/envs/tensorflow_p36/cpu/li
b/python3.6/site-packages/tensorflow_core/python/ops/resource_variable_ops.
py:1630: calling BaseResourceVariable.__init__ (from tensorflow.python.ops.
resource_variable_ops) with constraint is deprecated and will be removed in
a future version.
Instructions for updating:
If using Keras pass *_constraint arguments to layers.
Downloading data from https://github.com/keras-team/keras-applications/rele
ases/download/resnet/resnet50_weights_tf_dim_ordering_tf_kernels_notop.h5
94773248/94765736 [==============================] - 2s 0us/step
WARNING:tensorflow:OMP_NUM_THREADS is no longer used by the default Keras c
onfig. To configure the number of threads, use tf.config.threading APIs.
Model: "resnet50"
__________________________________________________________________________
________________________
Layer (type)                    Output Shape         Param #     Connected
to
==========================================================================
========================
input_1 (InputLayer)            [(None, 256, 256, 3) 0
__________________________________________________________________________
________________________
conv1_pad (ZeroPadding2D)       (None, 262, 262, 3)  0           input_1[0]
[0]
__________________________________________________________________________
________________________
conv1_conv (Conv2D)             (None, 128, 128, 64) 9472        conv1_pad
[0][0]
__________________________________________________________________________
________________________
conv1_bn (BatchNormalization)   (None, 128, 128, 64) 256         conv1_conv
[0][0]
__________________________________________________________________________
________________________
conv1_relu (Activation)         (None, 128, 128, 64) 0           conv1_bn
[0][0]
__________________________________________________________________________
________________________
pool1_pad (ZeroPadding2D)       (None, 130, 130, 64) 0           conv1_relu
[0][0]
__________________________________________________________________________
________________________
pool1_pool (MaxPooling2D)       (None, 64, 64, 64)   0           pool1_pad
[0][0]
__________________________________________________________________________
________________________
conv2_block1_1_conv (Conv2D)    (None, 64, 64, 64)   4160        pool1_pool
[0][0]
__________________________________________________________________________
________________________
conv2_block1_1_bn (BatchNormali (None, 64, 64, 64)   256         conv2_bloc
k1_1_conv[0][0]
__________________________________________________________________________
________________________
conv2_block1_1_relu (Activation (None, 64, 64, 64)   0           conv2_bloc
```

```
k1_1_bn[0][0]
__________________________________________________________________________________________________
conv2_block1_2_conv (Conv2D)    (None, 64, 64, 64)   36928       conv2_bloc
k1_1_relu[0][0]
__________________________________________________________________________________________________
conv2_block1_2_bn (BatchNormali (None, 64, 64, 64)   256         conv2_bloc
k1_2_conv[0][0]
__________________________________________________________________________________________________
conv2_block1_2_relu (Activation (None, 64, 64, 64)   0           conv2_bloc
k1_2_bn[0][0]
__________________________________________________________________________________________________
conv2_block1_0_conv (Conv2D)    (None, 64, 64, 256)  16640       pool1_pool
[0][0]
__________________________________________________________________________________________________
conv2_block1_3_conv (Conv2D)    (None, 64, 64, 256)  16640       conv2_bloc
k1_2_relu[0][0]
__________________________________________________________________________________________________
conv2_block1_0_bn (BatchNormali (None, 64, 64, 256)  1024        conv2_bloc
k1_0_conv[0][0]
__________________________________________________________________________________________________
conv2_block1_3_bn (BatchNormali (None, 64, 64, 256)  1024        conv2_bloc
k1_3_conv[0][0]
__________________________________________________________________________________________________
conv2_block1_add (Add)          (None, 64, 64, 256)  0           conv2_bloc
k1_0_bn[0][0]
                                                                 conv2_bloc
k1_3_bn[0][0]
__________________________________________________________________________________________________
conv2_block1_out (Activation)   (None, 64, 64, 256)  0           conv2_bloc
k1_add[0][0]
__________________________________________________________________________________________________
conv2_block2_1_conv (Conv2D)    (None, 64, 64, 64)   16448       conv2_bloc
k1_out[0][0]
__________________________________________________________________________________________________
conv2_block2_1_bn (BatchNormali (None, 64, 64, 64)   256         conv2_bloc
k2_1_conv[0][0]
__________________________________________________________________________________________________
conv2_block2_1_relu (Activation (None, 64, 64, 64)   0           conv2_bloc
k2_1_bn[0][0]
__________________________________________________________________________________________________
conv2_block2_2_conv (Conv2D)    (None, 64, 64, 64)   36928       conv2_bloc
k2_1_relu[0][0]
__________________________________________________________________________________________________
```

```
__________________________
conv2_block2_2_bn (BatchNormali (None, 64, 64, 64)   256         conv2_bloc
k2_2_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block2_2_relu (Activation (None, 64, 64, 64)   0           conv2_bloc
k2_2_bn[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block2_3_conv (Conv2D)    (None, 64, 64, 256)  16640       conv2_bloc
k2_2_relu[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block2_3_bn (BatchNormali (None, 64, 64, 256)  1024        conv2_bloc
k2_3_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block2_add (Add)          (None, 64, 64, 256)  0           conv2_bloc
k1_out[0][0]
                                                                 conv2_bloc
k2_3_bn[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block2_out (Activation)   (None, 64, 64, 256)  0           conv2_bloc
k2_add[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block3_1_conv (Conv2D)    (None, 64, 64, 64)   16448       conv2_bloc
k2_out[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block3_1_bn (BatchNormali (None, 64, 64, 64)   256         conv2_bloc
k3_1_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block3_1_relu (Activation (None, 64, 64, 64)   0           conv2_bloc
k3_1_bn[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block3_2_conv (Conv2D)    (None, 64, 64, 64)   36928       conv2_bloc
k3_1_relu[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block3_2_bn (BatchNormali (None, 64, 64, 64)   256         conv2_bloc
k3_2_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block3_2_relu (Activation (None, 64, 64, 64)   0           conv2_bloc
k3_2_bn[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block3_3_conv (Conv2D)    (None, 64, 64, 256)  16640       conv2_bloc
k3_2_relu[0][0]
__________________________________________________________________________________________________
__________________________
conv2_block3_3_bn (BatchNormali (None, 64, 64, 256)  1024        conv2_bloc
```

```
k3_3_conv[0][0]
__________________________________________________________________________________________________
conv2_block3_add (Add)          (None, 64, 64, 256)  0           conv2_bloc
k2_out[0][0]
                                                                 conv2_bloc
k3_3_bn[0][0]
__________________________________________________________________________________________________
conv2_block3_out (Activation)   (None, 64, 64, 256)  0           conv2_bloc
k3_add[0][0]
__________________________________________________________________________________________________
conv3_block1_1_conv (Conv2D)    (None, 32, 32, 128)  32896       conv2_bloc
k3_out[0][0]
__________________________________________________________________________________________________
conv3_block1_1_bn (BatchNormali (None, 32, 32, 128)  512         conv3_bloc
k1_1_conv[0][0]
__________________________________________________________________________________________________
conv3_block1_1_relu (Activation (None, 32, 32, 128)  0           conv3_bloc
k1_1_bn[0][0]
__________________________________________________________________________________________________
conv3_block1_2_conv (Conv2D)    (None, 32, 32, 128)  147584      conv3_bloc
k1_1_relu[0][0]
__________________________________________________________________________________________________
conv3_block1_2_bn (BatchNormali (None, 32, 32, 128)  512         conv3_bloc
k1_2_conv[0][0]
__________________________________________________________________________________________________
conv3_block1_2_relu (Activation (None, 32, 32, 128)  0           conv3_bloc
k1_2_bn[0][0]
__________________________________________________________________________________________________
conv3_block1_0_conv (Conv2D)    (None, 32, 32, 512)  131584      conv2_bloc
k3_out[0][0]
__________________________________________________________________________________________________
conv3_block1_3_conv (Conv2D)    (None, 32, 32, 512)  66048       conv3_bloc
k1_2_relu[0][0]
__________________________________________________________________________________________________
conv3_block1_0_bn (BatchNormali (None, 32, 32, 512)  2048        conv3_bloc
k1_0_conv[0][0]
__________________________________________________________________________________________________
conv3_block1_3_bn (BatchNormali (None, 32, 32, 512)  2048        conv3_bloc
k1_3_conv[0][0]
__________________________________________________________________________________________________
conv3_block1_add (Add)          (None, 32, 32, 512)  0           conv3_bloc
k1_0_bn[0][0]
                                                                 conv3_bloc
```

```
k1_3_bn[0][0]
__________________________________________________________________________________________________
conv3_block1_out (Activation)   (None, 32, 32, 512)  0           conv3_bloc
k1_add[0][0]
__________________________________________________________________________________________________
conv3_block2_1_conv (Conv2D)    (None, 32, 32, 128)  65664       conv3_bloc
k1_out[0][0]
__________________________________________________________________________________________________
conv3_block2_1_bn (BatchNormali (None, 32, 32, 128)  512         conv3_bloc
k2_1_conv[0][0]
__________________________________________________________________________________________________
conv3_block2_1_relu (Activation (None, 32, 32, 128)  0           conv3_bloc
k2_1_bn[0][0]
__________________________________________________________________________________________________
conv3_block2_2_conv (Conv2D)    (None, 32, 32, 128)  147584      conv3_bloc
k2_1_relu[0][0]
__________________________________________________________________________________________________
conv3_block2_2_bn (BatchNormali (None, 32, 32, 128)  512         conv3_bloc
k2_2_conv[0][0]
__________________________________________________________________________________________________
conv3_block2_2_relu (Activation (None, 32, 32, 128)  0           conv3_bloc
k2_2_bn[0][0]
__________________________________________________________________________________________________
conv3_block2_3_conv (Conv2D)    (None, 32, 32, 512)  66048       conv3_bloc
k2_2_relu[0][0]
__________________________________________________________________________________________________
conv3_block2_3_bn (BatchNormali (None, 32, 32, 512)  2048        conv3_bloc
k2_3_conv[0][0]
__________________________________________________________________________________________________
conv3_block2_add (Add)          (None, 32, 32, 512)  0           conv3_bloc
k1_out[0][0]
                                                                 conv3_bloc
k2_3_bn[0][0]
__________________________________________________________________________________________________
conv3_block2_out (Activation)   (None, 32, 32, 512)  0           conv3_bloc
k2_add[0][0]
__________________________________________________________________________________________________
conv3_block3_1_conv (Conv2D)    (None, 32, 32, 128)  65664       conv3_bloc
k2_out[0][0]
__________________________________________________________________________________________________
conv3_block3_1_bn (BatchNormali (None, 32, 32, 128)  512         conv3_bloc
k3_1_conv[0][0]
__________________________________________________________________________________________
```

```
__________________________
conv3_block3_1_relu (Activation (None, 32, 32, 128)  0           conv3_bloc
k3_1_bn[0][0]
__________________________________________________________________________
__________________________
conv3_block3_2_conv (Conv2D)    (None, 32, 32, 128)  147584      conv3_bloc
k3_1_relu[0][0]
__________________________________________________________________________
__________________________
conv3_block3_2_bn (BatchNormali (None, 32, 32, 128)  512         conv3_bloc
k3_2_conv[0][0]
__________________________________________________________________________
__________________________
conv3_block3_2_relu (Activation (None, 32, 32, 128)  0           conv3_bloc
k3_2_bn[0][0]
__________________________________________________________________________
__________________________
conv3_block3_3_conv (Conv2D)    (None, 32, 32, 512)  66048       conv3_bloc
k3_2_relu[0][0]
__________________________________________________________________________
__________________________
conv3_block3_3_bn (BatchNormali (None, 32, 32, 512)  2048        conv3_bloc
k3_3_conv[0][0]
__________________________________________________________________________
__________________________
conv3_block3_add (Add)          (None, 32, 32, 512)  0           conv3_bloc
k2_out[0][0]
                                                                 conv3_bloc
k3_3_bn[0][0]
__________________________________________________________________________
__________________________
conv3_block3_out (Activation)   (None, 32, 32, 512)  0           conv3_bloc
k3_add[0][0]
__________________________________________________________________________
__________________________
conv3_block4_1_conv (Conv2D)    (None, 32, 32, 128)  65664       conv3_bloc
k3_out[0][0]
__________________________________________________________________________
__________________________
conv3_block4_1_bn (BatchNormali (None, 32, 32, 128)  512         conv3_bloc
k4_1_conv[0][0]
__________________________________________________________________________
__________________________
conv3_block4_1_relu (Activation (None, 32, 32, 128)  0           conv3_bloc
k4_1_bn[0][0]
__________________________________________________________________________
__________________________
conv3_block4_2_conv (Conv2D)    (None, 32, 32, 128)  147584      conv3_bloc
k4_1_relu[0][0]
__________________________________________________________________________
__________________________
conv3_block4_2_bn (BatchNormali (None, 32, 32, 128)  512         conv3_bloc
k4_2_conv[0][0]
__________________________________________________________________________
__________________________
conv3_block4_2_relu (Activation (None, 32, 32, 128)  0           conv3_bloc
```

```
k4_2_bn[0][0]
__________________________________________________________________________________________________
conv3_block4_3_conv (Conv2D)    (None, 32, 32, 512)  66048       conv3_bloc
k4_2_relu[0][0]
__________________________________________________________________________________________________
conv3_block4_3_bn (BatchNormali (None, 32, 32, 512)  2048        conv3_bloc
k4_3_conv[0][0]
__________________________________________________________________________________________________
conv3_block4_add (Add)          (None, 32, 32, 512)  0           conv3_bloc
k3_out[0][0]
                                                                 conv3_bloc
k4_3_bn[0][0]
__________________________________________________________________________________________________
conv3_block4_out (Activation)   (None, 32, 32, 512)  0           conv3_bloc
k4_add[0][0]
__________________________________________________________________________________________________
conv4_block1_1_conv (Conv2D)    (None, 16, 16, 256)  131328      conv3_bloc
k4_out[0][0]
__________________________________________________________________________________________________
conv4_block1_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k1_1_conv[0][0]
__________________________________________________________________________________________________
conv4_block1_1_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k1_1_bn[0][0]
__________________________________________________________________________________________________
conv4_block1_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_bloc
k1_1_relu[0][0]
__________________________________________________________________________________________________
conv4_block1_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k1_2_conv[0][0]
__________________________________________________________________________________________________
conv4_block1_2_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k1_2_bn[0][0]
__________________________________________________________________________________________________
conv4_block1_0_conv (Conv2D)    (None, 16, 16, 1024) 525312      conv3_bloc
k4_out[0][0]
__________________________________________________________________________________________________
conv4_block1_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_bloc
k1_2_relu[0][0]
__________________________________________________________________________________________________
conv4_block1_0_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_bloc
k1_0_conv[0][0]
__________________________________________________________________________________________________
```

```
__________________________
conv4_block1_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_bloc
k1_3_conv[0][0]
__________________________________________________________________________
__________________________
conv4_block1_add (Add)          (None, 16, 16, 1024) 0           conv4_bloc
k1_0_bn[0][0]
                                                                 conv4_bloc
k1_3_bn[0][0]
__________________________________________________________________________
__________________________
conv4_block1_out (Activation)   (None, 16, 16, 1024) 0           conv4_bloc
k1_add[0][0]
__________________________________________________________________________
__________________________
conv4_block2_1_conv (Conv2D)    (None, 16, 16, 256)  262400      conv4_bloc
k1_out[0][0]
__________________________________________________________________________
__________________________
conv4_block2_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k2_1_conv[0][0]
__________________________________________________________________________
__________________________
conv4_block2_1_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k2_1_bn[0][0]
__________________________________________________________________________
__________________________
conv4_block2_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_bloc
k2_1_relu[0][0]
__________________________________________________________________________
__________________________
conv4_block2_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k2_2_conv[0][0]
__________________________________________________________________________
__________________________
conv4_block2_2_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k2_2_bn[0][0]
__________________________________________________________________________
__________________________
conv4_block2_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_bloc
k2_2_relu[0][0]
__________________________________________________________________________
__________________________
conv4_block2_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_bloc
k2_3_conv[0][0]
__________________________________________________________________________
__________________________
conv4_block2_add (Add)          (None, 16, 16, 1024) 0           conv4_bloc
k1_out[0][0]
                                                                 conv4_bloc
k2_3_bn[0][0]
__________________________________________________________________________
__________________________
conv4_block2_out (Activation)   (None, 16, 16, 1024) 0           conv4_bloc
k2_add[0][0]
__________________________________________________________________________
```

```
__________________________
conv4_block3_1_conv (Conv2D)    (None, 16, 16, 256)  262400      conv4_bloc
k2_out[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block3_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k3_1_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block3_1_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k3_1_bn[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block3_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_bloc
k3_1_relu[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block3_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k3_2_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block3_2_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k3_2_bn[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block3_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_bloc
k3_2_relu[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block3_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_bloc
k3_3_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block3_add (Add)          (None, 16, 16, 1024) 0           conv4_bloc
k2_out[0][0]
                                                                 conv4_bloc
k3_3_bn[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block3_out (Activation)   (None, 16, 16, 1024) 0           conv4_bloc
k3_add[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block4_1_conv (Conv2D)    (None, 16, 16, 256)  262400      conv4_bloc
k3_out[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block4_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k4_1_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block4_1_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k4_1_bn[0][0]
__________________________________________________________________________________________________
__________________________
conv4_block4_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_bloc
```

```
k4_1_relu[0][0]
__________________________________________________________________________________________________
conv4_block4_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k4_2_conv[0][0]
__________________________________________________________________________________________________
conv4_block4_2_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k4_2_bn[0][0]
__________________________________________________________________________________________________
conv4_block4_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_bloc
k4_2_relu[0][0]
__________________________________________________________________________________________________
conv4_block4_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_bloc
k4_3_conv[0][0]
__________________________________________________________________________________________________
conv4_block4_add (Add)          (None, 16, 16, 1024) 0           conv4_bloc
k3_out[0][0]
                                                                 conv4_bloc
k4_3_bn[0][0]
__________________________________________________________________________________________________
conv4_block4_out (Activation)   (None, 16, 16, 1024) 0           conv4_bloc
k4_add[0][0]
__________________________________________________________________________________________________
conv4_block5_1_conv (Conv2D)    (None, 16, 16, 256)  262400      conv4_bloc
k4_out[0][0]
__________________________________________________________________________________________________
conv4_block5_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k5_1_conv[0][0]
__________________________________________________________________________________________________
conv4_block5_1_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k5_1_bn[0][0]
__________________________________________________________________________________________________
conv4_block5_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_bloc
k5_1_relu[0][0]
__________________________________________________________________________________________________
conv4_block5_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k5_2_conv[0][0]
__________________________________________________________________________________________________
conv4_block5_2_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k5_2_bn[0][0]
__________________________________________________________________________________________________
conv4_block5_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_bloc
k5_2_relu[0][0]
__________________________________________________________________________________________________
```

```
__________________________
conv4_block5_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_bloc
k5_3_conv[0][0]
__________________________________________________________________________
__________________________
conv4_block5_add (Add)          (None, 16, 16, 1024) 0           conv4_bloc
k4_out[0][0]
                                                                 conv4_bloc
k5_3_bn[0][0]
__________________________________________________________________________
__________________________
conv4_block5_out (Activation)   (None, 16, 16, 1024) 0           conv4_bloc
k5_add[0][0]
__________________________________________________________________________
__________________________
conv4_block6_1_conv (Conv2D)    (None, 16, 16, 256)  262400      conv4_bloc
k5_out[0][0]
__________________________________________________________________________
__________________________
conv4_block6_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k6_1_conv[0][0]
__________________________________________________________________________
__________________________
conv4_block6_1_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k6_1_bn[0][0]
__________________________________________________________________________
__________________________
conv4_block6_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_bloc
k6_1_relu[0][0]
__________________________________________________________________________
__________________________
conv4_block6_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_bloc
k6_2_conv[0][0]
__________________________________________________________________________
__________________________
conv4_block6_2_relu (Activation (None, 16, 16, 256)  0           conv4_bloc
k6_2_bn[0][0]
__________________________________________________________________________
__________________________
conv4_block6_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_bloc
k6_2_relu[0][0]
__________________________________________________________________________
__________________________
conv4_block6_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_bloc
k6_3_conv[0][0]
__________________________________________________________________________
__________________________
conv4_block6_add (Add)          (None, 16, 16, 1024) 0           conv4_bloc
k5_out[0][0]
                                                                 conv4_bloc
k6_3_bn[0][0]
__________________________________________________________________________
__________________________
conv4_block6_out (Activation)   (None, 16, 16, 1024) 0           conv4_bloc
k6_add[0][0]
__________________________________________________________________________
```

```
__________________________
conv5_block1_1_conv (Conv2D)    (None, 8, 8, 512)    524800      conv4_bloc
k6_out[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block1_1_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_bloc
k1_1_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block1_1_relu (Activation (None, 8, 8, 512)    0           conv5_bloc
k1_1_bn[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block1_2_conv (Conv2D)    (None, 8, 8, 512)    2359808     conv5_bloc
k1_1_relu[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block1_2_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_bloc
k1_2_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block1_2_relu (Activation (None, 8, 8, 512)    0           conv5_bloc
k1_2_bn[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block1_0_conv (Conv2D)    (None, 8, 8, 2048)   2099200     conv4_bloc
k6_out[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block1_3_conv (Conv2D)    (None, 8, 8, 2048)   1050624     conv5_bloc
k1_2_relu[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block1_0_bn (BatchNormali (None, 8, 8, 2048)   8192        conv5_bloc
k1_0_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block1_3_bn (BatchNormali (None, 8, 8, 2048)   8192        conv5_bloc
k1_3_conv[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block1_add (Add)          (None, 8, 8, 2048)   0           conv5_bloc
k1_0_bn[0][0]
                                                                 conv5_bloc
k1_3_bn[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block1_out (Activation)   (None, 8, 8, 2048)   0           conv5_bloc
k1_add[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block2_1_conv (Conv2D)    (None, 8, 8, 512)    1049088     conv5_bloc
k1_out[0][0]
__________________________________________________________________________________________________
__________________________
conv5_block2_1_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_bloc
```

```
k2_1_conv[0][0]
__________________________________________________________________________________________________
conv5_block2_1_relu (Activation (None, 8, 8, 512)    0           conv5_bloc
k2_1_bn[0][0]
__________________________________________________________________________________________________
conv5_block2_2_conv (Conv2D)    (None, 8, 8, 512)    2359808     conv5_bloc
k2_1_relu[0][0]
__________________________________________________________________________________________________
conv5_block2_2_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_bloc
k2_2_conv[0][0]
__________________________________________________________________________________________________
conv5_block2_2_relu (Activation (None, 8, 8, 512)    0           conv5_bloc
k2_2_bn[0][0]
__________________________________________________________________________________________________
conv5_block2_3_conv (Conv2D)    (None, 8, 8, 2048)   1050624     conv5_bloc
k2_2_relu[0][0]
__________________________________________________________________________________________________
conv5_block2_3_bn (BatchNormali (None, 8, 8, 2048)   8192        conv5_bloc
k2_3_conv[0][0]
__________________________________________________________________________________________________
conv5_block2_add (Add)          (None, 8, 8, 2048)   0           conv5_bloc
k1_out[0][0]
                                                                 conv5_bloc
k2_3_bn[0][0]
__________________________________________________________________________________________________
conv5_block2_out (Activation)   (None, 8, 8, 2048)   0           conv5_bloc
k2_add[0][0]
__________________________________________________________________________________________________
conv5_block3_1_conv (Conv2D)    (None, 8, 8, 512)    1049088     conv5_bloc
k2_out[0][0]
__________________________________________________________________________________________________
conv5_block3_1_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_bloc
k3_1_conv[0][0]
__________________________________________________________________________________________________
conv5_block3_1_relu (Activation (None, 8, 8, 512)    0           conv5_bloc
k3_1_bn[0][0]
__________________________________________________________________________________________________
conv5_block3_2_conv (Conv2D)    (None, 8, 8, 512)    2359808     conv5_bloc
k3_1_relu[0][0]
__________________________________________________________________________________________________
conv5_block3_2_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_bloc
k3_2_conv[0][0]
__________________________________________________________________________________________________
```

```
__________________________
conv5_block3_2_relu (Activation (None, 8, 8, 512)    0           conv5_bloc
k3_2_bn[0][0]
__________________________________________________________________________
__________________________
conv5_block3_3_conv (Conv2D)    (None, 8, 8, 2048)   1050624     conv5_bloc
k3_2_relu[0][0]
__________________________________________________________________________
__________________________
conv5_block3_3_bn (BatchNormali (None, 8, 8, 2048)   8192        conv5_bloc
k3_3_conv[0][0]
__________________________________________________________________________
__________________________
conv5_block3_add (Add)          (None, 8, 8, 2048)   0           conv5_bloc
k2_out[0][0]
                                                                 conv5_bloc
k3_3_bn[0][0]
__________________________________________________________________________
__________________________
conv5_block3_out (Activation)   (None, 8, 8, 2048)   0           conv5_bloc
k3_add[0][0]
==========================================================================
========================
Total params: 23,587,712
Trainable params: 23,534,592
Non-trainable params: 53,120
__________________________________________________________________________
__________________________
```

In [16]:

```
# before this i tried with trainable layer but the accuracy was less as compared
for layer in clf_model.layers:
    layers.trainable = False
```

In [17]:

```
head = clf_model.output
head = AveragePooling2D(pool_size=(4,4))(head)
head = Flatten(name='Flatten')(head)
head = Dense(256, activation='relu')(head)
head = Dropout(0.3)(head)
head = Dense(256, activation='relu')(head)
head = Dropout(0.3)(head)
head = Dense(2, activation='softmax')(head)

model = Model(clf_model.input, head)
model.compile(loss = 'categorical_crossentropy',
              optimizer='adam',
              metrics= ["accuracy"]
             )
model.summary()
```

```
Model: "model"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            [(None, 256, 256, 3) 0
__________________________________________________________________________________________________
conv1_pad (ZeroPadding2D)       (None, 262, 262, 3)  0           input_1[0]
[0]
__________________________________________________________________________________________________
conv1_conv (Conv2D)             (None, 128, 128, 64) 9472        conv1_pad[0]
[0]
__________________________________________________________________________________________________
conv1_bn (BatchNormalization)   (None, 128, 128, 64) 256         conv1_conv
[0][0]
__________________________________________________________________________________________________
conv1_relu (Activation)         (None, 128, 128, 64) 0           conv1_bn[0]
[0]
__________________________________________________________________________________________________
pool1_pad (ZeroPadding2D)       (None, 130, 130, 64) 0           conv1_relu
[0][0]
__________________________________________________________________________________________________
pool1_pool (MaxPooling2D)       (None, 64, 64, 64)   0           pool1_pad[0]
[0]
__________________________________________________________________________________________________
conv2_block1_1_conv (Conv2D)    (None, 64, 64, 64)   4160        pool1_pool
[0][0]
__________________________________________________________________________________________________
conv2_block1_1_bn (BatchNormali (None, 64, 64, 64)   256         conv2_block1
_1_conv[0][0]
__________________________________________________________________________________________________
conv2_block1_1_relu (Activation (None, 64, 64, 64)   0           conv2_block1
_1_bn[0][0]
__________________________________________________________________________________________________
conv2_block1_2_conv (Conv2D)    (None, 64, 64, 64)   36928       conv2_block1
_1_relu[0][0]
__________________________________________________________________________________________________
conv2_block1_2_bn (BatchNormali (None, 64, 64, 64)   256         conv2_block1
_2_conv[0][0]
__________________________________________________________________________________________________
conv2_block1_2_relu (Activation (None, 64, 64, 64)   0           conv2_block1
_2_bn[0][0]
```

```
__________________________________________________________________________________________________
_______________________
conv2_block1_0_conv (Conv2D)    (None, 64, 64, 256)  16640       pool1_pool
[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block1_3_conv (Conv2D)    (None, 64, 64, 256)  16640       conv2_block1
_2_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block1_0_bn (BatchNormali (None, 64, 64, 256)  1024        conv2_block1
_0_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block1_3_bn (BatchNormali (None, 64, 64, 256)  1024        conv2_block1
_3_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block1_add (Add)          (None, 64, 64, 256)  0           conv2_block1
_0_bn[0][0]
                                                                  conv2_block1
_3_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block1_out (Activation)   (None, 64, 64, 256)  0           conv2_block1
_add[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block2_1_conv (Conv2D)    (None, 64, 64, 64)   16448       conv2_block1
_out[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block2_1_bn (BatchNormali (None, 64, 64, 64)   256         conv2_block2
_1_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block2_1_relu (Activation (None, 64, 64, 64)   0           conv2_block2
_1_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block2_2_conv (Conv2D)    (None, 64, 64, 64)   36928       conv2_block2
_1_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block2_2_bn (BatchNormali (None, 64, 64, 64)   256         conv2_block2
_2_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block2_2_relu (Activation (None, 64, 64, 64)   0           conv2_block2
_2_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv2_block2_3_conv (Conv2D)    (None, 64, 64, 256)  16640       conv2_block2
_2_relu[0][0]
__________________________________________________________________________________________________
_______________________
```

```
conv2_block2_3_bn (BatchNormali (None, 64, 64, 256)  1024        conv2_block2
_3_conv[0][0]
__________________________________________________________________________________________________
conv2_block2_add (Add)          (None, 64, 64, 256)  0           conv2_block1
_out[0][0]
                                                                 conv2_block2
_3_bn[0][0]
__________________________________________________________________________________________________
conv2_block2_out (Activation)   (None, 64, 64, 256)  0           conv2_block2
_add[0][0]
__________________________________________________________________________________________________
conv2_block3_1_conv (Conv2D)    (None, 64, 64, 64)   16448       conv2_block2
_out[0][0]
__________________________________________________________________________________________________
conv2_block3_1_bn (BatchNormali (None, 64, 64, 64)   256         conv2_block3
_1_conv[0][0]
__________________________________________________________________________________________________
conv2_block3_1_relu (Activation (None, 64, 64, 64)   0           conv2_block3
_1_bn[0][0]
__________________________________________________________________________________________________
conv2_block3_2_conv (Conv2D)    (None, 64, 64, 64)   36928       conv2_block3
_1_relu[0][0]
__________________________________________________________________________________________________
conv2_block3_2_bn (BatchNormali (None, 64, 64, 64)   256         conv2_block3
_2_conv[0][0]
__________________________________________________________________________________________________
conv2_block3_2_relu (Activation (None, 64, 64, 64)   0           conv2_block3
_2_bn[0][0]
__________________________________________________________________________________________________
conv2_block3_3_conv (Conv2D)    (None, 64, 64, 256)  16640       conv2_block3
_2_relu[0][0]
__________________________________________________________________________________________________
conv2_block3_3_bn (BatchNormali (None, 64, 64, 256)  1024        conv2_block3
_3_conv[0][0]
__________________________________________________________________________________________________
conv2_block3_add (Add)          (None, 64, 64, 256)  0           conv2_block2
_out[0][0]
                                                                 conv2_block3
_3_bn[0][0]
__________________________________________________________________________________________________
conv2_block3_out (Activation)   (None, 64, 64, 256)  0           conv2_block3
_add[0][0]
__________________________________________________________________________________________________
```

```
conv3_block1_1_conv (Conv2D)    (None, 32, 32, 128)  32896       conv2_block3
_out[0][0]
__________________________________________________________________________________________________
conv3_block1_1_bn (BatchNormali (None, 32, 32, 128)  512         conv3_block1
_1_conv[0][0]
__________________________________________________________________________________________________
conv3_block1_1_relu (Activation (None, 32, 32, 128)  0           conv3_block1
_1_bn[0][0]
__________________________________________________________________________________________________
conv3_block1_2_conv (Conv2D)    (None, 32, 32, 128)  147584      conv3_block1
_1_relu[0][0]
__________________________________________________________________________________________________
conv3_block1_2_bn (BatchNormali (None, 32, 32, 128)  512         conv3_block1
_2_conv[0][0]
__________________________________________________________________________________________________
conv3_block1_2_relu (Activation (None, 32, 32, 128)  0           conv3_block1
_2_bn[0][0]
__________________________________________________________________________________________________
conv3_block1_0_conv (Conv2D)    (None, 32, 32, 512)  131584      conv2_block3
_out[0][0]
__________________________________________________________________________________________________
conv3_block1_3_conv (Conv2D)    (None, 32, 32, 512)  66048       conv3_block1
_2_relu[0][0]
__________________________________________________________________________________________________
conv3_block1_0_bn (BatchNormali (None, 32, 32, 512)  2048        conv3_block1
_0_conv[0][0]
__________________________________________________________________________________________________
conv3_block1_3_bn (BatchNormali (None, 32, 32, 512)  2048        conv3_block1
_3_conv[0][0]
__________________________________________________________________________________________________
conv3_block1_add (Add)          (None, 32, 32, 512)  0           conv3_block1
_0_bn[0][0]
                                                                 conv3_block1
_3_bn[0][0]
__________________________________________________________________________________________________
conv3_block1_out (Activation)   (None, 32, 32, 512)  0           conv3_block1
_add[0][0]
__________________________________________________________________________________________________
conv3_block2_1_conv (Conv2D)    (None, 32, 32, 128)  65664       conv3_block1
_out[0][0]
__________________________________________________________________________________________________
conv3_block2_1_bn (BatchNormali (None, 32, 32, 128)  512         conv3_block2
_1_conv[0][0]
```

```
__________________________________________________________________________________________________
_______________________
conv3_block2_1_relu (Activation (None, 32, 32, 128)  0           conv3_block2
_1_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block2_2_conv (Conv2D)    (None, 32, 32, 128)  147584      conv3_block2
_1_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block2_2_bn (BatchNormali (None, 32, 32, 128)  512         conv3_block2
_2_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block2_2_relu (Activation (None, 32, 32, 128)  0           conv3_block2
_2_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block2_3_conv (Conv2D)    (None, 32, 32, 512)  66048       conv3_block2
_2_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block2_3_bn (BatchNormali (None, 32, 32, 512)  2048        conv3_block2
_3_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block2_add (Add)          (None, 32, 32, 512)  0           conv3_block1
_out[0][0]
                                                                 conv3_block2
_3_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block2_out (Activation)   (None, 32, 32, 512)  0           conv3_block2
_add[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block3_1_conv (Conv2D)    (None, 32, 32, 128)  65664       conv3_block2
_out[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block3_1_bn (BatchNormali (None, 32, 32, 128)  512         conv3_block3
_1_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block3_1_relu (Activation (None, 32, 32, 128)  0           conv3_block3
_1_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block3_2_conv (Conv2D)    (None, 32, 32, 128)  147584      conv3_block3
_1_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block3_2_bn (BatchNormali (None, 32, 32, 128)  512         conv3_block3
_2_conv[0][0]
__________________________________________________________________________________________________
_______________________
```

```
conv3_block3_2_relu (Activation (None, 32, 32, 128)  0           conv3_block3
_2_bn[0][0]
__________________________________________________________________________________________________
conv3_block3_3_conv (Conv2D)    (None, 32, 32, 512)  66048       conv3_block3
_2_relu[0][0]
__________________________________________________________________________________________________
conv3_block3_3_bn (BatchNormali (None, 32, 32, 512)  2048        conv3_block3
_3_conv[0][0]
__________________________________________________________________________________________________
conv3_block3_add (Add)          (None, 32, 32, 512)  0           conv3_block2
_out[0][0]
                                                                 conv3_block3
_3_bn[0][0]
__________________________________________________________________________________________________
conv3_block3_out (Activation)   (None, 32, 32, 512)  0           conv3_block3
_add[0][0]
__________________________________________________________________________________________________
conv3_block4_1_conv (Conv2D)    (None, 32, 32, 128)  65664       conv3_block3
_out[0][0]
__________________________________________________________________________________________________
conv3_block4_1_bn (BatchNormali (None, 32, 32, 128)  512         conv3_block4
_1_conv[0][0]
__________________________________________________________________________________________________
conv3_block4_1_relu (Activation (None, 32, 32, 128)  0           conv3_block4
_1_bn[0][0]
__________________________________________________________________________________________________
conv3_block4_2_conv (Conv2D)    (None, 32, 32, 128)  147584      conv3_block4
_1_relu[0][0]
__________________________________________________________________________________________________
conv3_block4_2_bn (BatchNormali (None, 32, 32, 128)  512         conv3_block4
_2_conv[0][0]
__________________________________________________________________________________________________
conv3_block4_2_relu (Activation (None, 32, 32, 128)  0           conv3_block4
_2_bn[0][0]
__________________________________________________________________________________________________
conv3_block4_3_conv (Conv2D)    (None, 32, 32, 512)  66048       conv3_block4
_2_relu[0][0]
__________________________________________________________________________________________________
conv3_block4_3_bn (BatchNormali (None, 32, 32, 512)  2048        conv3_block4
_3_conv[0][0]
__________________________________________________________________________________________________
conv3_block4_add (Add)          (None, 32, 32, 512)  0           conv3_block3
_out[0][0]
```

```
                                                                    conv3_block4
_3_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv3_block4_out (Activation)   (None, 32, 32, 512)  0           conv3_block4
_add[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_1_conv (Conv2D)    (None, 16, 16, 256)  131328      conv3_block4
_out[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block1
_1_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_1_relu (Activation (None, 16, 16, 256)  0           conv4_block1
_1_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_block1
_1_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block1
_2_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_2_relu (Activation (None, 16, 16, 256)  0           conv4_block1
_2_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_0_conv (Conv2D)    (None, 16, 16, 1024) 525312      conv3_block4
_out[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_block1
_2_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_0_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_block1
_0_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_block1
_3_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_add (Add)          (None, 16, 16, 1024) 0           conv4_block1
_0_bn[0][0]
                                                                    conv4_block1
_3_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block1_out (Activation)   (None, 16, 16, 1024) 0           conv4_block1
_add[0][0]
```

```
__________________________________________________________________________________________________
_______________________
conv4_block2_1_conv (Conv2D)    (None, 16, 16, 256)  262400      conv4_block1
_out[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block2_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block2
_1_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block2_1_relu (Activation (None, 16, 16, 256)  0           conv4_block2
_1_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block2_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_block2
_1_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block2_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block2
_2_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block2_2_relu (Activation (None, 16, 16, 256)  0           conv4_block2
_2_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block2_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_block2
_2_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block2_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_block2
_3_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block2_add (Add)          (None, 16, 16, 1024) 0           conv4_block1
_out[0][0]
                                                                 conv4_block2
_3_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block2_out (Activation)   (None, 16, 16, 1024) 0           conv4_block2
_add[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block3_1_conv (Conv2D)    (None, 16, 16, 256)  262400      conv4_block2
_out[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block3_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block3
_1_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block3_1_relu (Activation (None, 16, 16, 256)  0           conv4_block3
_1_bn[0][0]
__________________________________________________________________________________________________
_______________________
```

```
conv4_block3_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_block3
_1_relu[0][0]
__________________________________________________________________________________________________
conv4_block3_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block3
_2_conv[0][0]
__________________________________________________________________________________________________
conv4_block3_2_relu (Activation (None, 16, 16, 256)  0           conv4_block3
_2_bn[0][0]
__________________________________________________________________________________________________
conv4_block3_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_block3
_2_relu[0][0]
__________________________________________________________________________________________________
conv4_block3_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_block3
_3_conv[0][0]
__________________________________________________________________________________________________
conv4_block3_add (Add)          (None, 16, 16, 1024) 0           conv4_block2
_out[0][0]
                                                                 conv4_block3
_3_bn[0][0]
__________________________________________________________________________________________________
conv4_block3_out (Activation)   (None, 16, 16, 1024) 0           conv4_block3
_add[0][0]
__________________________________________________________________________________________________
conv4_block4_1_conv (Conv2D)    (None, 16, 16, 256)  262400      conv4_block3
_out[0][0]
__________________________________________________________________________________________________
conv4_block4_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block4
_1_conv[0][0]
__________________________________________________________________________________________________
conv4_block4_1_relu (Activation (None, 16, 16, 256)  0           conv4_block4
_1_bn[0][0]
__________________________________________________________________________________________________
conv4_block4_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_block4
_1_relu[0][0]
__________________________________________________________________________________________________
conv4_block4_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block4
_2_conv[0][0]
__________________________________________________________________________________________________
conv4_block4_2_relu (Activation (None, 16, 16, 256)  0           conv4_block4
_2_bn[0][0]
__________________________________________________________________________________________________
conv4_block4_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_block4
_2_relu[0][0]
```

```
__________________________________________________________________________________________________
_____________________
conv4_block4_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_block4
_3_conv[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block4_add (Add)          (None, 16, 16, 1024) 0           conv4_block3
_out[0][0]
                                                                 conv4_block4
_3_bn[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block4_out (Activation)   (None, 16, 16, 1024) 0           conv4_block4
_add[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block5_1_conv (Conv2D)    (None, 16, 16, 256)  262400      conv4_block4
_out[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block5_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block5
_1_conv[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block5_1_relu (Activation (None, 16, 16, 256)  0           conv4_block5
_1_bn[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block5_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_block5
_1_relu[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block5_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block5
_2_conv[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block5_2_relu (Activation (None, 16, 16, 256)  0           conv4_block5
_2_bn[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block5_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_block5
_2_relu[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block5_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_block5
_3_conv[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block5_add (Add)          (None, 16, 16, 1024) 0           conv4_block4
_out[0][0]
                                                                 conv4_block5
_3_bn[0][0]
__________________________________________________________________________________________________
_____________________
conv4_block5_out (Activation)   (None, 16, 16, 1024) 0           conv4_block5
_add[0][0]
```

```
__________________________________________________________________________________________________
_______________________
conv4_block6_1_conv (Conv2D)    (None, 16, 16, 256)  262400      conv4_block5
_out[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block6_1_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block6
_1_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block6_1_relu (Activation (None, 16, 16, 256)  0           conv4_block6
_1_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block6_2_conv (Conv2D)    (None, 16, 16, 256)  590080      conv4_block6
_1_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block6_2_bn (BatchNormali (None, 16, 16, 256)  1024        conv4_block6
_2_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block6_2_relu (Activation (None, 16, 16, 256)  0           conv4_block6
_2_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block6_3_conv (Conv2D)    (None, 16, 16, 1024) 263168      conv4_block6
_2_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block6_3_bn (BatchNormali (None, 16, 16, 1024) 4096        conv4_block6
_3_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block6_add (Add)          (None, 16, 16, 1024) 0           conv4_block5
_out[0][0]
                                                                 conv4_block6
_3_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv4_block6_out (Activation)   (None, 16, 16, 1024) 0           conv4_block6
_add[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block1_1_conv (Conv2D)    (None, 8, 8, 512)    524800      conv4_block6
_out[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block1_1_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_block1
_1_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block1_1_relu (Activation (None, 8, 8, 512)    0           conv5_block1
_1_bn[0][0]
__________________________________________________________________________________________________
_______________________
```

```
conv5_block1_2_conv (Conv2D)    (None, 8, 8, 512)    2359808     conv5_block1
_1_relu[0][0]
__________________________________________________________________________________________________
conv5_block1_2_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_block1
_2_conv[0][0]
__________________________________________________________________________________________________
conv5_block1_2_relu (Activation (None, 8, 8, 512)    0           conv5_block1
_2_bn[0][0]
__________________________________________________________________________________________________
conv5_block1_0_conv (Conv2D)    (None, 8, 8, 2048)   2099200     conv4_block6
_out[0][0]
__________________________________________________________________________________________________
conv5_block1_3_conv (Conv2D)    (None, 8, 8, 2048)   1050624     conv5_block1
_2_relu[0][0]
__________________________________________________________________________________________________
conv5_block1_0_bn (BatchNormali (None, 8, 8, 2048)   8192        conv5_block1
_0_conv[0][0]
__________________________________________________________________________________________________
conv5_block1_3_bn (BatchNormali (None, 8, 8, 2048)   8192        conv5_block1
_3_conv[0][0]
__________________________________________________________________________________________________
conv5_block1_add (Add)          (None, 8, 8, 2048)   0           conv5_block1
_0_bn[0][0]
                                                                 conv5_block1
_3_bn[0][0]
__________________________________________________________________________________________________
conv5_block1_out (Activation)   (None, 8, 8, 2048)   0           conv5_block1
_add[0][0]
__________________________________________________________________________________________________
conv5_block2_1_conv (Conv2D)    (None, 8, 8, 512)    1049088     conv5_block1
_out[0][0]
__________________________________________________________________________________________________
conv5_block2_1_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_block2
_1_conv[0][0]
__________________________________________________________________________________________________
conv5_block2_1_relu (Activation (None, 8, 8, 512)    0           conv5_block2
_1_bn[0][0]
__________________________________________________________________________________________________
conv5_block2_2_conv (Conv2D)    (None, 8, 8, 512)    2359808     conv5_block2
_1_relu[0][0]
__________________________________________________________________________________________________
conv5_block2_2_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_block2
_2_conv[0][0]
```

```
__________________________________________________________________________________________________
_______________________
conv5_block2_2_relu (Activation (None, 8, 8, 512)    0           conv5_block2
_2_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block2_3_conv (Conv2D)    (None, 8, 8, 2048)   1050624     conv5_block2
_2_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block2_3_bn (BatchNormali (None, 8, 8, 2048)   8192        conv5_block2
_3_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block2_add (Add)          (None, 8, 8, 2048)   0           conv5_block1
_out[0][0]
                                                                 conv5_block2
_3_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block2_out (Activation)   (None, 8, 8, 2048)   0           conv5_block2
_add[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block3_1_conv (Conv2D)    (None, 8, 8, 512)    1049088     conv5_block2
_out[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block3_1_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_block3
_1_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block3_1_relu (Activation (None, 8, 8, 512)    0           conv5_block3
_1_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block3_2_conv (Conv2D)    (None, 8, 8, 512)    2359808     conv5_block3
_1_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block3_2_bn (BatchNormali (None, 8, 8, 512)    2048        conv5_block3
_2_conv[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block3_2_relu (Activation (None, 8, 8, 512)    0           conv5_block3
_2_bn[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block3_3_conv (Conv2D)    (None, 8, 8, 2048)   1050624     conv5_block3
_2_relu[0][0]
__________________________________________________________________________________________________
_______________________
conv5_block3_3_bn (BatchNormali (None, 8, 8, 2048)   8192        conv5_block3
_3_conv[0][0]
__________________________________________________________________________________________________
_______________________
```

```
conv5_block3_add (Add)          (None, 8, 8, 2048)   0           conv5_block2
_out[0][0]
                                                                 conv5_block3
_3_bn[0][0]
__________________________________________________________________________________________________
conv5_block3_out (Activation)   (None, 8, 8, 2048)   0           conv5_block3
_add[0][0]
__________________________________________________________________________________________________
average_pooling2d (AveragePooli (None, 2, 2, 2048)   0           conv5_block3
_out[0][0]
__________________________________________________________________________________________________
Flatten (Flatten)               (None, 8192)         0           average_pool
ing2d[0][0]
__________________________________________________________________________________________________
dense (Dense)                   (None, 256)          2097408     Flatten[0]
[0]
__________________________________________________________________________________________________
dropout (Dropout)               (None, 256)          0           dense[0][0]
__________________________________________________________________________________________________
dense_1 (Dense)                 (None, 256)          65792       dropout[0]
[0]
__________________________________________________________________________________________________
dropout_1 (Dropout)             (None, 256)          0           dense_1[0]
[0]
__________________________________________________________________________________________________
dense_2 (Dense)                 (None, 2)            514         dropout_1[0]
[0]
==================================================================================================
Total params: 25,751,426
Trainable params: 25,698,306
Non-trainable params: 53,120
__________________________________________________________________________________________________
```

In [18]:

```
earlystopping = EarlyStopping(monitor='val_loss',
                              mode='min',
                              verbose=1,
                              patience=15
                             )
checkpointer = ModelCheckpoint(filepath="clf-resnet-weights.hdf5",
                               verbose=1,
                               save_best_only=True
                              )
reduce_lr = ReduceLROnPlateau(monitor='val_loss',
                              mode='min',
                              verbose=1,
                              patience=10,
                              min_delta=0.0001,
                              factor=0.2
                             )
callbacks = [checkpointer, earlystopping, reduce_lr]
```

In [ ]:

```
h = model.fit(train_generator,
              steps_per_epoch= train_generator.n // train_generator.batch_size,
              epochs = 5,
              validation_data= valid_generator,
              validation_steps= valid_generator.n // valid_generator.batch_size,
              callbacks=[checkpointer, earlystopping])
```

```
WARNING:tensorflow:From /home/ec2-user/anaconda3/envs/tensorflow_p36/lib/pyth
on3.6/site-packages/horovod/tensorflow/__init__.py:163: The name tf.global_va
riables is deprecated. Please use tf.compat.v1.global_variables instead.

WARNING:tensorflow:From /home/ec2-user/anaconda3/envs/tensorflow_p36/lib/pyth
on3.6/site-packages/horovod/tensorflow/__init__.py:189: The name tf.get_defau
lt_graph is deprecated. Please use tf.compat.v1.get_default_graph instead.

[2022-05-23 11:43:45.239 ip-172-16-74-117:15211 INFO utils.py:27] RULE_JOB_ST
OP_SIGNAL_FILENAME: None
[2022-05-23 11:43:45.364 ip-172-16-74-117:15211 INFO profiler_config_parser.p
y:111] Unable to find config at /opt/ml/input/config/profilerconfig.json. Pro
filer is disabled.
Epoch 1/5
```

In [ ]:

```
model
```

In [22]:

```
# saving model achitecture in json file
model_json = model.to_json()
with open("clf-resnet-model.json", "w") as json_file:
    json_file.write(model_json)
```

7: CLASSIFIACTION MODEL EVALUATION

In [23]:

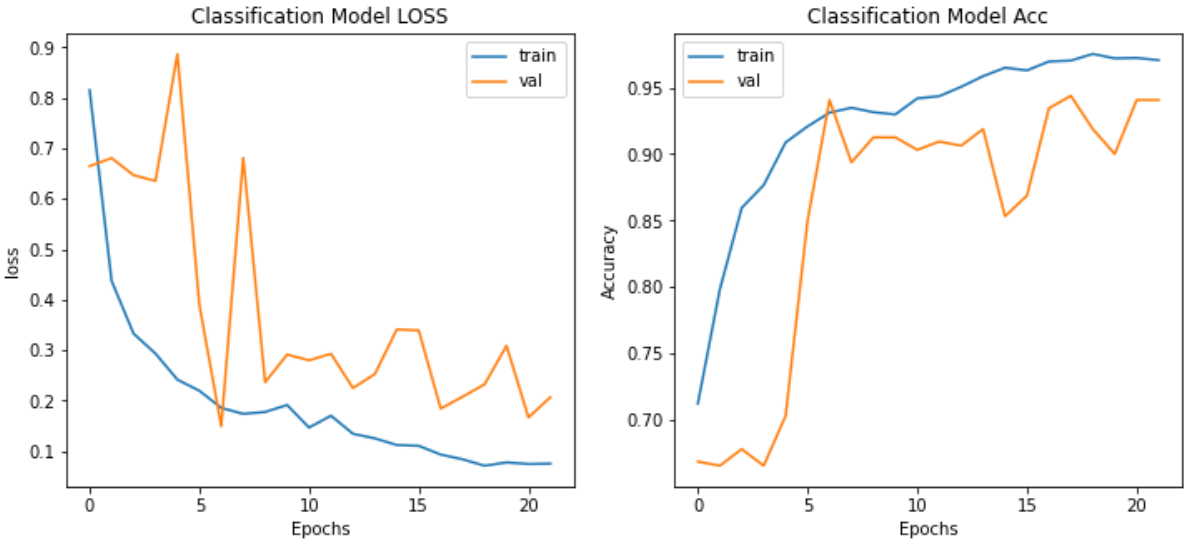
```
h.history.keys()
```

Out[23]:

```
dict_keys(['loss', 'accuracy', 'val_loss', 'val_accuracy'])
```

In [24]:

```
plt.figure(figsize=(12,5))
plt.subplot(1,2,1)
plt.plot(h.history['loss']);
plt.plot(h.history['val_loss']);
plt.title("Classification Model LOSS");
plt.ylabel("loss");
plt.xlabel("Epochs");
plt.legend(['train', 'val']);

plt.subplot(1,2,2)
plt.plot(h.history['accuracy']);
plt.plot(h.history['val_accuracy']);
plt.title("Classification Model Acc");
plt.ylabel("Accuracy");
plt.xlabel("Epochs");
plt.legend(['train', 'val']);
```



In [25]:

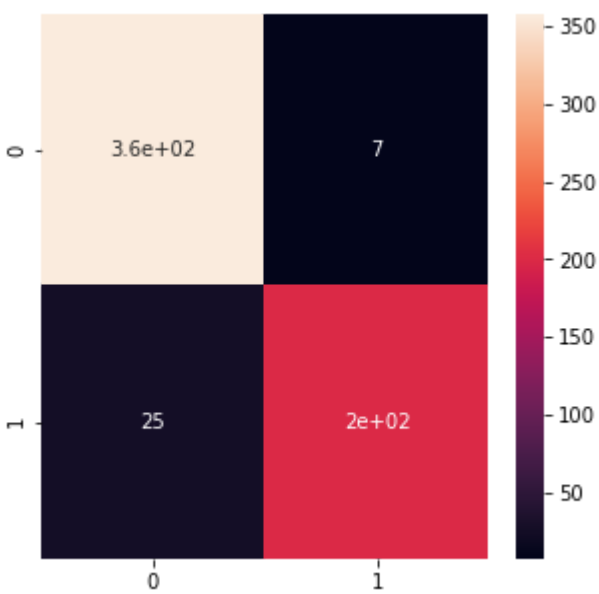
```
_, acc = model.evaluate(test_generator)
print("Test accuracy : {} %".format(acc*100))
```

```
37/37 [==============================] - 5s 132ms/step - loss: 0.1995 - accur
acy: 0.9458
Test accuracy : 94.57626938819885 %
```

In [26]:

```
prediction = model.predict(test_generator)

pred = np.argmax(prediction, axis=1)
#pred = np.asarray(pred).astype('str')
original = np.asarray(test['mask']).astype('int')

from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
accuracy = accuracy_score(original, pred)
print(accuracy)

cm = confusion_matrix(original, pred)

report = classification_report(original, pred, labels = [0,1])
print(report)
plt.figure(figsize = (5,5))
sns.heatmap(cm, annot=True);
```

```
0.9457627118644067
              precision    recall  f1-score   support

           0       0.93      0.98      0.96       365
           1       0.97      0.89      0.93       225

    accuracy                           0.95       590
   macro avg       0.95      0.93      0.94       590
weighted avg       0.95      0.95      0.95       590
```



8: BUILDING A SEGMENTATION MODEL TO LOCALIZE TUMOR

In [27]:

```
brain_df_mask = brain_df[brain_df['mask'] == 1]
brain_df_mask.shape
```

Out[27]:

```
(1373, 4)
```

In [28]:

```
# creating test, train and val sets
X_train, X_val = train_test_split(brain_df_mask, test_size=0.15)
X_test, X_val = train_test_split(X_val, test_size=0.5)
print("Train size is {}, valid size is {} & test size is {}".format(len(X_train), len(X_va
l), len(X_test)))

train_ids = list(X_train.image_path)
train_mask = list(X_train.mask_path)

val_ids = list(X_val.image_path)
val_mask= list(X_val.mask_path)
```

```
Train size is 1167, valid size is 103 & test size is 103
```

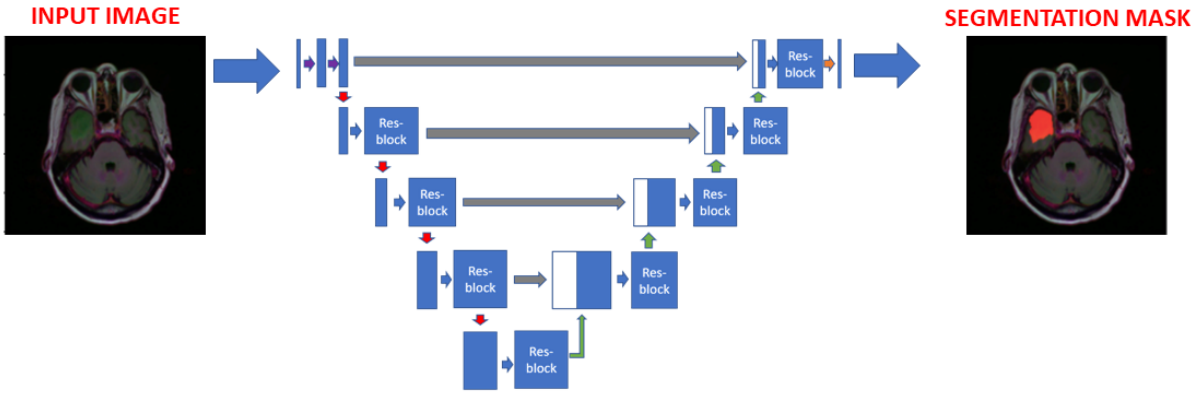
In [29]:

```
class DataGenerator(tf.keras.utils.Sequence):
  def __init__(self, ids , mask, image_dir = './', batch_size = 16, img_h = 256, img_w = 2
56, shuffle = True):

    self.ids = ids
    self.mask = mask
    self.image_dir = image_dir
    self.batch_size = batch_size
    self.img_h = img_h
    self.img_w = img_w
    self.shuffle = shuffle
    self.on_epoch_end()

  def __len__(self):
    'Get the number of batches per epoch'

    return int(np.floor(len(self.ids)) / self.batch_size)

  def __getitem__(self, index):
    'Generate a batch of data'

    #generate index of batch_size length
    indexes = self.indexes[index* self.batch_size : (index+1) * self.batch_size]

    #get the ImageId corresponding to the indexes created above based on batch size
    list_ids = [self.ids[i] for i in indexes]

    #get the MaskId corresponding to the indexes created above based on batch size
    list_mask = [self.mask[i] for i in indexes]


    #generate data for the X(features) and y(label)
    X, y = self.__data_generation(list_ids, list_mask)

    #returning the data
    return X, y

  def on_epoch_end(self):
    'Used for updating the indices after each epoch, once at the beginning as well as at t
he end of each epoch'

    #getting the array of indices based on the input dataframe
    self.indexes = np.arange(len(self.ids))

    #if shuffle is true, shuffle the indices
    if self.shuffle:
      np.random.shuffle(self.indexes)

  def __data_generation(self, list_ids, list_mask):
    'generate the data corresponding the indexes in a given batch of images'

    # create empty arrays of shape (batch_size,height,width,depth)
    #Depth is 3 for input and depth is taken as 1 for output becasue mask consist only of
```

```
 1 channel.
    X = np.empty((self.batch_size, self.img_h, self.img_w, 3))
    y = np.empty((self.batch_size, self.img_h, self.img_w, 1))

    #iterate through the dataframe rows, whose size is equal to the batch_size
    for i in range(len(list_ids)):
      #path of the image
      img_path = str(list_ids[i])

      #mask path
      mask_path = str(list_mask[i])

      #reading the original image and the corresponding mask image
      img = io.imread(img_path)
      mask = io.imread(mask_path)

      #resizing and coverting them to array of type float64
      img = cv2.resize(img,(self.img_h,self.img_w))
      img = np.array(img, dtype = np.float64)

      mask = cv2.resize(mask,(self.img_h,self.img_w))
      mask = np.array(mask, dtype = np.float64)

      #standardising
      img -= img.mean()
      img /= img.std()

      mask -= mask.mean()
      mask /= mask.std()

      #Adding image to the empty array
      X[i,] = img

      #expanding the dimnesion of the image from (256,256) to (256,256,1)
      y[i,] = np.expand_dims(mask, axis = 2)

    #normalizing y
    y = (y > 0).astype(int)

    return X, y

train_data = DataGenerator(train_ids, train_mask)
val_data = DataGenerator(val_ids, val_mask)
```

RESUNET

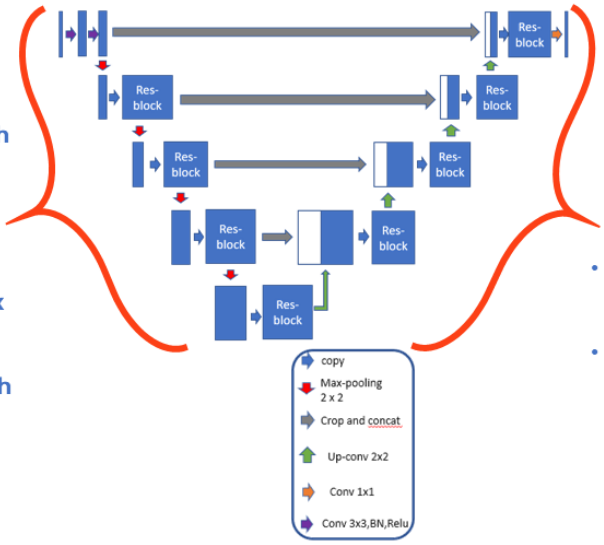
- ResUNet architecture combines UNet backbone architecture with residual blocks to overcome the vanishing gradients problems present in deep architectures.
- Unet architecture is based on Fully Convolutional Networks and modified in a way that it performs well on segmentation tasks.
- Resunet consists of three parts:
  - (1) Encoder or contracting path
  - (2) Bottleneck
  - (3) Decoder or expansive path



RESUNET ARCHITECTURE:

CONTRACTION PATH (ENCODER)

- The contraction path consist of several contraction blocks, each block takes an input that passes through res-blocks followed by 2x2 max pooling. Feature maps after each block doubles, which helps the model learn complex features effectively.



EXPANSION PATH (DECODER)

- Significant advantage of this architecture lies in expansion or decoder section. Each block takes in the up-sampled input from the previous layer and concatenates with the corresponding output features from the res-blocks in the contraction path. This is then again passed through the res-block followed by 2x2 up-sampling convolution layers.
- This helps to ensure that features learned while contracting are used while reconstructing the image.
- Finally in the last layer of expansion path, the output from the res-block is passed through 1x1 convolution layer to produce the desired output with the same size as the input.

BOTTLENECK

- The bottleneck block, serves as a connection between contraction path and expansion path.
- The block takes the input and then passes through a res-block followed by 2x2 up-sampling convolution layers.

In [30]:

```
# lets create model now
def resblock(X, f):
    '''
    function for creating res block
    '''
    X_copy = X  #copy of input

    # main path
    X = Conv2D(f, kernel_size=(1,1), kernel_initializer='he_normal')(X)
    X = BatchNormalization()(X)
    X = Activation('relu')(X)

    X = Conv2D(f, kernel_size=(3,3), padding='same', kernel_initializer='he_normal')(X)
    X = BatchNormalization()(X)

    # shortcut path
    X_copy = Conv2D(f, kernel_size=(1,1), kernel_initializer='he_normal')(X_copy)
    X_copy = BatchNormalization()(X_copy)

    # Adding the output from main path and short path together
    X = Add()([X, X_copy])
    X = Activation('relu')(X)

    return X

def upsample_concat(x, skip):
    '''
    funtion for upsampling image
    '''
    X = UpSampling2D((2,2))(x)
    merge = Concatenate()([X, skip])

    return merge
```

In [31]:

```
input_shape = (256,256,3)
X_input = Input(input_shape) #iniating tensor of input shape

# Stage 1
conv_1 = Conv2D(16, 3, activation='relu', padding='same', kernel_initializer='he_normal')(
X_input)
conv_1 = BatchNormalization()(conv_1)
conv_1 = Conv2D(16, 3, activation='relu', padding='same', kernel_initializer='he_normal')(
conv_1)
conv_1 = BatchNormalization()(conv_1)
pool_1 = MaxPool2D((2,2))(conv_1)

# stage 2
conv_2 = resblock(pool_1, 32)
pool_2 = MaxPool2D((2,2))(conv_2)

# Stage 3
conv_3 = resblock(pool_2, 64)
pool_3 = MaxPool2D((2,2))(conv_3)

# Stage 4
conv_4 = resblock(pool_3, 128)
pool_4 = MaxPool2D((2,2))(conv_4)

# Stage 5 (bottle neck)
conv_5 = resblock(pool_4, 256)

# Upsample Stage 1
up_1 = upsample_concat(conv_5, conv_4)
up_1 = resblock(up_1, 128)

# Upsample Stage 2
up_2 = upsample_concat(up_1, conv_3)
up_2 = resblock(up_2, 64)

# Upsample Stage 3
up_3 = upsample_concat(up_2, conv_2)
up_3 = resblock(up_3, 32)

# Upsample Stage 4
up_4 = upsample_concat(up_3, conv_1)
up_4 = resblock(up_4, 16)

# final output
out = Conv2D(1, (1,1), kernel_initializer='he_normal', padding='same', activation='sigmoi
d')(up_4)

seg_model = Model(X_input, out)
seg_model.summary()
```

```
Model: "functional_3"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_2 (InputLayer)            [(None, 256, 256, 3) 0
__________________________________________________________________________________________________
conv2d (Conv2D)                 (None, 256, 256, 16) 448         input_2[0][0]
__________________________________________________________________________________________________
batch_normalization (BatchNorma (None, 256, 256, 16) 64          conv2d[0][0]
__________________________________________________________________________________________________
conv2d_1 (Conv2D)               (None, 256, 256, 16) 2320        batch_normalization[0][0]
__________________________________________________________________________________________________
batch_normalization_1 (BatchNor (None, 256, 256, 16) 64          conv2d_1[0][0]
__________________________________________________________________________________________________
max_pooling2d (MaxPooling2D)    (None, 128, 128, 16) 0           batch_normalization_1[0][0]
__________________________________________________________________________________________________
conv2d_2 (Conv2D)               (None, 128, 128, 32) 544         max_pooling2d[0][0]
__________________________________________________________________________________________________
batch_normalization_2 (BatchNor (None, 128, 128, 32) 128         conv2d_2[0][0]
__________________________________________________________________________________________________
activation (Activation)         (None, 128, 128, 32) 0           batch_normalization_2[0][0]
__________________________________________________________________________________________________
conv2d_3 (Conv2D)               (None, 128, 128, 32) 9248        activation[0][0]
__________________________________________________________________________________________________
conv2d_4 (Conv2D)               (None, 128, 128, 32) 544         max_pooling2d[0][0]
__________________________________________________________________________________________________
batch_normalization_3 (BatchNor (None, 128, 128, 32) 128         conv2d_3[0][0]
__________________________________________________________________________________________________
batch_normalization_4 (BatchNor (None, 128, 128, 32) 128         conv2d_4
```

```
[0][0]
__________________________________________________________________________________________________
add (Add)                       (None, 128, 128, 32) 0           batch_norm
alization_3[0][0]
                                                                 batch_norm
alization_4[0][0]
__________________________________________________________________________________________________
activation_1 (Activation)       (None, 128, 128, 32) 0           add[0][0]
__________________________________________________________________________________________________
max_pooling2d_1 (MaxPooling2D)  (None, 64, 64, 32)   0           activation
_1[0][0]
__________________________________________________________________________________________________
conv2d_5 (Conv2D)               (None, 64, 64, 64)   2112        max_poolin
g2d_1[0][0]
__________________________________________________________________________________________________
batch_normalization_5 (BatchNor (None, 64, 64, 64)   256         conv2d_5
[0][0]
__________________________________________________________________________________________________
activation_2 (Activation)       (None, 64, 64, 64)   0           batch_norm
alization_5[0][0]
__________________________________________________________________________________________________
conv2d_6 (Conv2D)               (None, 64, 64, 64)   36928       activation
_2[0][0]
__________________________________________________________________________________________________
conv2d_7 (Conv2D)               (None, 64, 64, 64)   2112        max_poolin
g2d_1[0][0]
__________________________________________________________________________________________________
batch_normalization_6 (BatchNor (None, 64, 64, 64)   256         conv2d_6
[0][0]
__________________________________________________________________________________________________
batch_normalization_7 (BatchNor (None, 64, 64, 64)   256         conv2d_7
[0][0]
__________________________________________________________________________________________________
add_1 (Add)                     (None, 64, 64, 64)   0           batch_norm
alization_6[0][0]
                                                                 batch_norm
alization_7[0][0]
__________________________________________________________________________________________________
activation_3 (Activation)       (None, 64, 64, 64)   0           add_1[0]
[0]
__________________________________________________________________________________________________
max_pooling2d_2 (MaxPooling2D)  (None, 32, 32, 64)   0           activation
_3[0][0]
```

```
__________________________________________________________________________________________________
conv2d_8 (Conv2D)               (None, 32, 32, 128)  8320        max_poolin
g2d_2[0][0]
__________________________________________________________________________________________________
batch_normalization_8 (BatchNor (None, 32, 32, 128)  512         conv2d_8
[0][0]
__________________________________________________________________________________________________
activation_4 (Activation)       (None, 32, 32, 128)  0           batch_norm
alization_8[0][0]
__________________________________________________________________________________________________
conv2d_9 (Conv2D)               (None, 32, 32, 128)  147584      activation
_4[0][0]
__________________________________________________________________________________________________
conv2d_10 (Conv2D)              (None, 32, 32, 128)  8320        max_poolin
g2d_2[0][0]
__________________________________________________________________________________________________
batch_normalization_9 (BatchNor (None, 32, 32, 128)  512         conv2d_9
[0][0]
__________________________________________________________________________________________________
batch_normalization_10 (BatchNo (None, 32, 32, 128)  512         conv2d_10
[0][0]
__________________________________________________________________________________________________
add_2 (Add)                     (None, 32, 32, 128)  0           batch_norm
alization_9[0][0]
                                                                 batch_norm
alization_10[0][0]
__________________________________________________________________________________________________
activation_5 (Activation)       (None, 32, 32, 128)  0           add_2[0]
[0]
__________________________________________________________________________________________________
max_pooling2d_3 (MaxPooling2D)  (None, 16, 16, 128)  0           activation
_5[0][0]
__________________________________________________________________________________________________
conv2d_11 (Conv2D)              (None, 16, 16, 256)  33024       max_poolin
g2d_3[0][0]
__________________________________________________________________________________________________
batch_normalization_11 (BatchNo (None, 16, 16, 256)  1024        conv2d_11
[0][0]
__________________________________________________________________________________________________
activation_6 (Activation)       (None, 16, 16, 256)  0           batch_norm
alization_11[0][0]
__________________________________________________________________________________________________
```

```
conv2d_12 (Conv2D)              (None, 16, 16, 256)  590080      activation
_6[0][0]
__________________________________________________________________________________________________
conv2d_13 (Conv2D)              (None, 16, 16, 256)  33024       max_poolin
g2d_3[0][0]
__________________________________________________________________________________________________
batch_normalization_12 (BatchNo (None, 16, 16, 256)  1024        conv2d_12
[0][0]
__________________________________________________________________________________________________
batch_normalization_13 (BatchNo (None, 16, 16, 256)  1024        conv2d_13
[0][0]
__________________________________________________________________________________________________
add_3 (Add)                     (None, 16, 16, 256)  0           batch_norm
alization_12[0][0]
                                                                 batch_norm
alization_13[0][0]
__________________________________________________________________________________________________
activation_7 (Activation)       (None, 16, 16, 256)  0           add_3[0]
[0]
__________________________________________________________________________________________________
up_sampling2d (UpSampling2D)    (None, 32, 32, 256)  0           activation
_7[0][0]
__________________________________________________________________________________________________
concatenate (Concatenate)       (None, 32, 32, 384)  0           up_samplin
g2d[0][0]
                                                                 activation
_5[0][0]
__________________________________________________________________________________________________
conv2d_14 (Conv2D)              (None, 32, 32, 128)  49280       concatenat
e[0][0]
__________________________________________________________________________________________________
batch_normalization_14 (BatchNo (None, 32, 32, 128)  512         conv2d_14
[0][0]
__________________________________________________________________________________________________
activation_8 (Activation)       (None, 32, 32, 128)  0           batch_norm
alization_14[0][0]
__________________________________________________________________________________________________
conv2d_15 (Conv2D)              (None, 32, 32, 128)  147584      activation
_8[0][0]
__________________________________________________________________________________________________
conv2d_16 (Conv2D)              (None, 32, 32, 128)  49280       concatenat
e[0][0]
__________________________________________________________________________________________________
```

```
batch_normalization_15 (BatchNo (None, 32, 32, 128)  512         conv2d_15
[0][0]
__________________________________________________________________________________________________
__________________________
batch_normalization_16 (BatchNo (None, 32, 32, 128)  512         conv2d_16
[0][0]
__________________________________________________________________________________________________
__________________________
add_4 (Add)                     (None, 32, 32, 128)  0           batch_norm
alization_15[0][0]
                                                                 batch_norm
alization_16[0][0]
__________________________________________________________________________________________________
__________________________
activation_9 (Activation)       (None, 32, 32, 128)  0           add_4[0]
[0]
__________________________________________________________________________________________________
__________________________
up_sampling2d_1 (UpSampling2D)  (None, 64, 64, 128)  0           activation
_9[0][0]
__________________________________________________________________________________________________
__________________________
concatenate_1 (Concatenate)     (None, 64, 64, 192)  0           up_samplin
g2d_1[0][0]
                                                                 activation
_3[0][0]
__________________________________________________________________________________________________
__________________________
conv2d_17 (Conv2D)              (None, 64, 64, 64)   12352       concatenat
e_1[0][0]
__________________________________________________________________________________________________
__________________________
batch_normalization_17 (BatchNo (None, 64, 64, 64)   256         conv2d_17
[0][0]
__________________________________________________________________________________________________
__________________________
activation_10 (Activation)      (None, 64, 64, 64)   0           batch_norm
alization_17[0][0]
__________________________________________________________________________________________________
__________________________
conv2d_18 (Conv2D)              (None, 64, 64, 64)   36928       activation
_10[0][0]
__________________________________________________________________________________________________
__________________________
conv2d_19 (Conv2D)              (None, 64, 64, 64)   12352       concatenat
e_1[0][0]
__________________________________________________________________________________________________
__________________________
batch_normalization_18 (BatchNo (None, 64, 64, 64)   256         conv2d_18
[0][0]
__________________________________________________________________________________________________
__________________________
batch_normalization_19 (BatchNo (None, 64, 64, 64)   256         conv2d_19
[0][0]
__________________________________________________________________________________________________
__________________________
```

```
add_5 (Add)                     (None, 64, 64, 64)   0           batch_norm
alization_18[0][0]
                                                                batch_norm
alization_19[0][0]
__________________________________________________________________________________________________
activation_11 (Activation)      (None, 64, 64, 64)   0           add_5[0]
[0]
__________________________________________________________________________________________________
up_sampling2d_2 (UpSampling2D)  (None, 128, 128, 64) 0           activation
_11[0][0]
__________________________________________________________________________________________________
concatenate_2 (Concatenate)     (None, 128, 128, 96) 0           up_samplin
g2d_2[0][0]
                                                                activation
_1[0][0]
__________________________________________________________________________________________________
conv2d_20 (Conv2D)              (None, 128, 128, 32) 3104        concatenat
e_2[0][0]
__________________________________________________________________________________________________
batch_normalization_20 (BatchNo (None, 128, 128, 32) 128         conv2d_20
[0][0]
__________________________________________________________________________________________________
activation_12 (Activation)      (None, 128, 128, 32) 0           batch_norm
alization_20[0][0]
__________________________________________________________________________________________________
conv2d_21 (Conv2D)              (None, 128, 128, 32) 9248        activation
_12[0][0]
__________________________________________________________________________________________________
conv2d_22 (Conv2D)              (None, 128, 128, 32) 3104        concatenat
e_2[0][0]
__________________________________________________________________________________________________
batch_normalization_21 (BatchNo (None, 128, 128, 32) 128         conv2d_21
[0][0]
__________________________________________________________________________________________________
batch_normalization_22 (BatchNo (None, 128, 128, 32) 128         conv2d_22
[0][0]
__________________________________________________________________________________________________
add_6 (Add)                     (None, 128, 128, 32) 0           batch_norm
alization_21[0][0]
                                                                batch_norm
alization_22[0][0]
__________________________________________________________________________________________________
activation_13 (Activation)      (None, 128, 128, 32) 0           add_6[0]
[0]
```

```
__________________________________________________________________________________________________
__________________________
up_sampling2d_3 (UpSampling2D)  (None, 256, 256, 32) 0           activation
_13[0][0]
__________________________________________________________________________________________________
__________________________
concatenate_3 (Concatenate)     (None, 256, 256, 48) 0           up_samplin
g2d_3[0][0]
                                                                 batch_norm
alization_1[0][0]
__________________________________________________________________________________________________
__________________________
conv2d_23 (Conv2D)              (None, 256, 256, 16) 784         concatenat
e_3[0][0]
__________________________________________________________________________________________________
__________________________
batch_normalization_23 (BatchNo (None, 256, 256, 16) 64          conv2d_23
[0][0]
__________________________________________________________________________________________________
__________________________
activation_14 (Activation)      (None, 256, 256, 16) 0           batch_norm
alization_23[0][0]
__________________________________________________________________________________________________
__________________________
conv2d_24 (Conv2D)              (None, 256, 256, 16) 2320        activation
_14[0][0]
__________________________________________________________________________________________________
__________________________
conv2d_25 (Conv2D)              (None, 256, 256, 16) 784         concatenat
e_3[0][0]
__________________________________________________________________________________________________
__________________________
batch_normalization_24 (BatchNo (None, 256, 256, 16) 64          conv2d_24
[0][0]
__________________________________________________________________________________________________
__________________________
batch_normalization_25 (BatchNo (None, 256, 256, 16) 64          conv2d_25
[0][0]
__________________________________________________________________________________________________
__________________________
add_7 (Add)                     (None, 256, 256, 16) 0           batch_norm
alization_24[0][0]
                                                                 batch_norm
alization_25[0][0]
__________________________________________________________________________________________________
__________________________
activation_15 (Activation)      (None, 256, 256, 16) 0           add_7[0]
[0]
__________________________________________________________________________________________________
__________________________
conv2d_26 (Conv2D)              (None, 256, 256, 1)  17          activation
_15[0][0]
==================================================================================================
========================
Total params: 1,210,513
Trainable params: 1,206,129
```

```
Non-trainable params: 4,384
_____________________________________________________________________________________________
_____________________
```

9: TRAINING SEGMENTATION MODEL

In [32]:

```
# Define a custom loss function for ResUNet model
'''
actual link for refrence (https://github.com/nabsabraham/focal-tversky-unet/blob/master/lo
sses.py)
'''
from keras.losses import binary_crossentropy

epsilon = 1e-5
smooth = 1

def tversky(y_true, y_pred):
    y_true_pos = K.flatten(y_true)
    y_pred_pos = K.flatten(y_pred)
    true_pos = K.sum(y_true_pos * y_pred_pos)
    false_neg = K.sum(y_true_pos * (1-y_pred_pos))
    false_pos = K.sum((1-y_true_pos)*y_pred_pos)
    alpha = 0.7
    return (true_pos + smooth)/(true_pos + alpha*false_neg + (1-alpha)*false_pos + smooth)

def focal_tversky(y_true,y_pred):
    y_true = tf.cast(y_true, tf.float32)
    y_pred = tf.cast(y_pred, tf.float32)

    pt_1 = tversky(y_true, y_pred)
    gamma = 0.75
    return K.pow((1-pt_1), gamma)

def tversky_loss(y_true, y_pred):
    return 1 - tversky(y_true,y_pred)
```

In [33]:

```
# compling model and callbacks functions
adam = tf.keras.optimizers.Adam(lr = 0.05, epsilon = 0.1)
seg_model.compile(optimizer = adam,
                  loss = focal_tversky,
                  metrics = [tversky]
                 )
#callbacks
earlystopping = EarlyStopping(monitor='val_loss',
                              mode='min',
                              verbose=1,
                              patience=20
                             )
# save the best model with lower validation loss
checkpointer = ModelCheckpoint(filepath="ResUNet-segModel-weights.hdf5",
                               verbose=1,
                               save_best_only=True
                              )
reduce_lr = ReduceLROnPlateau(monitor='val_loss',
                              mode='min',
                              verbose=1,
                              patience=10,
                              min_delta=0.0001,
                              factor=0.2
                             )
```

In [34]:

```
h = seg_model.fit(train_data,
                  epochs = 60,
                  validation_data = val_data,
                  callbacks = [checkpointer, earlystopping, reduce_lr]
                 )
```

```
Epoch 1/60
72/72 [==============================] - ETA: 0s - loss: 0.7294 - tversky: 0.
3389
Epoch 00001: val_loss improved from inf to 0.87963, saving model to ResUNet-s
egModel-weights.hdf5
72/72 [==============================] - 11s 152ms/step - loss: 0.7294 - tver
sky: 0.3389 - val_loss: 0.8796 - val_tversky: 0.1568
Epoch 2/60
72/72 [==============================] - ETA: 0s - loss: 0.4460 - tversky: 0.
6571
Epoch 00002: val_loss improved from 0.87963 to 0.52528, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 10s 138ms/step - loss: 0.4460 - tver
sky: 0.6571 - val_loss: 0.5253 - val_tversky: 0.5755
Epoch 3/60
72/72 [==============================] - ETA: 0s - loss: 0.3664 - tversky: 0.
7360
Epoch 00003: val_loss improved from 0.52528 to 0.39022, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 10s 145ms/step - loss: 0.3664 - tver
sky: 0.7360 - val_loss: 0.3902 - val_tversky: 0.7139
Epoch 4/60
72/72 [==============================] - ETA: 0s - loss: 0.3364 - tversky: 0.
7641
Epoch 00004: val_loss improved from 0.39022 to 0.37156, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 9s 130ms/step - loss: 0.3364 - tvers
ky: 0.7641 - val_loss: 0.3716 - val_tversky: 0.7299
Epoch 5/60
72/72 [==============================] - ETA: 0s - loss: 0.3033 - tversky: 0.
7942
Epoch 00005: val_loss improved from 0.37156 to 0.33388, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 10s 134ms/step - loss: 0.3033 - tver
sky: 0.7942 - val_loss: 0.3339 - val_tversky: 0.7673
Epoch 6/60
72/72 [==============================] - ETA: 0s - loss: 0.2770 - tversky: 0.
8178
Epoch 00006: val_loss did not improve from 0.33388
72/72 [==============================] - 9s 123ms/step - loss: 0.2770 - tvers
ky: 0.8178 - val_loss: 0.3523 - val_tversky: 0.7505
Epoch 7/60
72/72 [==============================] - ETA: 0s - loss: 0.2593 - tversky: 0.
8334
Epoch 00007: val_loss improved from 0.33388 to 0.26869, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 10s 142ms/step - loss: 0.2593 - tver
sky: 0.8334 - val_loss: 0.2687 - val_tversky: 0.8235
Epoch 8/60
72/72 [==============================] - ETA: 0s - loss: 0.2466 - tversky: 0.
8429
Epoch 00008: val_loss did not improve from 0.26869
72/72 [==============================] - 8s 118ms/step - loss: 0.2466 - tvers
ky: 0.8429 - val_loss: 0.2786 - val_tversky: 0.8165
Epoch 9/60
```

```
72/72 [==============================] - ETA: 0s - loss: 0.2285 - tversky: 0.
8586
Epoch 00009: val_loss did not improve from 0.26869
72/72 [==============================] - 9s 129ms/step - loss: 0.2285 - tvers
ky: 0.8586 - val_loss: 0.2713 - val_tversky: 0.8224
Epoch 10/60
72/72 [==============================] - ETA: 0s - loss: 0.2221 - tversky: 0.
8641
Epoch 00010: val_loss did not improve from 0.26869
72/72 [==============================] - 10s 143ms/step - loss: 0.2221 - tver
sky: 0.8641 - val_loss: 0.3318 - val_tversky: 0.7660
Epoch 11/60
72/72 [==============================] - ETA: 0s - loss: 0.2236 - tversky: 0.
8629
Epoch 00011: val_loss improved from 0.26869 to 0.25092, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 9s 128ms/step - loss: 0.2236 - tvers
ky: 0.8629 - val_loss: 0.2509 - val_tversky: 0.8413
Epoch 12/60
72/72 [==============================] - ETA: 0s - loss: 0.2006 - tversky: 0.
8814
Epoch 00012: val_loss improved from 0.25092 to 0.25004, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 10s 134ms/step - loss: 0.2006 - tver
sky: 0.8814 - val_loss: 0.2500 - val_tversky: 0.8423
Epoch 13/60
72/72 [==============================] - ETA: 0s - loss: 0.1869 - tversky: 0.
8924
Epoch 00013: val_loss improved from 0.25004 to 0.22796, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 9s 124ms/step - loss: 0.1869 - tvers
ky: 0.8924 - val_loss: 0.2280 - val_tversky: 0.8601
Epoch 14/60
72/72 [==============================] - ETA: 0s - loss: 0.1755 - tversky: 0.
9010
Epoch 00014: val_loss improved from 0.22796 to 0.19868, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 11s 149ms/step - loss: 0.1755 - tver
sky: 0.9010 - val_loss: 0.1987 - val_tversky: 0.8835
Epoch 15/60
72/72 [==============================] - ETA: 0s - loss: 0.1672 - tversky: 0.
9072
Epoch 00015: val_loss did not improve from 0.19868
72/72 [==============================] - 9s 121ms/step - loss: 0.1672 - tvers
ky: 0.9072 - val_loss: 0.2138 - val_tversky: 0.8718
Epoch 16/60
72/72 [==============================] - ETA: 0s - loss: 0.1605 - tversky: 0.
9122
Epoch 00016: val_loss did not improve from 0.19868
72/72 [==============================] - 9s 123ms/step - loss: 0.1605 - tvers
ky: 0.9122 - val_loss: 0.2909 - val_tversky: 0.8068
Epoch 17/60
72/72 [==============================] - ETA: 0s - loss: 0.1548 - tversky: 0.
9163
Epoch 00017: val_loss did not improve from 0.19868
72/72 [==============================] - 10s 133ms/step - loss: 0.1548 - tver
```

```
sky: 0.9163 - val_loss: 0.2267 - val_tversky: 0.8606
Epoch 18/60
72/72 [==============================] - ETA: 0s - loss: 0.1463 - tversky: 0.
9225
Epoch 00018: val_loss did not improve from 0.19868
72/72 [==============================] - 9s 127ms/step - loss: 0.1463 - tvers
ky: 0.9225 - val_loss: 0.2058 - val_tversky: 0.8783
Epoch 19/60
72/72 [==============================] - ETA: 0s - loss: 0.1416 - tversky: 0.
9259
Epoch 00019: val_loss did not improve from 0.19868
72/72 [==============================] - 9s 121ms/step - loss: 0.1416 - tvers
ky: 0.9259 - val_loss: 0.2029 - val_tversky: 0.8801
Epoch 20/60
72/72 [==============================] - ETA: 0s - loss: 0.1420 - tversky: 0.
9255
Epoch 00020: val_loss improved from 0.19868 to 0.19234, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 10s 135ms/step - loss: 0.1420 - tver
sky: 0.9255 - val_loss: 0.1923 - val_tversky: 0.8885
Epoch 21/60
72/72 [==============================] - ETA: 0s - loss: 0.1357 - tversky: 0.
9300
Epoch 00021: val_loss improved from 0.19234 to 0.18578, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 11s 147ms/step - loss: 0.1357 - tver
sky: 0.9300 - val_loss: 0.1858 - val_tversky: 0.8938
Epoch 22/60
72/72 [==============================] - ETA: 0s - loss: 0.1329 - tversky: 0.
9318
Epoch 00022: val_loss did not improve from 0.18578
72/72 [==============================] - 9s 121ms/step - loss: 0.1329 - tvers
ky: 0.9318 - val_loss: 0.1936 - val_tversky: 0.8879
Epoch 23/60
72/72 [==============================] - ETA: 0s - loss: 0.1316 - tversky: 0.
9326
Epoch 00023: val_loss did not improve from 0.18578
72/72 [==============================] - 9s 119ms/step - loss: 0.1316 - tvers
ky: 0.9326 - val_loss: 0.1910 - val_tversky: 0.8893
Epoch 24/60
72/72 [==============================] - ETA: 0s - loss: 0.1241 - tversky: 0.
9378
Epoch 00024: val_loss improved from 0.18578 to 0.18191, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 10s 135ms/step - loss: 0.1241 - tver
sky: 0.9378 - val_loss: 0.1819 - val_tversky: 0.8966
Epoch 25/60
72/72 [==============================] - ETA: 0s - loss: 0.1265 - tversky: 0.
9363
Epoch 00025: val_loss improved from 0.18191 to 0.18061, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 11s 156ms/step - loss: 0.1265 - tver
sky: 0.9363 - val_loss: 0.1806 - val_tversky: 0.8977
Epoch 26/60
72/72 [==============================] - ETA: 0s - loss: 0.1218 - tversky: 0.
9393
```

```
Epoch 00026: val_loss did not improve from 0.18061
72/72 [==============================] - 9s 123ms/step - loss: 0.1218 - tvers
ky: 0.9393 - val_loss: 0.1840 - val_tversky: 0.8939
Epoch 27/60
72/72 [==============================] - ETA: 0s - loss: 0.1202 - tversky: 0.
9405
Epoch 00027: val_loss did not improve from 0.18061
72/72 [==============================] - 9s 129ms/step - loss: 0.1202 - tvers
ky: 0.9405 - val_loss: 0.1872 - val_tversky: 0.8922
Epoch 28/60
72/72 [==============================] - ETA: 0s - loss: 0.1169 - tversky: 0.
9426
Epoch 00028: val_loss did not improve from 0.18061
72/72 [==============================] - 10s 135ms/step - loss: 0.1169 - tver
sky: 0.9426 - val_loss: 0.1971 - val_tversky: 0.8845
Epoch 29/60
72/72 [==============================] - ETA: 0s - loss: 0.1128 - tversky: 0.
9452
Epoch 00029: val_loss improved from 0.18061 to 0.18038, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 10s 133ms/step - loss: 0.1128 - tver
sky: 0.9452 - val_loss: 0.1804 - val_tversky: 0.8972
Epoch 30/60
72/72 [==============================] - ETA: 0s - loss: 0.1110 - tversky: 0.
9464
Epoch 00030: val_loss did not improve from 0.18038
72/72 [==============================] - 9s 128ms/step - loss: 0.1110 - tvers
ky: 0.9464 - val_loss: 0.1991 - val_tversky: 0.8829
Epoch 31/60
72/72 [==============================] - ETA: 0s - loss: 0.1084 - tversky: 0.
9481
Epoch 00031: val_loss improved from 0.18038 to 0.17823, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 9s 126ms/step - loss: 0.1084 - tvers
ky: 0.9481 - val_loss: 0.1782 - val_tversky: 0.8993
Epoch 32/60
72/72 [==============================] - ETA: 0s - loss: 0.1078 - tversky: 0.
9485
Epoch 00032: val_loss did not improve from 0.17823
72/72 [==============================] - 10s 143ms/step - loss: 0.1078 - tver
sky: 0.9485 - val_loss: 0.1899 - val_tversky: 0.8902
Epoch 33/60
72/72 [==============================] - ETA: 0s - loss: 0.1075 - tversky: 0.
9486
Epoch 00033: val_loss did not improve from 0.17823
72/72 [==============================] - 8s 118ms/step - loss: 0.1075 - tvers
ky: 0.9486 - val_loss: 0.1824 - val_tversky: 0.8962
Epoch 34/60
72/72 [==============================] - ETA: 0s - loss: 0.1032 - tversky: 0.
9514
Epoch 00034: val_loss did not improve from 0.17823
72/72 [==============================] - 9s 129ms/step - loss: 0.1032 - tvers
ky: 0.9514 - val_loss: 0.2146 - val_tversky: 0.8711
Epoch 35/60
72/72 [==============================] - ETA: 0s - loss: 0.1032 - tversky: 0.
9514
```

```
Epoch 00035: val_loss did not improve from 0.17823
72/72 [==============================] - 9s 120ms/step - loss: 0.1032 - tvers
ky: 0.9514 - val_loss: 0.1857 - val_tversky: 0.8930
Epoch 36/60
72/72 [==============================] - ETA: 0s - loss: 0.1044 - tversky: 0.
9506
Epoch 00036: val_loss did not improve from 0.17823
72/72 [==============================] - 11s 148ms/step - loss: 0.1044 - tver
sky: 0.9506 - val_loss: 0.1791 - val_tversky: 0.8987
Epoch 37/60
72/72 [==============================] - ETA: 0s - loss: 0.1022 - tversky: 0.
9520
Epoch 00037: val_loss did not improve from 0.17823
72/72 [==============================] - 10s 132ms/step - loss: 0.1022 - tver
sky: 0.9520 - val_loss: 0.1913 - val_tversky: 0.8895
Epoch 38/60
72/72 [==============================] - ETA: 0s - loss: 0.0986 - tversky: 0.
9543
Epoch 00038: val_loss did not improve from 0.17823
72/72 [==============================] - 9s 126ms/step - loss: 0.0986 - tvers
ky: 0.9543 - val_loss: 0.1884 - val_tversky: 0.8914
Epoch 39/60
72/72 [==============================] - ETA: 0s - loss: 0.0960 - tversky: 0.
9558
Epoch 00039: val_loss did not improve from 0.17823
72/72 [==============================] - 10s 134ms/step - loss: 0.0960 - tver
sky: 0.9558 - val_loss: 0.1943 - val_tversky: 0.8869
Epoch 40/60
72/72 [==============================] - ETA: 0s - loss: 0.0927 - tversky: 0.
9579
Epoch 00040: val_loss improved from 0.17823 to 0.17746, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 10s 145ms/step - loss: 0.0927 - tver
sky: 0.9579 - val_loss: 0.1775 - val_tversky: 0.9002
Epoch 41/60
72/72 [==============================] - ETA: 0s - loss: 0.0954 - tversky: 0.
9562
Epoch 00041: val_loss did not improve from 0.17746
72/72 [==============================] - 9s 122ms/step - loss: 0.0954 - tvers
ky: 0.9562 - val_loss: 0.1892 - val_tversky: 0.8902
Epoch 42/60
72/72 [==============================] - ETA: 0s - loss: 0.0959 - tversky: 0.
9560
Epoch 00042: val_loss improved from 0.17746 to 0.17239, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 9s 126ms/step - loss: 0.0959 - tvers
ky: 0.9560 - val_loss: 0.1724 - val_tversky: 0.9038
Epoch 43/60
72/72 [==============================] - ETA: 0s - loss: 0.0914 - tversky: 0.
9587
Epoch 00043: val_loss did not improve from 0.17239
72/72 [==============================] - 10s 141ms/step - loss: 0.0914 - tver
sky: 0.9587 - val_loss: 0.1907 - val_tversky: 0.8900
Epoch 44/60
72/72 [==============================] - ETA: 0s - loss: 0.0892 - tversky: 0.
9600
```

```
Epoch 00044: val_loss did not improve from 0.17239
72/72 [==============================] - 10s 132ms/step - loss: 0.0892 - tver
sky: 0.9600 - val_loss: 0.1808 - val_tversky: 0.8974
Epoch 45/60
72/72 [==============================] - ETA: 0s - loss: 0.0877 - tversky: 0.
9610
Epoch 00045: val_loss did not improve from 0.17239
72/72 [==============================] - 9s 122ms/step - loss: 0.0877 - tvers
ky: 0.9610 - val_loss: 0.2140 - val_tversky: 0.8716
Epoch 46/60
72/72 [==============================] - ETA: 0s - loss: 0.0871 - tversky: 0.
9613
Epoch 00046: val_loss did not improve from 0.17239
72/72 [==============================] - 9s 127ms/step - loss: 0.0871 - tvers
ky: 0.9613 - val_loss: 0.1783 - val_tversky: 0.8990
Epoch 47/60
72/72 [==============================] - ETA: 0s - loss: 0.0864 - tversky: 0.
9617
Epoch 00047: val_loss did not improve from 0.17239
72/72 [==============================] - 11s 153ms/step - loss: 0.0864 - tver
sky: 0.9617 - val_loss: 0.1775 - val_tversky: 0.8994
Epoch 48/60
72/72 [==============================] - ETA: 0s - loss: 0.0846 - tversky: 0.
9628
Epoch 00048: val_loss did not improve from 0.17239
72/72 [==============================] - 9s 127ms/step - loss: 0.0846 - tvers
ky: 0.9628 - val_loss: 0.1840 - val_tversky: 0.8941
Epoch 49/60
72/72 [==============================] - ETA: 0s - loss: 0.0836 - tversky: 0.
9633
Epoch 00049: val_loss improved from 0.17239 to 0.17229, saving model to ResUN
et-segModel-weights.hdf5
72/72 [==============================] - 9s 128ms/step - loss: 0.0836 - tvers
ky: 0.9633 - val_loss: 0.1723 - val_tversky: 0.9040
Epoch 50/60
72/72 [==============================] - ETA: 0s - loss: 0.0846 - tversky: 0.
9627
Epoch 00050: val_loss did not improve from 0.17229
72/72 [==============================] - 9s 119ms/step - loss: 0.0846 - tvers
ky: 0.9627 - val_loss: 0.2094 - val_tversky: 0.8754
Epoch 51/60
72/72 [==============================] - ETA: 0s - loss: 0.0842 - tversky: 0.
9629
Epoch 00051: val_loss did not improve from 0.17229
72/72 [==============================] - 11s 158ms/step - loss: 0.0842 - tver
sky: 0.9629 - val_loss: 0.1838 - val_tversky: 0.8951
Epoch 52/60
72/72 [==============================] - ETA: 0s - loss: 0.0823 - tversky: 0.
9641
Epoch 00052: val_loss did not improve from 0.17229

Epoch 00052: ReduceLROnPlateau reducing learning rate to 0.01000000014901161
2.
72/72 [==============================] - 9s 121ms/step - loss: 0.0823 - tvers
ky: 0.9641 - val_loss: 0.1892 - val_tversky: 0.8912
Epoch 53/60
```

```
72/72 [==============================] - ETA: 0s - loss: 0.0782 - tversky: 0.
9665
Epoch 00053: val_loss did not improve from 0.17229
72/72 [==============================] - 9s 127ms/step - loss: 0.0782 - tvers
ky: 0.9665 - val_loss: 0.1944 - val_tversky: 0.8866
Epoch 54/60
72/72 [==============================] - ETA: 0s - loss: 0.0756 - tversky: 0.
9680
Epoch 00054: val_loss did not improve from 0.17229
72/72 [==============================] - 9s 123ms/step - loss: 0.0756 - tvers
ky: 0.9680 - val_loss: 0.1782 - val_tversky: 0.8994
Epoch 55/60
72/72 [==============================] - ETA: 0s - loss: 0.0734 - tversky: 0.
9692
Epoch 00055: val_loss did not improve from 0.17229
72/72 [==============================] - 9s 126ms/step - loss: 0.0734 - tvers
ky: 0.9692 - val_loss: 0.1854 - val_tversky: 0.8940
Epoch 56/60
72/72 [==============================] - ETA: 0s - loss: 0.0736 - tversky: 0.
9690
Epoch 00056: val_loss did not improve from 0.17229
72/72 [==============================] - 9s 121ms/step - loss: 0.0736 - tvers
ky: 0.9690 - val_loss: 0.1817 - val_tversky: 0.8967
Epoch 57/60
72/72 [==============================] - ETA: 0s - loss: 0.0733 - tversky: 0.
9692
Epoch 00057: val_loss did not improve from 0.17229
72/72 [==============================] - 9s 118ms/step - loss: 0.0733 - tvers
ky: 0.9692 - val_loss: 0.1860 - val_tversky: 0.8934
Epoch 58/60
72/72 [==============================] - ETA: 0s - loss: 0.0718 - tversky: 0.
9701
Epoch 00058: val_loss did not improve from 0.17229
72/72 [==============================] - 12s 161ms/step - loss: 0.0718 - tver
sky: 0.9701 - val_loss: 0.1822 - val_tversky: 0.8964
Epoch 59/60
72/72 [==============================] - ETA: 0s - loss: 0.0725 - tversky: 0.
9696
Epoch 00059: val_loss did not improve from 0.17229
72/72 [==============================] - 9s 121ms/step - loss: 0.0725 - tvers
ky: 0.9696 - val_loss: 0.1786 - val_tversky: 0.8988
Epoch 60/60
72/72 [==============================] - ETA: 0s - loss: 0.0716 - tversky: 0.
9701
Epoch 00060: val_loss did not improve from 0.17229
72/72 [==============================] - 9s 121ms/step - loss: 0.0716 - tvers
ky: 0.9701 - val_loss: 0.1788 - val_tversky: 0.8979
```

In [35]:

```
# saving model achitecture in json file
seg_model_json = seg_model.to_json()
with open("ResUNet-seg-model.json", "w") as json_file:
    json_file.write(seg_model_json)
```

10: SEGMENTATION MODEL EVALUATION

In [36]:

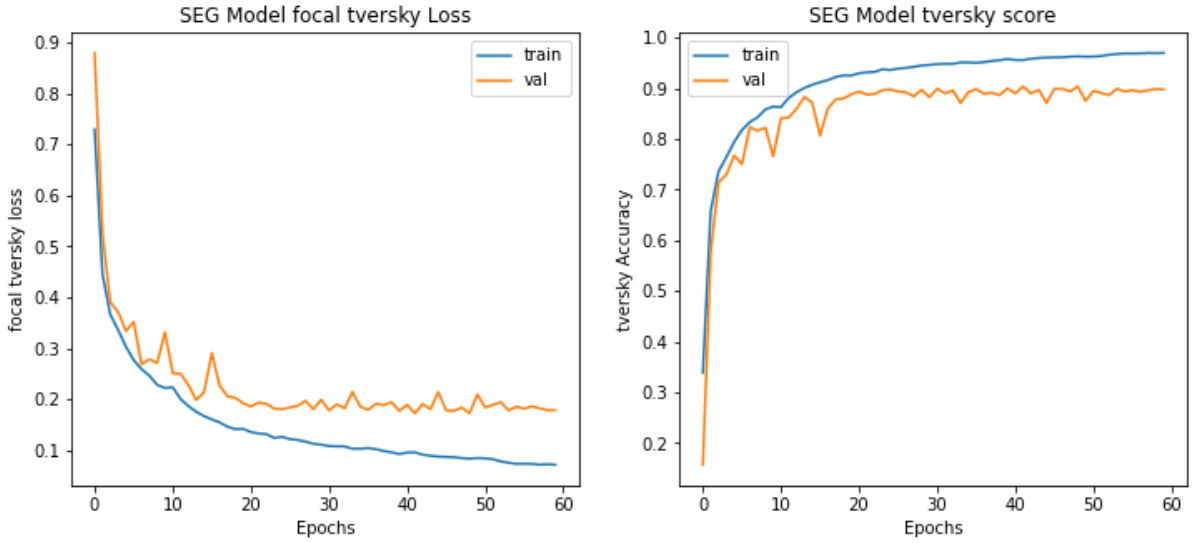
```
h.history.keys()
```

Out[36]:

```
dict_keys(['loss', 'tversky', 'val_loss', 'val_tversky', 'lr'])
```

In [37]:

```
plt.figure(figsize=(12,5))
plt.subplot(1,2,1)
plt.plot(h.history['loss']);
plt.plot(h.history['val_loss']);
plt.title("SEG Model focal tversky Loss");
plt.ylabel("focal tversky loss");
plt.xlabel("Epochs");
plt.legend(['train', 'val']);

plt.subplot(1,2,2)
plt.plot(h.history['tversky']);
plt.plot(h.history['val_tversky']);
plt.title("SEG Model tversky score");
plt.ylabel("tversky Accuracy");
plt.xlabel("Epochs");
plt.legend(['train', 'val']);
```



In [38]:

```
test_ids = list(X_test.image_path)
test_mask = list(X_test.mask_path)
test_data = DataGenerator(test_ids, test_mask)
_, tv = seg_model.evaluate(test_data)
print("Segmentation tversky is {:.2f}%".format(tv*100))
```

```
6/6 [==============================] - 1s 105ms/step - loss: 0.1886 - tversk
y: 0.8913
Segmentation tversky is 89.13%
```

11: SEGMENTATION MODEL PREFORMACE

(COMBINING CLASSIFICATION AND SEGMENTAION MODEL BUILDING PIPELINE)

In [39]:

```
def prediction(test, model, model_seg):
    '''
    Predcition function which takes dataframe containing ImageID as Input and perform 2 ty
pe of prediction on the image
    Initially, image is passed through the classification network which predicts whether t
he image has defect or not, if the model
    is 99% sure that the image has no defect, then the image is labeled as no-defect, if t
he model is not sure, it passes the image to the
    segmentation network, it again checks if the image has defect or not, if it has defec
t, then the type and location of defect is found
    '''
    # empty list to store results
    mask, image_id, has_mask = [], [], []

    #itetrating through each image in test data
    for i in test.image_path:

        img = io.imread(i)
        #normalizing
        img = img *1./255.
        #reshaping
        img = cv2.resize(img, (256,256))
        # converting img into array
        img = np.array(img, dtype=np.float64)
        #reshaping the image from 256,256,3 to 1,256,256,3
        img = np.reshape(img, (1,256,256,3))

        #making prediction for tumor in image
        is_defect = model.predict(img)

        #if tumour is not present we append the details of the image to the list
        if np.argmax(is_defect)==0:
            image_id.append(i)
            has_mask.append(0)
            mask.append('No mask :)')
            continue

        #Creating a empty array of shape 1,256,256,1
        X = np.empty((1,256,256,3))
        # read the image
        img = io.imread(i)
        #resizing the image and coverting them to array of type float64
        img = cv2.resize(img, (256,256))
        img = np.array(img, dtype=np.float64)

        # standardising the image
        img -= img.mean()
        img /= img.std()
        #converting the shape of image from 256,256,3 to 1,256,256,3
        X[0,] = img

        #make prediction of mask
        predict = model_seg.predict(X)
```

```
        # if sum of predicted mask is 0 then there is not tumour
        if predict.round().astype(int).sum()==0:
            image_id.append(i)
            has_mask.append(0)
            mask.append('No mask :)')
        else:
        #if the sum of pixel values are more than 0, then there is tumour
            image_id.append(i)
            has_mask.append(1)
            mask.append(predict)

    return pd.DataFrame({'image_path': image_id,'predicted_mask': mask,'has_mask': has_mas
k})
```

In [40]:

```
# making prediction
df_pred = prediction(test, model, seg_model)
df_pred
```

Out[40]:

	image_path	predicted_mask	has_mask
0	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	[[[[1.1940815e-06], [4.7767826e-06], [3.556578...	1
1	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	No mask :)	0
2	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	No mask :)	0
3	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	No mask :)	0
4	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	No mask :)	0
...	...	...	...
585	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	No mask :)	0
586	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	No mask :)	0
587	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	[[[[8.969336e-07], [4.048056e-06], [2.8011048e...	1
588	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	No mask :)	0
589	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	No mask :)	0

590 rows × 3 columns

In [41]:

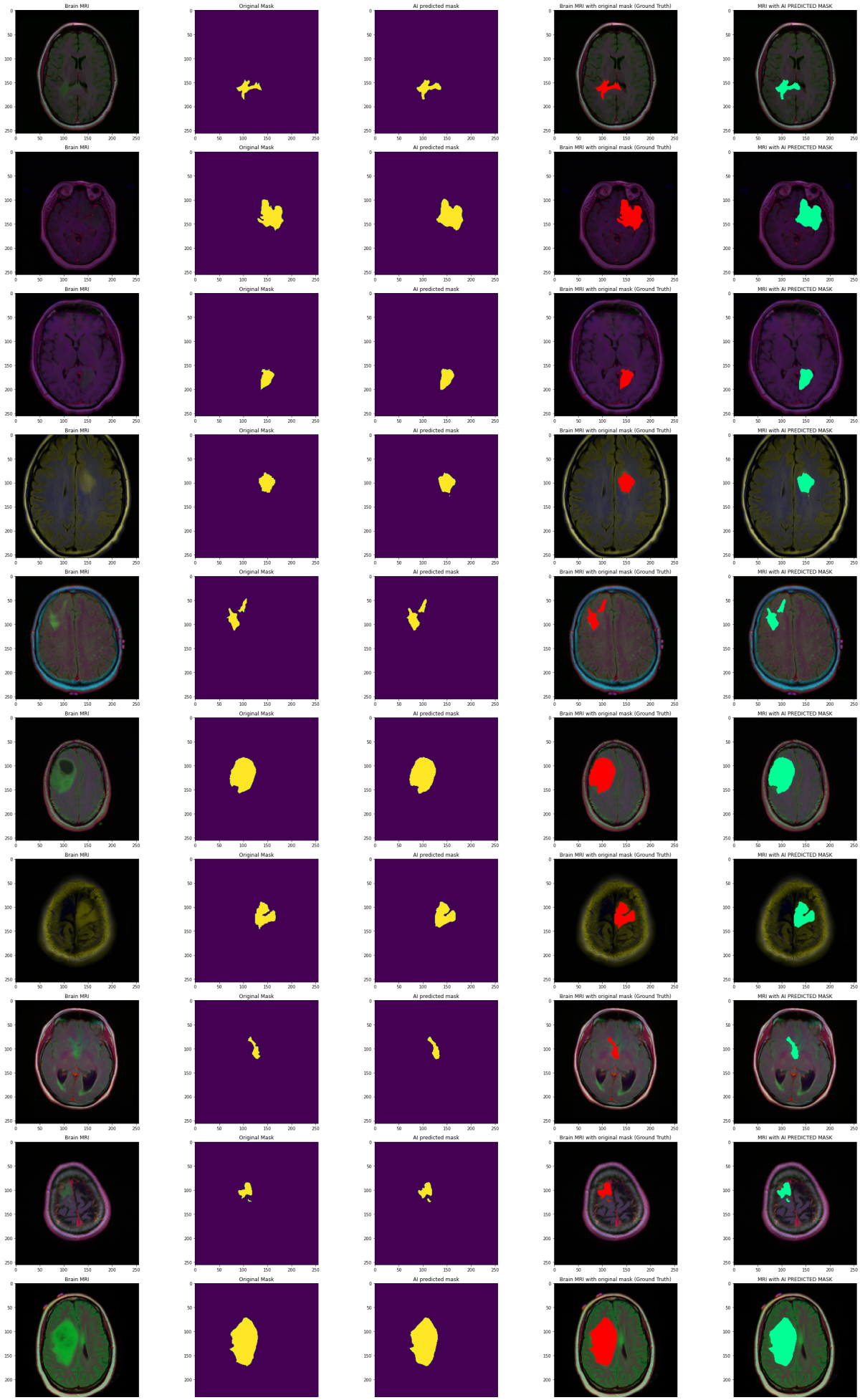
```
# merging original and prediction df
df_pred = test.merge(df_pred, on='image_path')
df_pred.head(10)
```

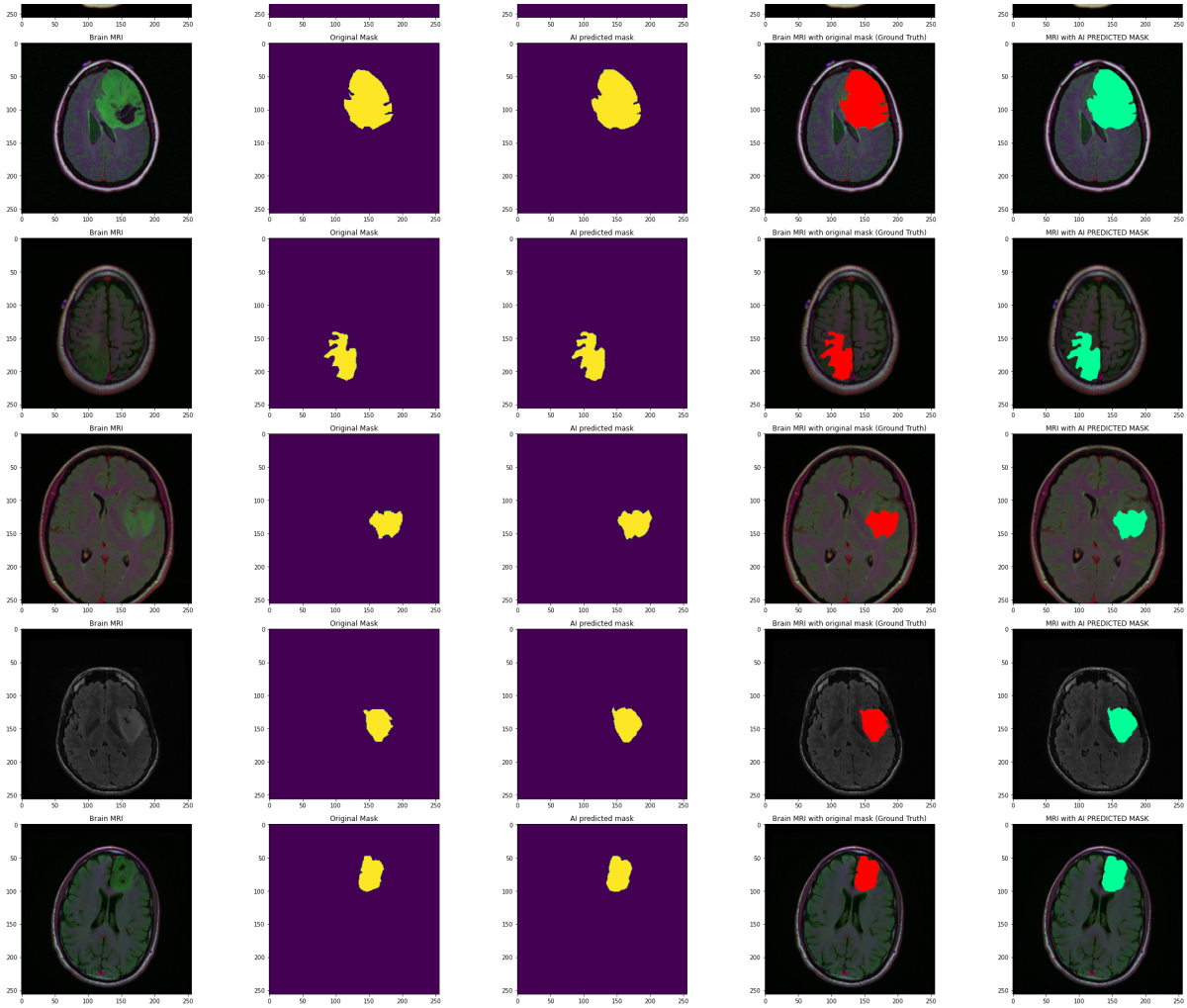
Out[41]:

	image_path	mask_path	mask	predicted_mask	has_mask
0	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	1	[[[[1.1940815e-06], [4.7767826e-06], [3.556578...	1
1	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	0	No mask :)	0
2	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	0	No mask :)	0
3	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	0	No mask :)	0
4	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	0	No mask :)	0
5	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	0	No mask :)	0
6	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	1	[[[[7.3062284e-07], [3.296093e-06], [2.9563191...	1
7	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	0	No mask :)	0
8	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	1	[[[[1.7690845e-06], [5.9955073e-06], [3.761689...	1
9	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	/kaggle/input/lgg-mri-segmentation/kaggle_3m/T...	0	No mask :)	0

In [42]:

```
#visualizing prediction
count = 0
fig, axs = plt.subplots(15,5, figsize=(30,70))

for i in range(len(df_pred)):
    if df_pred.has_mask[i]==1 and count<15:
        #read mri images
        img = io.imread(df_pred.image_path[i])
        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        axs[count][0].imshow(img)
        axs[count][0].title.set_text('Brain MRI')

        #read original mask
        mask = io.imread(df_pred.mask_path[i])
        axs[count][1].imshow(mask)
        axs[count][1].title.set_text('Original Mask')

        #read predicted mask
        pred = np.array(df_pred.predicted_mask[i]).squeeze().round()
        axs[count][2].imshow(pred)
        axs[count][2].title.set_text('AI predicted mask')

        #overlay original mask with MRI
        img[mask==255] = (255,0,0)
        axs[count][3].imshow(img)
        axs[count][3].title.set_text('Brain MRI with original mask (Ground Truth)')

        #overlay predicted mask and MRI
        img_ = io.imread(df_pred.image_path[i])
        img_ = cv2.cvtColor(img_, cv2.COLOR_BGR2RGB)
        img_[pred==1] = (0,255,150)
        axs[count][4].imshow(img_)
        axs[count][4].title.set_text('MRI with AI PREDICTED MASK')

        count +=1
    if (count==15):
        break

fig.tight_layout()
```


Brain MRI
Original Mask
AI predicted mask
Brain MRI with original mask (Ground Truth)
MRI with AI PREDICTED MASK



the predictions made by AI are almost correct :)