

Final term project

Team Name

Micky Kumar
Iram
Harold

Import

```
In [55]: import keras
from keras.models import Sequential
from keras.layers import Dense, Dropout, Flatten
from keras.layers import Conv2D, MaxPooling2D
from tensorflow.keras import regularizers
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.layers import Input, Conv2D, MaxPooling2D, GlobalMaxPooling2D,
Flatten, Dense, GlobalAveragePooling2D, Activation, MaxPool2D, AvgPool2D, Dropout, Conv1D, MaxPooling1D
from keras.models import Model
from keras import applications
import tensorflow as tf
from tqdm import tqdm
from sklearn.model_selection import train_test_split
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt
import seaborn as sns
import cv2
import glob
import os

from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OneHotEncoder

from keras.utils import to_categorical
from numpy import array
from sklearn.preprocessing import LabelBinarizer
from sklearn.preprocessing import OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn import preprocessing
from sklearn.utils import shuffle
```

data preparation

Project Overview

Based on the COVID-19 Cases dataset from open data toronto, this project will show detail about covid-19 by the Toronto public health. The details include demographic, geographic, and severity information for all confirmed and likely cases. The images for the X-ray is provided by kaggle. The Kaggle link is below: <https://www.kaggle.com/praveengovi/coronahack-chest-xraydataset/> (<https://www.kaggle.com/praveengovi/coronahack-chest-xraydataset/>)

Research question and problem framing

Based on this dataset, we are trying to predict the outcome based on age. This will allow people to know what age are getting more affected by COVID-19 and how to support them.

Who: The stakeholders for this project will be people creating a medicine for COVID-19, but also everyone in the community.

What: In this problem, we are trying to investigate about COVID-19 and the effect on it

When: This problem needs to be completed as soon as possible because this is affecting everyone's daily life and health.

Where: The COVID-19 is affecting worldwide and it is a worldwide pandemic.

Types of features

The dataset that is being used are:

- `_id`: This created a unique id for the cases that comes in.
- `Outbreak Associated`: This talks about where the outbreak is occurring in Toronto.
- `Age Group`: This feature is about age groups for COVID-19
- `Neighbourhood Name`: This feature talks about Toronto divided into 140 geographically distinct neighborhoods and can be used to understand where most cases are coming from
- `FSA`: This feature creates a unique code based on client postal code and help understand where the case is occurring.
- `Source of Infection`: The feature talks about before symptoms start are potential acquisition sources. this include travels, close contact with a case, institutional setting, healthcare setting, community, pending, unknown/missing, and N/A.
- `Classification`: The feature is about categorize the cases as confirmed according to standard criteria
- `Episode Date`: The feature talks about the date when the symptom occurred.
- `Reported Date`: The feature talks about date on which the case was reported to Toronto Public Health
- `Client Gender`: The feature that classifies people based on their assigned biological sex
- `Outcome`: This feature is include fatal (cases with a fatal outcome reported), resolved (case that is more than 14 days or the patient is recovered), Active (All other cases that is remaining)
- `Currently Hospitalized`: The feature talks about cases that are currently admitted to hospital
- `Currently in ICU`: The feature is about cases that are currently admitted to the intensive care unit with no discharge date.
- `Currently Intubated`: The feature is about cases that were intubated.
- `Ever Hospitalized`: The feature is about cases that were hospitalized.
- `Ever in ICU`: This feature is the cases that were admitted to the intensive care unit (ICU).
- `Ever Intubated`: This feature talks about cases that are currently intubated and that have been discharged.

In the `Chest_xray_Corona_Metadata.csv`, it contains 6 features and 5910 unique values. While the `Chest_xray_Corona_dataset_Summary.csv` has 5 feature and 7 values.

Explore

```
In [78]: covid19 = pd.read_csv('C:/Users/Micky/Downloads/COVID19-cases.csv')
train_df = pd.read_csv('C:/Users/Micky/Documents/GitHub/DL-final-project/Chest-xray/Chest_xray_Corona_Metadata.csv')
test_img_dir = 'C:/Users/Micky/Documents/GitHub/DL-final-project/Chest-xray/Coronahack-Chest-XRay-Dataset/Coronahack-Chest-XRay-Dataset/test'
train_img_dir = 'C:/Users/Micky/Documents/GitHub/DL-final-project/Chest-xray/Coronahack-Chest-XRay-Dataset/Coronahack-Chest-XRay-Dataset/test'
```

```
In [79]: covid19.shape
```

Out[79]: (14691, 17)

```
In [4]: covid19.head()
```

Out[4]:

	<code>_id</code>	<code>Outbreak Associated</code>	<code>Age Group</code>	<code>Neighbourhood Name</code>	<code>FSA</code>	<code>Source of Infection</code>	<code>Classification</code>	<code>Episode Date</code>	<code>Reported Date</code>	<code>Client Gender</code>	<code>Outcome</code>	<code>Currently Hospitalized</code>	<code>Currently in ICU</code>	<code>Currently Intubated</code>	<code>Ever Hospitalized</code>
0	14692	Sporadic	50-59	Malvern	M1B	Institutional	CONFIRMED	2020-03-25	2020-03-27	MALE	RESOLVED	No	No	No	No
1	14693	Sporadic	20-29	Malvern	M1B	Community	CONFIRMED	2020-03-20	2020-03-28	MALE	RESOLVED	No	No	No	Yes
2	14694	Sporadic	60-69	Malvern	M1B	Travel	CONFIRMED	2020-03-04	2020-03-08	FEMALE	RESOLVED	No	No	No	Yes
3	14695	Outbreak Associated	50-59	Rouge	M1B	N/A - Outbreak associated	CONFIRMED	2020-05-02	2020-05-04	FEMALE	RESOLVED	No	No	No	No
4	14696	Sporadic	30-39	Rouge	M1B	Close contact	CONFIRMED	2020-05-31	2020-06-06	FEMALE	RESOLVED	No	No	No	No

```
In [5]: covid19.tail()
```

Out[5]:

	<code>_id</code>	<code>Outbreak Associated</code>	<code>Age Group</code>	<code>Neighbourhood Name</code>	<code>FSA</code>	<code>Source of Infection</code>	<code>Classification</code>	<code>Episode Date</code>	<code>Reported Date</code>	<code>Client Gender</code>	<code>Outcome</code>	<code>Currently Hospitalized</code>	<code>Currently in ICU</code>	<code>Currently Intubated</code>	<code>Hospitalized</code>
14686	29378	Outbreak Associated	50-59	NaN	NaN	N/A - Outbreak associated	CONFIRMED	2020-06-02	2020-06-03	FEMALE	RESOLVED	No	No	No	
14687	29379	Outbreak Associated	50-59	NaN	NaN	N/A - Outbreak associated	CONFIRMED	2020-06-11	2020-06-15	FEMALE	RESOLVED	No	No	No	
14688	29380	Outbreak Associated	20-29	NaN	NaN	N/A - Outbreak associated	CONFIRMED	2020-05-09	2020-05-23	FEMALE	RESOLVED	No	No	No	
14689	29381	Outbreak Associated	40-49	NaN	NaN	N/A - Outbreak associated	CONFIRMED	2020-06-18	2020-06-19	FEMALE	RESOLVED	No	No	No	
14690	29382	Outbreak Associated	50-59	NaN	NaN	N/A - Outbreak associated	CONFIRMED	2020-06-22	2020-06-23	FEMALE	ACTIVE	Yes	No	No	

In [6]: covid19.describe()

Out[6]:

	_id
count	14691.000000
mean	22037.000000
std	4241.070737
min	14692.000000
25%	18364.500000
50%	22037.000000
75%	25709.500000
max	29382.000000

In [7]: covid19.head()

Out[7]:

	_id	Outbreak Associated	Age Group	Neighbourhood Name	FSA	Source of Infection	Classification	Episode Date	Reported Date	Client Gender	Outcome	Currently Hospitalized	Currently in ICU	Currently Intubated	Ever Hospitalized
0	14692	Sporadic	50-59	Malvern	M1B	Institutional	CONFIRMED	2020-03-25	2020-03-27	MALE	RESOLVED	No	No	No	N
1	14693	Sporadic	20-29	Malvern	M1B	Community	CONFIRMED	2020-03-20	2020-03-28	MALE	RESOLVED	No	No	No	Ye
2	14694	Sporadic	60-69	Malvern	M1B	Travel	CONFIRMED	2020-03-04	2020-03-08	FEMALE	RESOLVED	No	No	No	Ye
3	14695	Outbreak Associated	50-59	Rouge	M1B	N/A - Outbreak associated	CONFIRMED	2020-05-02	2020-05-04	FEMALE	RESOLVED	No	No	No	N
4	14696	Sporadic	30-39	Rouge	M1B	Close contact	CONFIRMED	2020-05-31	2020-06-06	FEMALE	RESOLVED	No	No	No	N

In [34]:

In [35]: train_df.head(5)

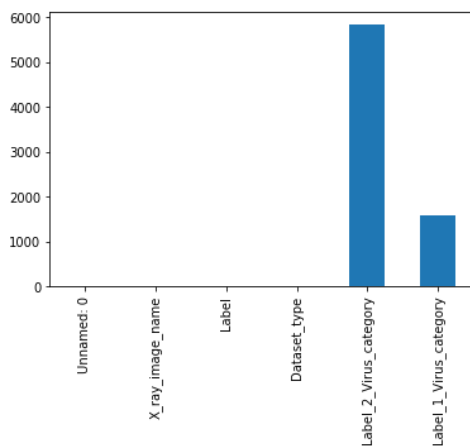
Out[35]:

	Unnamed: 0	X_ray_image_name	Label	Dataset_type	Label_2_Virus_category	Label_1_Virus_category
0	0	IM-0128-0001.jpeg	Normal	TRAIN	NaN	NaN
1	1	IM-0127-0001.jpeg	Normal	TRAIN	NaN	NaN
2	2	IM-0125-0001.jpeg	Normal	TRAIN	NaN	NaN
3	3	IM-0122-0001.jpeg	Normal	TRAIN	NaN	NaN
4	4	IM-0119-0001.jpeg	Normal	TRAIN	NaN	NaN

Missing Values

In [36]: missing_vals = train_df.isnull().sum()
missing_vals.plot(kind = 'bar')

Out[36]: <matplotlib.axes._subplots.AxesSubplot at 0x14cb4f1bbc8>



```
In [37]: train_df.dropna(how = 'all')
train_df.isnull().sum()
```

```
Out[37]: Unnamed: 0      0
X_ray_image_name      0
Label                 0
Dataset_type          0
Label_2_Virus_category 5841
Label_1_Virus_category 1576
dtype: int64
```

```
In [38]: train_data = train_df[train_df['Dataset_type'] == 'TRAIN']
test_data = train_df[train_df['Dataset_type'] == 'TEST']
assert train_data.shape[0] + test_data.shape[0] == train_df.shape[0]
print(f"Shape of train data : {train_data.shape}")
print(f"Shape of test data : {test_data.shape}")
test_data.sample(10)
```

Shape of train data : (5286, 6)
Shape of test data : (624, 6)

```
Out[38]:
```

	Unnamed: 0	X_ray_image_name	Label	Dataset_type	Label_2_Virus_category	Label_1_Virus_category
5484	5507	NORMAL2-IM-0079-0001.jpeg	Normal	TEST	NaN	NaN
5295	5318	IM-0010-0001.jpeg	Normal	TEST	NaN	NaN
5773	5796	person78_bacteria_386.jpeg	Pneumonia	TEST	NaN	bacteria
5761	5784	person81_bacteria_398.jpeg	Pneumonia	TEST	NaN	bacteria
5538	5561	person1613_virus_2799.jpeg	Pneumonia	TEST	NaN	Virus
5514	5537	NORMAL2-IM-0343-0001.jpeg	Normal	TEST	NaN	NaN
5444	5467	NORMAL2-IM-0221-0001.jpeg	Normal	TEST	NaN	NaN
5837	5860	person24_virus_58.jpeg	Pneumonia	TEST	NaN	Virus
5641	5664	person124_bacteria_589.jpeg	Pneumonia	TEST	NaN	bacteria
5879	5902	person1668_virus_2882.jpeg	Pneumonia	TEST	NaN	Virus

```
In [ ]: data_path = 'C:/UOFT/3546_TermProject/covid/DL-final-project/Chest_xray_seperate'
data_dir_list = os.listdir(data_path)
print(data_path)
```

Set Image Size and Epochs

```
In [ ]: img_rows=128
img_cols=128
num_channel=3
num_epoch=40
```

Define the number of classes

```
In [ ]: num_classes = 2
img_data_list=[]
```

```

In [ ]: def data_preperation():
    for dataset in data_dir_list:
        img_list=os.listdir(data_path+'/'+ dataset)
        print ('Loaded the images of dataset-'+'{ }\n'.format(dataset))
        for img in img_list:
            input_img=cv2.imread(data_path + '/' + dataset + '/' + img )
            if ( num_channel == 1):
                input_img=cv2.cvtColor(input_img, cv2.COLOR_BGR2GRAY)
            input_img_resize=cv2.resize(input_img,(128,128))
            img_data_list.append(input_img_resize)

def display_loss_accuracy(hist):
    train_loss=hist.history['loss']
    val_loss=hist.history['val_loss']
    train_acc=hist.history['accuracy']
    val_acc=hist.history['val_accuracy']
    xc=range(num_epoch)

    plt.figure(1,figsize=(7,5))
    plt.plot(xc,train_loss)
    plt.plot(xc,val_loss)
    plt.xlabel('num of Epochs')
    plt.ylabel('loss')
    plt.title('train_loss vs val_loss')
    plt.grid(True)
    plt.legend(['train', 'val'])
    #print plt.style.available # use bmh, classic,ggplot for big pictures
    plt.style.use(['classic'])

    plt.figure(2,figsize=(7,5))
    plt.plot(xc,train_acc)
    plt.plot(xc,val_acc)
    plt.xlabel('num of Epochs')
    plt.ylabel('accuracy')
    plt.title('train_acc vs val_acc')
    plt.grid(True)
    plt.legend(['train', 'val'],loc=4)
    #print plt.style.available # use bmh, classic,ggplot for big pictures
    plt.style.use(['classic'])

def get_featuremaps(model, layer_idx, X_batch):
    get_activations = K.function([model.layers[0].input, K.learning_phase()], [model.layers[layer_idx].output,])
    activations = get_activations([X_batch,0])
    return activations

def plot_featuremap_activations(activations):
    print (np.shape(activations))
    feature_maps = activations[0][0]
    print (np.shape(feature_maps))
    print (feature_maps.shape)

    fig=plt.figure(figsize=(16,16))
    plt.imshow(feature_maps[:, :, filter_num], cmap='gray')
    plt.savefig("featuremaps-layer-{}".format(layer_num) + "-filternum-{}".format(filter_num)+'.jpg')

    num_of_featuremaps=feature_maps.shape[2]
    fig=plt.figure(figsize=(16,16))
    plt.title("featuremaps-layer-{}".format(layer_num))
    subplot_num=int(np.ceil(np.sqrt(num_of_featuremaps)))
    for i in range(int(num_of_featuremaps)):
        ax = fig.add_subplot(subplot_num, subplot_num, i+1)
        #ax.imshow(output_image[0, :, i], interpolation='nearest' ) #to see the first filter
        ax.imshow(feature_maps[:, :, i], cmap='gray')
        plt.xticks([])
        plt.yticks([])
        plt.tight_layout()
    plt.show()
    fig.savefig("featuremaps-layer-{}".format(layer_num) + '.jpg')

# Plotting the confusion matrix
def plot_confusion_matrix(cm, classes,
                          normalize=False,
                          title='Confusion matrix',
                          cmap=plt.cm.Blues):
    """
    This function prints and plots the confusion matrix.
    Normalization can be applied by setting `normalize=True`.
    """
    plt.figure()
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)

    if normalize:
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]
        print("Normalized confusion matrix")
    else:

```

```

        print('Confusion matrix, without normalization')

    print(cm)

    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, cm[i, j],
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.tight_layout()
    plt.ylabel('True label')
    plt.xlabel('Predicted label')
    plt.show()

```

In [6]: data_preparation()

Loaded the images of dataset-NORMAL

Loaded the images of dataset-PNEUMONIA

In [7]: *# Calling Data Preperation*

```

img_data = np.array(img_data_list)
print(img_data.shape)
img_data = img_data.astype('float32')
img_data /= 255
print (img_data.shape)

# if num_channel==1:
#     img_data= np.expand_dims(img_data, axis=3)
#     print (img_data.shape)
# else:
#     if (K.image_data_format() == 'channels_first'):
#         img_data=np.rollaxis(img_data,3,1)

print (img_data.shape)

(5856, 128, 128, 3)
(5856, 128, 128, 3)
(5856, 128, 128, 3)

```

Assigning Labels

```

In [ ]: num_of_samples = img_data.shape[0]
labels = np.ones((num_of_samples,),dtype='int64')

labels[0:1582]=0
labels[1583:5856]=1

names = ['normal','pneumonia']

```

View Images

```

In [77]: assert os.path.isdir(test_img_dir) == True
assert os.path.isdir(train_img_dir) == True

sample_train_images = list(os.walk(train_img_dir))[0][2][:8]
sample_train_images = list(map(lambda x: os.path.join(train_img_dir, x), sample_train_images))

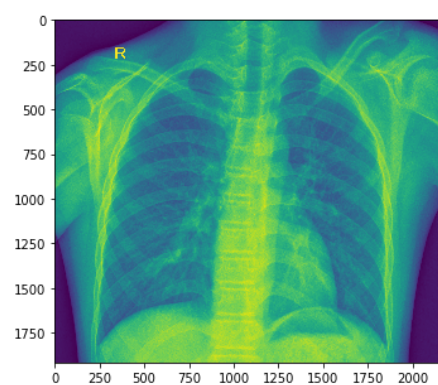
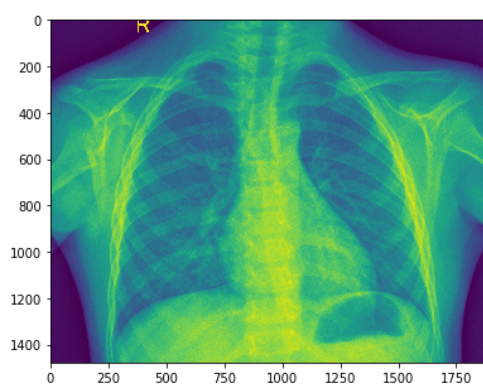
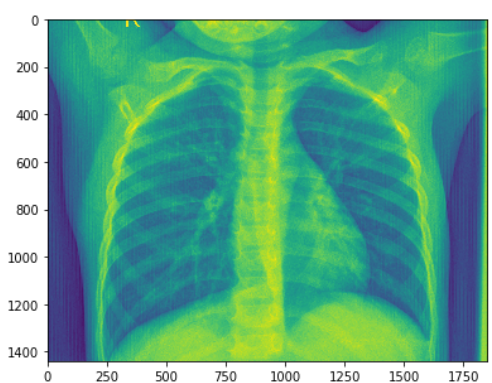
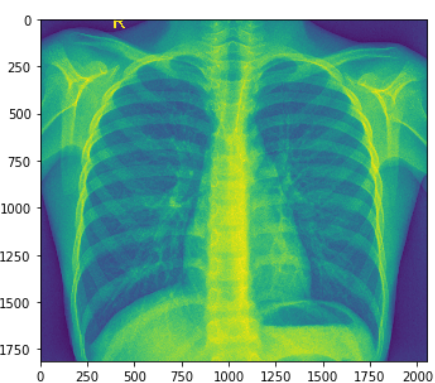
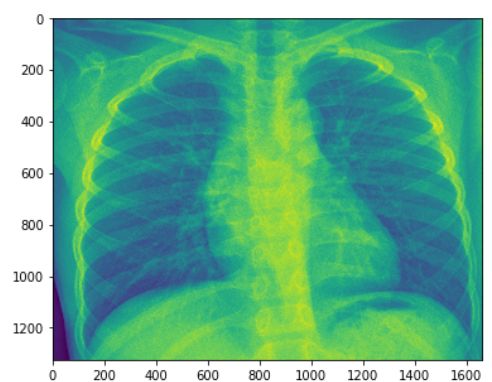
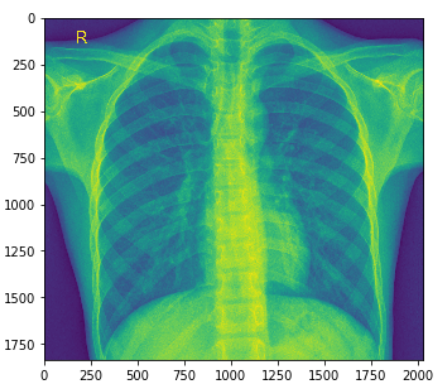
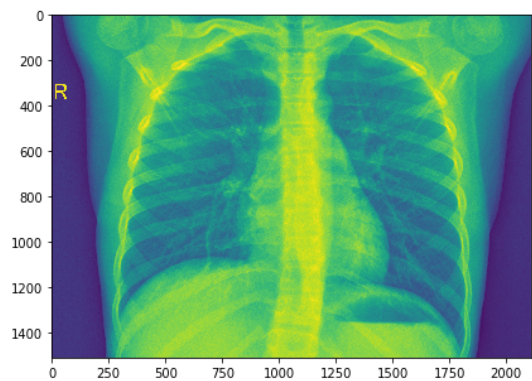
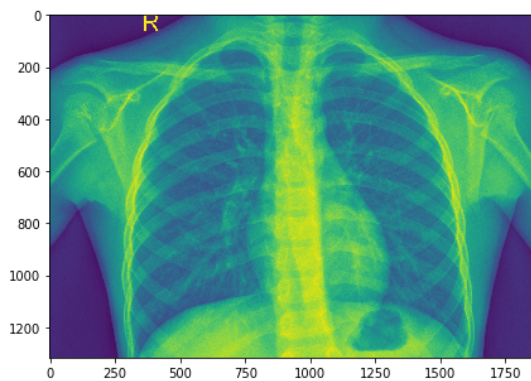
sample_test_images = list(os.walk(test_img_dir))[0][2][:8]
sample_test_images = list(map(lambda x: os.path.join(test_img_dir, x), sample_test_images))

```

View images from Train

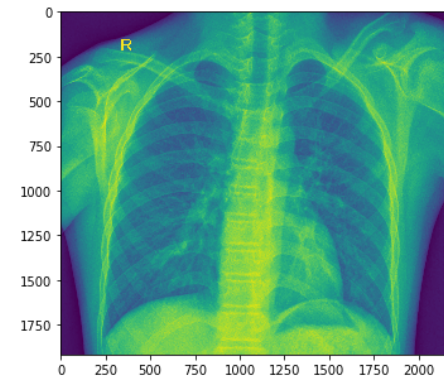
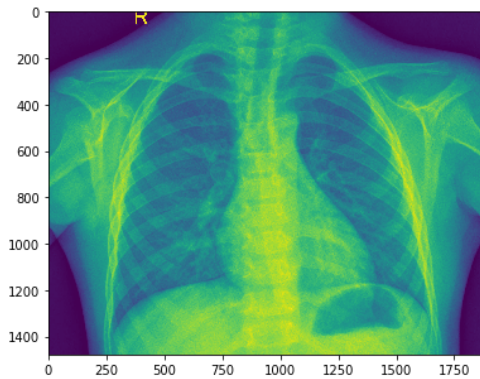
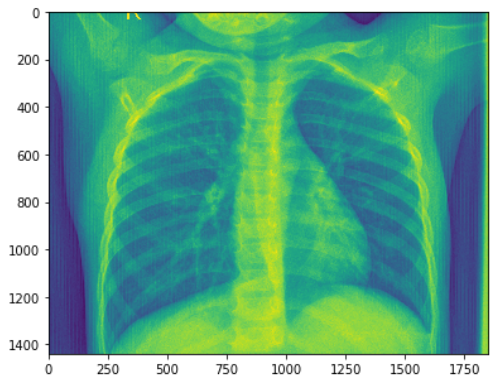
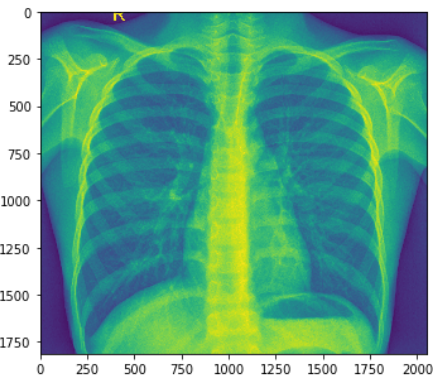
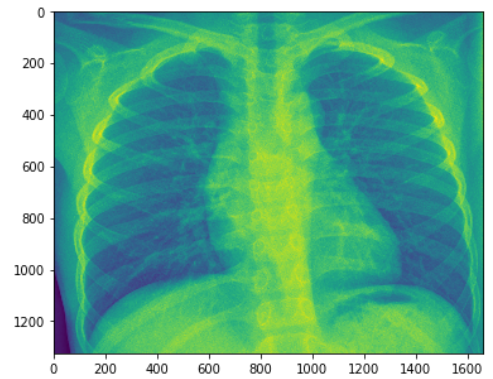
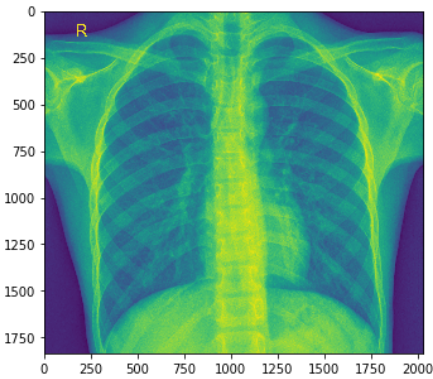
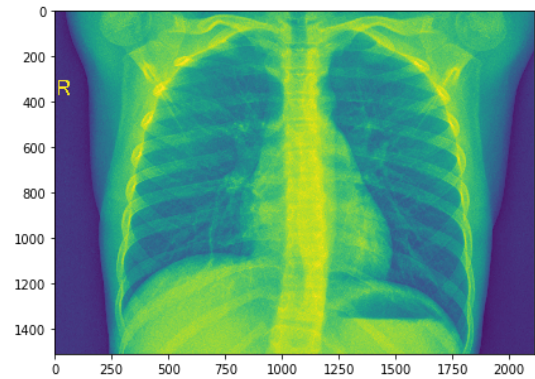
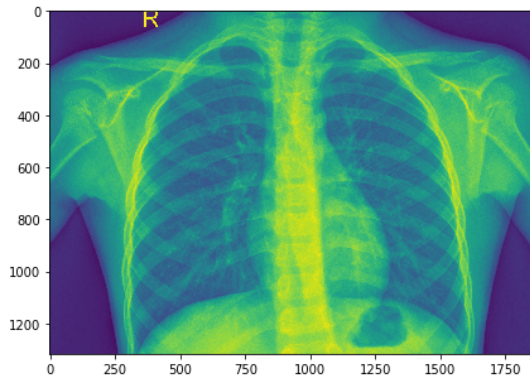
```
In [46]: from PIL import Image
plt.figure(figsize = (17,17))
for iterator, filename in enumerate(sample_train_images):
    image = Image.open(filename)
    plt.subplot(4,2,iterator+1)
    plt.imshow(image)

plt.tight_layout()
```



View images from Test


```
In [48]: plt.figure(figsize = (17,17))
for iterator, filename in enumerate(sample_test_images):
    image = Image.open(filename)
    plt.subplot(4,2,iterator+1)
    plt.imshow(image)
plt.tight_layout()
```



Use of proper training and test set


```

In [8]: # Drop all unwanted columns
# covid19 = covid19.drop('Episode Date',axis = 1)
# covid19 = covid19.drop('Reported Date',axis = 1)

# Create empty data frame
covid19_updated = pd.DataFrame()

# Apply Label Binarizer
lb = LabelBinarizer()
covid19['Outbreak Associated'] = lb.fit_transform(covid19['Outbreak Associated'])
covid19_updated = pd.concat([covid19_updated, covid19['Outbreak Associated']], axis=1)

dummies1 = pd.get_dummies(covid19["Age Group"], prefix='Age_Group')
covid19_updated = pd.concat([covid19_updated, dummies1], axis=1)

dummies2 = pd.get_dummies(covid19["Neighbourhood Name"], prefix='Neighbourhood_Name')
covid19_updated = pd.concat([covid19_updated, dummies2], axis=1)

fsa_dummies = pd.get_dummies(covid19.FSA, prefix='FSA')
covid19_updated = pd.concat([covid19_updated, fsa_dummies], axis=1)

dummies3 = pd.get_dummies(covid19["Source of Infection"], prefix='Source_of_Infection')
covid19_updated = pd.concat([covid19_updated, dummies3], axis=1)

covid19['Classification'] = lb.fit_transform(covid19['Classification'])
covid19_updated = pd.concat([covid19_updated, covid19['Classification']], axis=1)

dummies4 = pd.get_dummies(covid19["Client Gender"], prefix='Client_Gender')
covid19_updated = pd.concat([covid19_updated, dummies4], axis=1)

covid19['Currently Hospitalized'] = lb.fit_transform(covid19['Currently Hospitalized'])
covid19_updated = pd.concat([covid19_updated, covid19['Currently Hospitalized']], axis=1)

covid19['Currently in ICU'] = lb.fit_transform(covid19['Currently in ICU'])
covid19_updated = pd.concat([covid19_updated, covid19['Currently in ICU']], axis=1)

covid19['Currently Intubated'] = lb.fit_transform(covid19['Currently Intubated'])
covid19_updated = pd.concat([covid19_updated, covid19['Currently Intubated']], axis=1)

covid19['Ever Hospitalized'] = lb.fit_transform(covid19['Ever Hospitalized'])
covid19_updated = pd.concat([covid19_updated, covid19['Ever Hospitalized']], axis=1)

covid19['Ever in ICU'] = lb.fit_transform(covid19['Ever in ICU'])
covid19_updated = pd.concat([covid19_updated, covid19['Ever in ICU']], axis=1)

covid19['Ever Intubated'] = lb.fit_transform(covid19['Ever Intubated'])
covid19_updated = pd.concat([covid19_updated, covid19['Ever Intubated']], axis=1)

covid19_updated = pd.concat([covid19_updated, covid19['Outcome']], axis=1)

covid19_updated.Outcome = pd.factorize(covid19_updated.Outcome)[0]
covid19_updated.head()

# covid19_updated.to_csv("drive/My Drive/Final_Project/out.csv")

```

Out[8]:

	Outbreak Associated	Age_Group_19 and younger	Age_Group_20-29	Age_Group_30-39	Age_Group_40-49	Age_Group_50-59	Age_Group_60-69	Age_Group_70-79	Age_Group_80-89	Age_Group_90+	...
0	1	0	0	0	0	1	0	0	0	0	...
1	1	0	1	0	0	0	0	0	0	0	...
2	1	0	0	0	0	0	1	0	0	0	...
3	0	0	0	0	0	1	0	0	0	0	...
4	1	0	0	1	0	0	0	0	0	0	...

5 rows × 267 columns

```

In [9]: dataset = covid19_updated.values

X = dataset[:,0:261]
Y = dataset[:,261]

min_max_scaler = preprocessing.MinMaxScaler()
X_scale = min_max_scaler.fit_transform(X)

X_train, X_val_and_test, Y_train, Y_val_and_test = train_test_split(X_scale, Y, test_size=0.3)
X_val, X_test, Y_val, Y_test = train_test_split(X_val_and_test, Y_val_and_test, test_size=0.5)
print(X_train.shape, X_val.shape, X_test.shape, Y_train.shape, Y_val.shape, Y_test.shape)

(10283, 261) (2204, 261) (2204, 261) (10283,) (2204,) (2204,)

```

```
In [10]: model = Sequential([
    Dense(1000, activation='relu', kernel_regularizer=regularizers.l2(0.01), input_shape=(261,)),
    Dropout(0.3),
    Dense(1000, activation='relu', kernel_regularizer=regularizers.l2(0.01)),
    Dropout(0.3),
    Dense(1000, activation='relu', kernel_regularizer=regularizers.l2(0.01)),
    Dropout(0.3),
    Dense(1000, activation='relu', kernel_regularizer=regularizers.l2(0.01)),
    Dropout(0.3),
    Dense(1, activation='sigmoid', kernel_regularizer=regularizers.l2(0.01)),
])
```

```
In [11]: model.compile(optimizer='adam',
    loss='binary_crossentropy',
    metrics=['accuracy'])
```

```
hist_out = model.fit(X_train, Y_train,
                    batch_size=32, epochs=50,
                    validation_data=(X_val, Y_val))
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```
loss: 0.0201 - accuracy: 0.997 - ETA: 31s - loss: 0.0200 - accuracy: 0.997 - ETA: 30s - loss: 0.0200 - accuracy: 0.997 - ETA: 29s  
s - loss: 0.0199 - accuracy: 0.997 - ETA: 30s - loss: 0.0198 - accuracy: 0.997 - ETA: 29s - loss: 0.0197 - accuracy: 0.997 - ETA: 28s  
- loss: 0.0197 - accuracy: 0.997 - ETA: 29s - loss: 0.0196 - accuracy: 0.997 - ETA: 29s - loss: 0.0195 - accuracy: 0.997 - ETA: 28s  
- loss: 0.0195 - accuracy: 0.997 - ETA: 28s - loss: 0.0194 - accuracy: 0.997 - ETA: 27s - loss: 0.0193 - accuracy: 0.997 - ETA: 27s - lo  
ss: 0.0193 - accuracy: 0.997 - ETA: 27s - loss: 0.0192 - accuracy: 0.997 - ETA: 27s - loss: 0.0192 - accuracy: 0.997 - ETA: 26s - los  
s: 0.0191 - accuracy: 0.997 - ETA: 26s - loss: 0.0199 - accuracy: 0.997 - ETA: 25s - loss: 0.0198 - accuracy: 0.997 - ETA: 25s - loss:  
0.0197 - accuracy: 0.997 - ETA: 25s - loss: 0.0197 - accuracy: 0.997 - ETA: 24s - loss: 0.0196 - accuracy: 0.9975WARNING:tensorflow:Me  
thod (on_train_batch_end) is slow compared to the batch update (0.251071). Check your callbacks.  
267/322 [=====>.....] - ETA: 24s - loss: 0.0196 - accuracy: 0.9975WARNING:tensorflow:Method (on_train_batch_end) is  
slow compared to the batch update (0.251071). Check your callbacks.  
268/322 [=====>.....] - ETA: 23s - loss: 0.0195 - accuracy: 0.9976WARNING:tensorflow:Method (on_train_batch_end) is  
slow compared to the batch update (0.179530). Check your callbacks.  
322/322 [=====] - ETA: 23s - loss: 0.0203 - accuracy: 0.997 - ETA: 22s - loss: 0.0202 - accuracy: 0.997 - ET  
A: 22s - loss: 0.0201 - accuracy: 0.997 - ETA: 21s - loss: 0.0201 - accuracy: 0.997 - ETA: 21s - loss: 0.0200 - accuracy: 0.997 - ETA:  
20s - loss: 0.0200 - accuracy: 0.997 - ETA: 20s - loss: 0.0199 - accuracy: 0.997 - ETA: 19s - loss: 0.0198 - accuracy: 0.997 - ETA: 19  
s - loss: 0.0198 - accuracy: 0.997 - ETA: 19s - loss: 0.0197 - accuracy: 0.997 - ETA: 18s - loss: 0.0197 - accuracy: 0.997 - ETA: 18s  
- loss: 0.0196 - accuracy: 0.997 - ETA: 17s - loss: 0.0196 - accuracy: 0.997 - ETA: 17s - loss: 0.0201 - accuracy: 0.997 - ETA: 16s - lo  
ss: 0.0201 - accuracy: 0.997 - ETA: 16s - loss: 0.0200 - accuracy: 0.997 - ETA: 15s - loss: 0.0200 - accuracy: 0.997 - ETA: 15s - lo  
ss: 0.0199 - accuracy: 0.997 - ETA: 15s - loss: 0.0199 - accuracy: 0.997 - ETA: 14s - loss: 0.0198 - accuracy: 0.997 - ETA: 14s - los  
s: 0.0198 - accuracy: 0.997 - ETA: 13s - loss: 0.0197 - accuracy: 0.997 - ETA: 13s - loss: 0.0197 - accuracy: 0.997 - ETA: 13s - loss:  
0.0196 - accuracy: 0.997 - ETA: 12s - loss: 0.0196 - accuracy: 0.997 - ETA: 12s - loss: 0.0195 - accuracy: 0.997 - ETA: 12s - loss: 0.  
0195 - accuracy: 0.997 - ETA: 11s - loss: 0.0194 - accuracy: 0.997 - ETA: 11s - loss: 0.0194 - accuracy: 0.997 - ETA: 10s - loss: 0.01  
93 - accuracy: 0.997 - ETA: 10s - loss: 0.0205 - accuracy: 0.997 - ETA: 10s - loss: 0.0205 - accuracy: 0.997 - ETA: 9s - loss: 0.0204  
- accuracy: 0.997 - ETA: 9s - loss: 0.0204 - accuracy: 0.99 - ETA: 8s - loss: 0.0203 - accuracy: 0.99 - ETA: 8s - loss: 0.0203 - accu  
racy: 0.99 - ETA: 7s - loss: 0.0202 - accuracy: 0.99 - ETA: 7s - loss: 0.0202 - accuracy: 0.99 - ETA: 6s - loss: 0.0201 - accuracy: 0.  
99 - ETA: 6s - loss: 0.0201 - accuracy: 0.99 - ETA: 6s - loss: 0.0201 - accuracy: 0.99 - ETA: 5s - loss: 0.0200 - accuracy: 0.99 - ET  
A: 5s - loss: 0.0200 - accuracy: 0.99 - ETA: 4s - loss: 0.0199 - accuracy: 0.99 - ETA: 4s - loss: 0.0199 - accuracy: 0.99 - ETA: 3s -  
loss: 0.0198 - accuracy: 0.99 - ETA: 3s - loss: 0.0198 - accuracy: 0.99 - ETA: 2s - loss: 0.0197 - accuracy: 0.99 - ETA: 2s - loss:  
0.0197 - accuracy: 0.99 - ETA: 1s - loss: 0.0196 - accuracy: 0.99 - ETA: 1s - loss: 0.0196 - accuracy: 0.99 - ETA: 0s - loss: 0.0195  
- accuracy: 0.99 - ETA: 0s - loss: 0.0201 - accuracy: 0.99 - ETA: 0s - loss: 0.0201 - accuracy: 0.99 - 168s 520ms/step - loss: 0.0201  
- accuracy: 0.9975 - val_loss: 0.0233 - val_accuracy: 0.9968  
Epoch 15/50  
7/322 [.....] - ETA: 0s - loss: 0.0041 - accuracy: 1.00 - ETA: 58s - loss: 0.0043 - accuracy: 1.000 - ETA:  
1:46 - loss: 0.0044 - accuracy: 1.00 - ETA: 2:14 - loss: 0.0045 - accuracy: 1.00 - ETA: 2:25 - loss: 0.0045 - accuracy: 1.00 - ETA:  
2:32 - loss: 0.0045 - accuracy: 1.00 - ETA: 2:24 - loss: 0.0045 - accuracy: 1.0000WARNING:tensorflow:Method (on_train_batch_end) is s  
low compared to the batch update (0.174020). Check your callbacks.  
8/322 [.....] - ETA: 2:13 - loss: 0.0045 - accuracy: 1.0000WARNING:tensorflow:Method (on_train_batch_end) i  
s slow compared to the batch update (0.151026). Check your callbacks.  
82/322 [=====>.....] - ETA: 2:07 - loss: 0.0044 - accuracy: 1.00 - ETA: 1:59 - loss: 0.0239 - accuracy: 0.99 - ET  
A: 2:01 - loss: 0.0221 - accuracy: 0.99 - ETA: 2:04 - loss: 0.0206 - accuracy: 0.99 - ETA: 2:03 - loss: 0.0194 - accuracy: 0.99 - ETA:  
2:04 - loss: 0.0183 - accuracy: 0.99 - ETA: 2:07 - loss: 0.0174 - accuracy: 0.99 - ETA: 2:14 - loss: 0.0166 - accuracy: 0.99 - ETA: 2:  
21 - loss: 0.0266 - accuracy: 0.99 - ETA: 2:19 - loss: 0.0254 - accuracy: 0.99 - ETA: 2:22 - loss: 0.0243 - accuracy: 0.99 - ETA: 2:29  
- loss: 0.0234 - accuracy: 0.99 - ETA: 2:40 - loss: 0.0225 - accuracy: 0.99 - ETA: 2:43 - loss: 0.0217 - accuracy: 0.99 - ETA: 2:48 -  
loss: 0.0210 - accuracy: 0.99 - ETA: 2:51 - loss: 0.0203 - accuracy: 0.99 - ETA: 2:52 - loss: 0.0197 - accuracy: 0.99 - ETA: 2:51 - 1  
oss: 0.0191 - accuracy: 0.99 - ETA: 2:58 - loss: 0.0186 - accuracy: 0.99 - ETA: 3:01 - loss: 0.0181 - accuracy: 0.99 - ETA: 3:02 - los  
s: 0.0176 - accuracy: 0.99 - ETA: 3:01 - loss: 0.0172 - accuracy: 0.99 - ETA: 3:02 - loss: 0.0168 - accuracy: 0.99 - ETA: 3:03 - loss:  
0.0164 - accuracy: 0.99 - ETA: 3:02 - loss: 0.0160 - accuracy: 0.99 - ETA: 3:00 - loss: 0.0157 - accuracy: 0.99 - ETA: 3:06 - loss: 0.  
0153 - accuracy: 0.99 - ETA: 3:06 - loss: 0.0150 - accuracy: 0.99 - ETA: 3:05 - loss: 0.0147 - accuracy: 0.99 - ETA: 3:09 - loss: 0.01  
44 - accuracy: 0.99 - ETA: 3:08 - loss: 0.0141 - accuracy: 0.99 - ETA: 3:08 - loss: 0.0138 - accuracy: 0.99 - ETA: 3:11 - loss: 0.0136  
- accuracy: 0.99 - ETA: 3:13 - loss: 0.0133 - accuracy: 0.99 - ETA: 3:18 - loss: 0.0131 - accuracy: 0.99 - ETA: 3:21 - loss: 0.0128 -  
accuracy: 0.99 - ETA: 3:22 - loss: 0.0126 - accuracy: 0.99 - ETA: 3:23 - loss: 0.0167 - accuracy: 0.99 - ETA: 3:24 - loss: 0.0164 - a  
ccuracy: 0.99 - ETA: 3:24 - loss: 0.0161 - accuracy: 0.99 - ETA: 3:24 - loss: 0.0159 - accuracy: 0.99 - ETA: 3:26 - loss: 0.0156 - acc  
uracy: 0.99 - ETA: 3:31 - loss: 0.0154 - accuracy: 0.99 - ETA: 3:33 - loss: 0.0152 - accuracy: 0.99 - ETA: 3:37 - loss: 0.0149 - accur  
acy: 0.99 - ETA: 3:37 - loss: 0.0147 - accuracy: 0.99 - ETA: 3:37 - loss: 0.0145 - accuracy: 0.99 - ETA: 3:36 - loss: 0.0180 - accurac  
y: 0.99 - ETA: 3:34 - loss: 0.0177 - accuracy: 0.99 - ETA: 3:34 - loss: 0.0175 - accuracy: 0.99 - ETA: 3:37 - loss: 0.0173 - accuracy:  
0.99 - ETA: 3:38 - loss: 0.0170 - accuracy: 0.99 - ETA: 3:38 - loss: 0.0168 - accuracy: 0.99 - ETA: 3:36 - loss: 0.0166 - accuracy: 0.  
99 - ETA: 3:37 - loss: 0.0164 - accuracy: 0.99 - ETA: 3:37 - loss: 0.0162 - accuracy: 0.99 - ETA: 3:38 - loss: 0.0160 - accuracy: 0.99  
- ETA: 3:36 - loss: 0.0158 - accuracy: 0.99
```

```
oss: 0.0194 - accuracy: 0.99 - ETA: 1:46 - loss: 0.0193 - accuracy: 0.99 - ETA: 1:45 - loss: 0.0192 - accuracy: 0.99 - ETA: 1:44 - los  
s: 0.0192 - accuracy: 0.99 - ETA: 1:43 - loss: 0.0191 - accuracy: 0.99 - ETA: 1:42 - loss: 0.0190 - accuracy: 0.99 - ETA: 1:41 - loss:  
0.0189 - accuracy: 0.99 - ETA: 1:40 - loss: 0.0188 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0187 - accuracy: 0.99 - ETA: 1:38 - loss: 0.  
0186 - accuracy: 0.99 - ETA: 1:37 - loss: 0.0185 - accuracy: 0.99 - ETA: 1:36 - loss: 0.0185 - accuracy: 0.99 - ETA: 1:36 - loss: 0.01  
95 - accuracy: 0.99 - ETA: 1:35 - loss: 0.0194 - accuracy: 0.99 - ETA: 1:34 - loss: 0.0193 - accuracy: 0.99 - ETA: 1:33 - loss: 0.0192  
- accuracy: 0.99 - ETA: 1:32 - loss: 0.0191 - accuracy: 0.99 - ETA: 1:32 - loss: 0.0190 - accuracy: 0.99 - ETA: 1:31 - loss: 0.0190 -  
accuracy: 0.99 - ETA: 1:30 - loss: 0.0189 - accuracy: 0.99 - ETA: 1:29 - loss: 0.0188 - accuracy: 0.99 - ETA: 1:28 - loss: 0.0187 - a  
ccuracy: 0.99 - ETA: 1:27 - loss: 0.0186 - accuracy: 0.99 - ETA: 1:26 - loss: 0.0185 - accuracy: 0.99 - ETA: 1:25 - loss: 0.0185 - acc  
uracy: 0.99 - ETA: 1:24 - loss: 0.0184 - accuracy: 0.99 - ETA: 1:24 - loss: 0.0183 - accuracy: 0.99 - ETA: 1:23 - loss: 0.0182 - accur  
acy: 0.99 - ETA: 1:22 - loss: 0.0181 - accuracy: 0.99 - ETA: 1:21 - loss: 0.0181 - accuracy: 0.99 - ETA: 1:20 - loss: 0.0180 - accurac  
y: 0.99 - ETA: 1:19 - loss: 0.0179 - accuracy: 0.99 - ETA: 1:18 - loss: 0.0178 - accuracy: 0.99 - ETA: 1:18 - loss: 0.0178 - accuracy:  
0.99 - ETA: 1:17 - loss: 0.0177 - accuracy: 0.99 - ETA: 1:16 - loss: 0.0176 - accuracy: 0.99 - ETA: 1:15 - loss: 0.0175 - accuracy: 0.  
99 - ETA: 1:14 - loss: 0.0175 - accuracy: 0.99 - ETA: 1:13 - loss: 0.0174 - accuracy: 0.99 - ETA: 1:12 - loss: 0.0173 - accuracy: 0.99  
ETA: 1:12 - loss: 0.0172 - accuracy: 0.99 - ETA: 1:11 - loss: 0.0172 - accuracy: 0.99 - ETA: 1:10 - loss: 0.0171 - accuracy: 0.99 - E  
TA: 1:09 - loss: 0.0170 - accuracy: 0.99 - ETA: 1:08 - loss: 0.0170 - accuracy: 0.99 - ETA: 1:07 - loss: 0.0169 - accuracy: 0.99 - ET  
TA: 1:07 - loss: 0.0168 - accuracy: 0.99 - ETA: 1:06 - loss: 0.0168 - accuracy: 0.99 - ETA: 1:05 - loss: 0.0167 - accuracy: 0.99 - ET  
A: 1:04 - loss: 0.0166 - accuracy: 0.99 - ETA: 1:03 - loss: 0.0166 - accuracy: 0.99 - ETA: 1:03 - loss: 0.0165 - accuracy: 0.99 - ETA: 1:  
1:02 - loss: 0.0164 - accuracy: 0.99 - ETA: 1:01 - loss: 0.0164 - accuracy: 0.99 - ETA: 1:00 - loss: 0.0163 - accuracy: 0.99 - ETA: 1:  
00 - loss: 0.0162 - accuracy: 0.99 - ETA: 59s - loss: 0.0162 - accuracy: 0.9980 - ETA: 58s - loss: 0.0161 - accuracy: 0.998 - ETA: 58s  
- loss: 0.0161 - accuracy: 0.998 - ETA: 57s - loss: 0.0160 - accuracy: 0.998 - ETA: 56s - loss: 0.0159 - accuracy: 0.998 - ETA: 56s -  
loss: 0.0159 - accuracy: 0.998 - ETA: 55s - loss: 0.0158 - accuracy: 0.998 - ETA: 54s - loss: 0.0157 - accuracy: 0.998 - ETA: 54s - l  
oss: 0.0157 - accuracy: 0.998 - ETA: 53s - loss: 0.0156 - accuracy: 0.998 - ETA: 52s - loss: 0.0166 - accuracy: 0.998 - ETA: 51s - los  
s: 0.0165 - accuracy: 0.998 - ETA: 51s - loss: 0.0165 - accuracy: 0.998 - ETA: 50s - loss: 0.0164 - accuracy: 0.998 - ETA: 50s - loss:  
0.0164 - accuracy: 0.998 - ETA: 49s - loss: 0.0163 - accuracy: 0.998 - ETA: 48s - loss: 0.0163 - accuracy: 0.998 - ETA: 48s - loss: 0.  
0162 - accuracy: 0.998 - ETA: 47s - loss: 0.0161 - accuracy: 0.998 - ETA: 46s - loss: 0.0161 - accuracy: 0.998 - ETA: 46s - loss: 0.01  
60 - accuracy: 0.998 - ETA: 45s - loss: 0.0160 - accuracy: 0.998 - ETA: 45s - loss: 0.0159 - accuracy: 0.998 - ETA: 44s - loss: 0.0168  
- accuracy: 0.997 - ETA: 43s - loss: 0.0167 - accuracy: 0.998 - ETA: 43s - loss: 0.0167 - accuracy: 0.998 - ETA: 42s - loss: 0.0166 -  
accuracy: 0.998 - ETA: 42s - loss: 0.0165 - accuracy: 0.998 - ETA: 41s - loss: 0.0165 - accuracy: 0.998 - ETA: 41s - loss: 0.0164 - a  
ccuracy: 0.998 - ETA: 41s - loss: 0.0164 - accuracy: 0.998 - ETA: 40s - loss: 0.0163 - accuracy: 0.998 - ETA: 40s - loss: 0.0163 - acc  
uracy: 0.998 - ETA: 39s - loss: 0.0162 - accuracy: 0.998 - ETA: 39s - loss: 0.0162 - accuracy: 0.998 - ETA: 38s - loss: 0.0161 - accur  
acy: 0.998 - ETA: 38s - loss: 0.0161 - accuracy: 0.998 - ETA: 37s - loss: 0.0160 - accuracy: 0.998 - ETA: 37s - loss: 0.0160 - accurac  
y: 0.998 - ETA: 36s - loss: 0.0159 - accuracy: 0.998 - ETA: 35s - loss: 0.0159 - accuracy: 0.998 - ETA: 35s - loss: 0.0158 - accuracy:  
0.998 - ETA: 34s - loss: 0.0158 - accuracy: 0.998 - ETA: 33s - loss: 0.0158 - accuracy: 0.998 - ETA: 33s - loss: 0.0157 - accuracy: 0.  
998 - ETA: 32s - loss: 0.0163 - accuracy: 0.998 - ETA: 32s - loss: 0.0163 - accuracy: 0.998 - ETA: 31s - loss: 0.0162 - accuracy: 0.99  
8 - ETA: 30s - loss: 0.0162 - accuracy: 0.9980297322 [=====>...] - ETA: 30s - loss: 0.0161 - accuracy: 0.998 - E  
TA: 29s - loss: 0.0161 - accuracy: 0.998 - ETA: 29s - loss: 0.0161 - accuracy: 0.998 - ETA: 28s - loss: 0.0160 - accuracy: 0.998 - ET  
A: 27s - loss: 0.0160 - accuracy: 0.998 - ETA: 27s - loss: 0.0159 - accuracy: 0.998 - ETA: 26s - loss: 0.0159 - accuracy: 0.998 - ETA:  
26s - loss: 0.0158 - accuracy: 0.998 - ETA: 25s - loss: 0.0158 - accuracy: 0.998 - ETA: 24s - loss: 0.0157 - accuracy: 0.998 - ETA: 24  
s - loss: 0.0157 - accuracy: 0.998 - ETA: 23s - loss: 0.0156 - accuracy: 0.998 - ETA: 23s - loss: 0.0156 - accuracy: 0.998 - ETA: 22s  
- loss: 0.0156 - accuracy: 0.998 - ETA: 21s - loss: 0.0162 - accuracy: 0.998 - ETA: 21s - loss: 0.0162 - accuracy: 0.998 - ETA: 20s -  
loss: 0.0161 - accuracy: 0.998 - ETA: 20s - loss: 0.0161 - accuracy: 0.998 - ETA: 19s - loss: 0.0161 - accuracy: 0.998 - ETA: 19s - lo  
ss: 0.0160 - accuracy: 0.998 - ETA: 18s - loss: 0.0160 - accuracy: 0.998 - ETA: 18s - loss: 0.0159 - accuracy: 0.998 - ETA: 17s - los  
s: 0.0159 - accuracy: 0.998 - ETA: 17s - loss: 0.0158 - accuracy: 0.998 - ETA: 16s - loss: 0.0158 - accuracy: 0.998 - ETA: 15s - loss:  
0.0164 - accuracy: 0.998 - ETA: 15s - loss: 0.0163 - accuracy: 0.998 - ETA: 14s - loss: 0.0163 - accuracy: 0.998 - ETA: 14s - loss: 0.  
0163 - accuracy: 0.9980WARNING:tensorflow:Method (on_train_batch_end) is slow compared to the batch update (0.315580). Check your call  
backs.
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```
0.997 - ETA: 51s - loss: 0.0199 - accuracy: 0.997 - ETA: 50s - loss: 0.0198 - accuracy: 0.997 - ETA: 50s - loss: 0.0208  
- accuracy: 0.997 - ETA: 50s - loss: 0.0207 - accuracy: 0.997 - ETA: 49s - loss: 0.0206 - accuracy: 0.997 - ETA: 49s - loss: 0.0205 -  
- accuracy: 0.997 - ETA: 49s - loss: 0.0204 - accuracy: 0.997 - ETA: 48s - loss: 0.0204 - accuracy: 0.997 - ETA: 48s - loss: 0.0203 - a  
ccuracy: 0.997 - ETA: 47s - loss: 0.0202 - accuracy: 0.997 - ETA: 47s - loss: 0.0201 - accuracy: 0.997 - ETA: 47s - loss: 0.0200 - acc  
uracy: 0.997 - ETA: 47s - loss: 0.0199 - accuracy: 0.997 - ETA: 46s - loss: 0.0198 - accuracy: 0.997 - ETA: 46s - loss: 0.0198 - accur  
acy: 0.997 - ETA: 45s - loss: 0.0197 - accuracy: 0.997 - ETA: 45s - loss: 0.0196 - accuracy: 0.997 - ETA: 44s - loss: 0.0195 - accurac  
y: 0.997 - ETA: 44s - loss: 0.0194 - accuracy: 0.997 - ETA: 44s - loss: 0.0194 - accuracy: 0.997 - ETA: 43s - loss: 0.0193 - accuracy:  
0.997 - ETA: 43s - loss: 0.0192 - accuracy: 0.997 - ETA: 42s - loss: 0.0191 - accuracy: 0.997 - ETA: 42s - loss: 0.0200 - accuracy: 0.  
997 - ETA: 41s - loss: 0.0200 - accuracy: 0.997 - ETA: 41s - loss: 0.0199 - accuracy: 0.997 - ETA: 41s - loss: 0.0198 - accuracy: 0.99  
7 - ETA: 40s - loss: 0.0197 - accuracy: 0.997 - ETA: 40s - loss: 0.0197 - accuracy: 0.997 - ETA: 39s - loss: 0.0196 - accuracy: 0.997  
- ETA: 39s - loss: 0.0195 - accuracy: 0.997 - ETA: 38s - loss: 0.0203 - accuracy: 0.997 - ETA: 38s - loss: 0.0202 - accuracy: 0.997 -  
ETA: 38s - loss: 0.0201 - accuracy: 0.997 - ETA: 37s - loss: 0.0201 - accuracy: 0.997 - ETA: 37s - loss: 0.0200 - accuracy: 0.997 - ET  
A: 36s - loss: 0.0199 - accuracy: 0.997 - ETA: 36s - loss: 0.0199 - accuracy: 0.997 - ETA: 36s - loss: 0.0198 - accuracy: 0.997 - ETA:  
35s - loss: 0.0197 - accuracy: 0.997 - ETA: 35s - loss: 0.0197 - accuracy: 0.997 - ETA: 34s - loss: 0.0196 - accuracy: 0.997 - ETA: 34  
s - loss: 0.0195 - accuracy: 0.997 - ETA: 33s - loss: 0.0195 - accuracy: 0.997 - ETA: 33s - loss: 0.0194 - accuracy: 0.997 - ETA: 33s  
- loss: 0.0193 - accuracy: 0.997 - ETA: 32s - loss: 0.0193 - accuracy: 0.997 - ETA: 32s - loss: 0.0200 - accuracy: 0.997 - ETA: 32s -  
loss: 0.0200 - accuracy: 0.997 - ETA: 32s - loss: 0.0199 - accuracy: 0.997 - ETA: 31s - loss: 0.0198 - accuracy: 0.997 - ETA: 31s - lo  
ss: 0.0198 - accuracy: 0.997 - ETA: 31s - loss: 0.0197 - accuracy: 0.997 - ETA: 30s - loss: 0.0197 - accuracy: 0.997 - ETA: 30s - los  
s: 0.0196 - accuracy: 0.997 - ETA: 29s - loss: 0.0195 - accuracy: 0.997 - ETA: 29s - loss: 0.0195 - accuracy: 0.997 - ETA: 29s - loss:  
0.0194 - accuracy: 0.997 - ETA: 28s - loss: 0.0193 - accuracy: 0.997 - ETA: 28s - loss: 0.0200 - accuracy: 0.997 - ETA: 27s - loss: 0.  
0199 - accuracy: 0.997 - ETA: 27s - loss: 0.0199 - accuracy: 0.997 - ETA: 27s - loss: 0.0198 - accuracy: 0.997 - ETA: 26s - loss: 0.01  
98 - accuracy: 0.997 - ETA: 26s - loss: 0.0197 - accuracy: 0.997 - ETA: 25s - loss: 0.0196 - accuracy: 0.997 - ETA: 25s - loss: 0.0196  
- accuracy: 0.997 - ETA: 25s - loss: 0.0203 - accuracy: 0.997 - ETA: 24s - loss: 0.0209 - accuracy: 0.997 - ETA: 24s - loss: 0.0208 -  
accuracy: 0.997 - ETA: 23s - loss: 0.0208 - accuracy: 0.997 - ETA: 23s - loss: 0.0207 - accuracy: 0.997 - ETA: 22s - loss: 0.0213 - a  
ccuracy: 0.997 - ETA: 22s - loss: 0.0212 - accuracy: 0.997 - ETA: 22s - loss: 0.0212 - accuracy: 0.997 - ETA: 21s - loss: 0.0211 - acc  
uracy: 0.997 - ETA: 21s - loss: 0.0211 - accuracy: 0.997 - ETA: 21s - loss: 0.0210 - accuracy: 0.997 - ETA: 20s - loss: 0.0210 - accur  
acy: 0.997 - ETA: 20s - loss: 0.0210 - accuracy: 0.997 - ETA: 19s - loss: 0.0209 - accuracy: 0.997 - ETA: 19s - loss: 0.0209 - accurac  
y: 0.997 - ETA: 19s - loss: 0.0208 - accuracy: 0.997 - ETA: 18s - loss: 0.0214 - accuracy: 0.997 - ETA: 18s - loss: 0.0213 - accuracy:  
0.997 - ETA: 18s - loss: 0.0213 - accuracy: 0.997 - ETA: 17s - loss: 0.0212 - accuracy: 0.997 - ETA: 17s - loss: 0.0212 - accuracy: 0.  
997 - ETA: 16s - loss: 0.0211 - accuracy: 0.997 - ETA: 16s - loss: 0.0211 - accuracy: 0.997 - ETA: 15s - loss: 0.0210 - accuracy: 0.99  
7 - ETA: 15s - loss: 0.0210 - accuracy: 0.997 - ETA: 15s - loss: 0.0209 - accuracy: 0.997 - ETA: 14s - loss: 0.0209 - accuracy: 0.997  
- ETA: 14s - loss: 0.0208 - accuracy: 0.997 - ETA: 14s - loss: 0.0207 - accuracy: 0.997 - ETA: 13s - loss: 0.0207 - accuracy: 0.997 -  
ETA: 13s - loss: 0.0206 - accuracy: 0.997 - ETA: 13s - loss: 0.0206 - accuracy: 0.997 - ETA: 12s - loss: 0.0205 - accuracy: 0.997 - ET  
A: 12s - loss: 0.0205 - accuracy: 0.997 - ETA: 11s - loss: 0.0204 - accuracy: 0.997 - ETA: 11s - loss: 0.0203 - accuracy: 0.997 - ETA:  
11s - loss: 0.0203 - accuracy: 0.997WARNING:tensorflow:Method (on_train_batch_end) is slow compared to the batch update (0.175544). C  
heck your callbacks.  
322/322 [=====] - ETA: 10s - loss: 0.0202 - accuracy: 0.997 - ETA: 10s - loss: 0.0202 - accuracy: 0.997 - ET  
A: 9s - loss: 0.0201 - accuracy: 0.997 - ETA: 9s - loss: 0.0200 - accuracy: 0.99 - ETA: 9s - loss: 0.0200 - accuracy: 0.99 - ETA: 8s -  
loss: 0.0199 - accuracy: 0.99 - ETA: 8s - loss: 0.0206 - accuracy: 0.99 - ETA: 7s - loss: 0.0205 - accuracy: 0.99 - ETA: 7s - loss: 0.  
0205 - accuracy: 0.99 - ETA: 7s - loss: 0.0204 - accuracy: 0.99 - ETA: 6s - loss: 0.0203 - accuracy: 0.99 - ETA: 6s - loss: 0.0203 - a  
ccuracy: 0.99 - ETA: 5s - loss: 0.0202 - accuracy: 0.99 - ETA: 5s - loss: 0.0202 - accuracy: 0.99 - ETA: 5s - loss: 0.0201 - accuracy:  
0.99 - ETA: 4s - loss: 0.0201 - accuracy: 0.99 - ETA: 4s - loss: 0.0207 - accuracy: 0.99 - ETA: 3s - loss: 0.0206 - accuracy: 0.99 - E  
TA: 3s - loss: 0.0206 - accuracy: 0.99 - ETA: 3s - loss: 0.0205 - accuracy: 0.99 - ETA: 2s - loss: 0.0205 - accuracy: 0.99 - ETA: 2s -  
loss: 0.0204 - accuracy: 0.99 - ETA: 1s - loss: 0.0203 - accuracy: 0.99 - ETA: 1s - loss: 0.0203 - accuracy: 0.99 - ETA: 1s - loss: 0.  
0202 - accuracy: 0.99 - ETA: 0s - loss: 0.0202 - accuracy: 0.99 - ETA: 0s - loss: 0.0201 - accuracy: 0.99 - ETA: 0s - loss: 0.0201 - a  
ccuracy: 0.99 - 130s 405ms/step - loss: 0.0201 - accuracy: 0.9975 - val_loss: 0.0230 - val_accuracy: 0.9968  
Epoch 35/50  
186/322 [=====>.....] - ETA: 0s - loss: 0.0034 - accuracy: 1.00 - ETA: 28s - loss: 0.0033 - accuracy: 1.000 - ETA:  
35s - loss: 0.0034 - accuracy: 1.000 - ETA: 38s - loss: 0.0033 - accuracy: 1.000 - ETA: 49s - loss: 0.0033 - accuracy: 1.000 - ETA: 5  
3s - loss: 0.0033 - accuracy: 1.000 - ETA: 1:00 - loss: 0.0297 - accuracy: 0.99 - ETA: 1:00 - loss: 0.0264 - accuracy: 0.99 - ETA: 58s  
- loss: 0.0239 - accuracy: 0.9965 - ETA: 57s - loss: 0.0219 - accuracy: 0.996 - ETA: 1:01 - loss: 0.0203 - accuracy: 0.99 - ETA: 1:05  
- loss: 0.0190 - accuracy: 0.99 - ETA: 1:06 - loss: 0.0178 - accuracy: 0.99 - ETA: 1:06 - loss: 0.0168 - accuracy: 0.99 - ETA: 1:06 -  
loss: 0.0160 - accuracy: 0.99 - ETA: 1:05 - loss: 0.0152 - accuracy: 0.99 - ETA: 1:03 - loss: 0.0145 - accuracy: 0.99 - ETA: 1:04 - lo  
ss: 0.01
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```
2:10 - loss: 0.0519 - accuracy: 0.99 - ETA: 2:16 - loss: 0.0501 - accuracy: 0.99 - ETA: 2:27 - loss: 0.0485 - accuracy: 0.99 - ETA: 2:27 - loss: 0.0470 - accuracy: 0.99 - ETA: 2:40 - loss: 0.0456 - accuracy: 0.99 - ETA: 2:40 - loss: 0.0443 - accuracy: 0.99 - ETA: 2:37 - loss: 0.0431 - accuracy: 0.99 - ETA: 2:35 - loss: 0.0419 - accuracy: 0.99 - ETA: 2:34 - loss: 0.0407 - accuracy: 0.9946WARNING:tensorflow:Method (on_train_batch_end) is slow compared to the batch update (0.200052). Check your callbacks.  
139/322 [=====>.....] - ETA: 2:31 - loss: 0.0397 - accuracy: 0.99 - ETA: 2:27 - loss: 0.0386 - accuracy: 0.99 - ET  
A: 2:23 - loss: 0.0377 - accuracy: 0.99 - ETA: 2:20 - loss: 0.0367 - accuracy: 0.99 - ETA: 2:17 - loss: 0.0359 - accuracy: 0.99 - ET  
A: 2:14 - loss: 0.0350 - accuracy: 0.99 - ETA: 2:12 - loss: 0.0342 - accuracy: 0.99 - ETA: 2:10 - loss: 0.0334 - accuracy: 0.99 - ETA: 2:  
08 - loss: 0.0326 - accuracy: 0.99 - ETA: 2:05 - loss: 0.0319 - accuracy: 0.99 - ETA: 2:03 - loss: 0.0312 - accuracy: 0.99 - ETA: 2:01  
- loss: 0.0306 - accuracy: 0.99 - ETA: 1:58 - loss: 0.0299 - accuracy: 0.99 - ETA: 1:57 - loss: 0.0293 - accuracy: 0.99 - ETA: 1:56 -  
loss: 0.0287 - accuracy: 0.99 - ETA: 1:54 - loss: 0.0282 - accuracy: 0.99 - ETA: 1:53 - loss: 0.0276 - accuracy: 0.99 - ETA: 1:53 - l  
oss: 0.0271 - accuracy: 0.99 - ETA: 1:54 - loss: 0.0266 - accuracy: 0.99 - ETA: 1:53 - loss: 0.0261 - accuracy: 0.99 - ETA: 1:51 - los  
s: 0.0256 - accuracy: 0.99 - ETA: 1:51 - loss: 0.0252 - accuracy: 0.99 - ETA: 1:51 - loss: 0.0249 - accuracy: 0.99 - ETA: 1:51 - loss:  
0.0287 - accuracy: 0.99 - ETA: 1:50 - loss: 0.0283 - accuracy: 0.99 - ETA: 1:49 - loss: 0.0278 - accuracy: 0.99 - ETA: 1:48 - loss: 0.  
0274 - accuracy: 0.99 - ETA: 1:48 - loss: 0.0270 - accuracy: 0.99 - ETA: 1:48 - loss: 0.0266 - accuracy: 0.99 - ETA: 1:47 - loss: 0.02  
62 - accuracy: 0.99 - ETA: 1:48 - loss: 0.0258 - accuracy: 0.99 - ETA: 1:48 - loss: 0.0288 - accuracy: 0.99 - ETA: 1:47 - loss: 0.0284  
- accuracy: 0.99 - ETA: 1:46 - loss: 0.0280 - accuracy: 0.99 - ETA: 1:45 - loss: 0.0277 - accuracy: 0.99 - ETA: 1:44 - loss: 0.0273 -  
accuracy: 0.99 - ETA: 1:43 - loss: 0.0270 - accuracy: 0.99 - ETA: 1:43 - loss: 0.0266 - accuracy: 0.99 - ETA: 1:43 - loss: 0.0292 - a  
ccuracy: 0.99 - ETA: 1:42 - loss: 0.0289 - accuracy: 0.99 - ETA: 1:42 - loss: 0.0285 - accuracy: 0.99 - ETA: 1:41 - loss: 0.0282 - acc  
uracy: 0.99 - ETA: 1:40 - loss: 0.0279 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0275 - accuracy: 0.99 - ETA: 1:38 - loss: 0.0296 - accur  
acy: 0.99 - ETA: 1:37 - loss: 0.0292 - accuracy: 0.99 - ETA: 1:37 - loss: 0.0289 - accuracy: 0.99 - ETA: 1:36 - loss: 0.0286 - accurac  
y: 0.99 - ETA: 1:36 - loss: 0.0284 - accuracy: 0.99 - ETA: 1:35 - loss: 0.0281 - accuracy: 0.99 - ETA: 1:35 - loss: 0.0278 - accuracy:  
0.99 - ETA: 1:34 - loss: 0.0275 - accuracy: 0.99 - ETA: 1:33 - loss: 0.0273 - accuracy: 0.99 - ETA: 1:32 - loss: 0.0270 - accuracy: 0.  
99 - ETA: 1:32 - loss: 0.0288 - accuracy: 0.99 - ETA: 1:31 - loss: 0.0286 - accuracy: 0.99 - ETA: 1:30 - loss: 0.0283 - accuracy: 0.99  
- ETA: 1:30 - loss: 0.0281 - accuracy: 0.99 - ETA: 1:29 - loss: 0.0278 - accuracy: 0.99 - ETA: 1:28 - loss: 0.0276 - accuracy: 0.99 -  
ETA: 1:28 - loss: 0.0273 - accuracy: 0.99 - ETA: 1:27 - loss: 0.0271 - accuracy: 0.99 - ETA: 1:26 - loss: 0.0269 - accuracy: 0.99 - E  
TA: 1:26 - loss: 0.0284 - accuracy: 0.99 - ETA: 1:25 - loss: 0.0282 - accuracy: 0.99 - ETA: 1:25 - loss: 0.0280 - accuracy: 0.99 - ET  
A: 1:24 - loss: 0.0277 - accuracy: 0.99 - ETA: 1:23 - loss: 0.0275 - accuracy: 0.99 - ETA: 1:22 - loss: 0.0273 - accuracy: 0.99 - ET  
A: 1:22 - loss: 0.0271 - accuracy: 0.99 - ETA: 1:21 - loss: 0.0286 - accuracy: 0.99 - ETA: 1:21 - loss: 0.0283 - accuracy: 0.99 - ETA: 1:  
20 - loss: 0.0281 - accuracy: 0.99 - ETA: 1:19 - loss: 0.0279 - accuracy: 0.99 - ETA: 1:19 - loss: 0.0277 - accuracy: 0.99 - ETA: 1:18  
- loss: 0.0275 - accuracy: 0.99 - ETA: 1:18 - loss: 0.0273 - accuracy: 0.99 - ETA: 1:17 - loss: 0.0271 - accuracy: 0.99 - ETA: 1:17 -  
loss: 0.0269 - accuracy: 0.99 - ETA: 1:16 - loss: 0.0267 - accuracy: 0.99 - ETA: 1:16 - loss: 0.0265 - accuracy: 0.99 - ETA: 1:15 - l  
oss: 0.0263 - accuracy: 0.99 - ETA: 1:14 - loss: 0.0261 - accuracy: 0.99 - ETA: 1:14 - loss: 0.0259 - accuracy: 0.99 - ETA: 1:13 - los  
s: 0.0257 - accuracy: 0.99 - ETA: 1:13 - loss: 0.0255 - accuracy: 0.99 - ETA: 1:13 - loss: 0.0253 - accuracy: 0.99 - ETA: 1:12 - loss:  
0.0251 - accuracy: 0.99 - ETA: 1:12 - loss: 0.0250 - accuracy: 0.99 - ETA: 1:12 - loss: 0.0248 - accuracy: 0.99 - ETA: 1:11 - loss: 0.  
0246 - accuracy: 0.99 - ETA: 1:11 - loss: 0.0244 - accuracy: 0.99 - ETA: 1:11 - loss: 0.0242 - accuracy: 0.99 - ETA: 1:11 - loss: 0.02  
41 - accuracy: 0.99 - ETA: 1:11 - loss: 0.0239 - accuracy: 0.99 - ETA: 1:11 - loss: 0.0237 - accuracy: 0.99 - ETA: 1:11 - loss: 0.0236  
- accuracy: 0.99 - ETA: 1:11 - loss: 0.0234 - accuracy: 0.99 - ETA: 1:12 - loss: 0.0232 - accuracy: 0.99 - ETA: 1:12 - loss: 0.0231 -  
accuracy: 0.99 - ETA: 1:13 - loss: 0.0229 - accuracy: 0.99 - ETA: 1:13 - loss: 0.0228 - accuracy: 0.99 - ETA: 1:13 - loss: 0.0226 - a  
ccuracy: 0.99 - ETA: 1:13 - loss: 0.0224 - accuracy: 0.99 - ETA: 1:14 - loss: 0.0223 - accuracy: 0.99 - ETA: 1:15 - loss: 0.0221 - acc  
uracy: 0.99 - ETA: 1:16 - loss: 0.0237 - accuracy: 0.99 - ETA: 1:16 - loss: 0.0235 - accuracy: 0.99 - ETA: 1:17 - loss: 0.0249 - accur  
acy: 0.99 - ETA: 1:16 - loss: 0.0262 - accuracy: 0.9966WARNING:tensorflow:Method (on_train_batch_end) is slow compared to the batch up  
date (0.372591). Check your callbacks.  
141/322 [=====>.....] - ETA: 1:16 - loss: 0.0261 - accuracy: 0.99 - ETA: 1:16 - loss: 0.0271 - accuracy: 0.9965WARN  
ING:tensorflow:Method (on_train_batch_end) is slow compared to the batch update (0.270566). Check your callbacks.  
161/322 [=====>.....] - ETA: 1:16 - loss: 0.0271 - accuracy: 0.99 - ETA: 1:17 - loss: 0.0270 - accuracy: 0.99 - ET  
A: 1:17 - loss: 0.0270 - accuracy: 0.99 - ETA: 1:17 - loss: 0.0269 - accuracy: 0.99 - ETA: 1:16 - loss: 0.0268 - accuracy: 0.99 - ETA:  
1:16 - loss: 0.0267 - accuracy: 0.99 - ETA: 1:16 - loss: 0.0266 - accuracy: 0.99 - ETA: 1:16 - loss: 0.0276 - accuracy: 0.99 - ETA: 1:  
17 - loss: 0.0275 - accuracy: 0.99 - ETA: 1:17 - loss: 0.0274 - accuracy: 0.99 - ETA: 1:16 - loss: 0.0273 - accuracy: 0.99 - ETA: 1:16  
- loss: 0.0272 - accuracy: 0.99 - ETA: 1:18 - loss: 0.0270 - accuracy: 0.99 - ETA: 1:18 - loss: 0.0269 - accuracy: 0.99 - ETA: 1:17 -  
loss: 0.0268 - accuracy: 0.99 - ETA: 1:18 - loss: 0.0267 - accuracy: 0.99 - ETA: 1:18 - loss: 0.0266 - accuracy: 0.99 - ETA: 1:18 - l  
oss: 0.0265 - accuracy: 0.99 - ETA: 1:18 - loss: 0.0264 - accuracy: 0.99 - ETA: 1:17 - loss: 0.0263 - accuracy: 0.9967WARNING:tensorfl  
ow:Method (on_train_batch_end) is slow compared to the batch update (0.329083). Check your callbacks.  
164/322 [=====>.....] - ETA: 1:18 - loss: 0.0261 - accuracy: 0.99 - ETA: 1:17 - loss: 0.0260 - accuracy: 0.99 - ET  
A: 1:17 - loss: 0.0259 - accuracy: 0.9968WARNING:tensorflow:Method (on_train_batch_end) is slow compared to the batch update (0.22805  
8). Check your
```


[illegible]

2:03	-	loss:	0.0256	-	accuracy:	0.99	-	ETA:	2:03	-	loss:	0.0252	-	accuracy:	0.99	-	ETA:	2:03	-	loss:	0.0248	-	accuracy:	0.99	-	ETA:
2:04	-	loss:	0.0244	-	accuracy:	0.99	-	ETA:	2:05	-	loss:	0.0240	-	accuracy:	0.99	-	ETA:	2:04	-	loss:	0.0237	-	accuracy:	0.99	-	ETA:
2:03	-	loss:	0.0233	-	accuracy:	0.99	-	ETA:	2:02	-	loss:	0.0230	-	accuracy:	0.99	-	ETA:	2:02	-	loss:	0.0226	-	accuracy:	0.99	-	ETA:
2:02	-	loss:	0.0223	-	accuracy:	0.99	-	ETA:	2:01	-	loss:	0.0220	-	accuracy:	0.99	-	ETA:	1:59	-	loss:	0.0216	-	accuracy:	0.99	-	ETA:
1:58	-	loss:	0.0213	-	accuracy:	0.99	-	ETA:	1:58	-	loss:	0.0210	-	accuracy:	0.99	-	ETA:	1:57	-	loss:	0.0207	-	accuracy:	0.99	-	ETA:
1:55	-	loss:	0.0204	-	accuracy:	0.99	-	ETA:	1:54	-	loss:	0.0202	-	accuracy:	0.99	-	ETA:	1:53	-	loss:	0.0199	-	accuracy:	0.99	-	ETA:
1:52	-	loss:	0.0196	-	accuracy:	0.99	-	ETA:	1:52	-	loss:	0.0194	-	accuracy:	0.99	-	ETA:	1:51	-	loss:	0.0222	-	accuracy:	0.99	-	ETA:
1:50	-	loss:	0.0219	-	accuracy:	0.99	-	ETA:	1:49	-	loss:	0.0217	-	accuracy:	0.99	-	ETA:	1:49	-	loss:	0.0214	-	accuracy:	0.99	-	ETA:
1:48	-	loss:	0.0211	-	accuracy:	0.99	-	ETA:	1:47	-	loss:	0.0209	-	accuracy:	0.99	-	ETA:	1:46	-	loss:	0.0207	-	accuracy:	0.99	-	ETA:
1:46	-	loss:	0.0204	-	accuracy:	0.99	-	ETA:	1:46	-	loss:	0.0202	-	accuracy:	0.99	-	ETA:	1:45	-	loss:	0.0200	-	accuracy:	0.99	-	ETA:
1:44	-	loss:	0.0225	-	accuracy:	0.99	-	ETA:	1:43	-	loss:	0.0222	-	accuracy:	0.99	-	ETA:	1:42	-	loss:	0.0220	-	accuracy:	0.99	-	ETA:
1:43	-	loss:	0.0217	-	accuracy:	0.99	-	ETA:	1:42	-	loss:	0.0215	-	accuracy:	0.99	-	ETA:	1:41	-	loss:	0.0213	-	accuracy:	0.99	-	ETA:
1:40	-	loss:	0.0211	-	accuracy:	0.99	-	ETA:	1:41	-	loss:	0.0209	-	accuracy:	0.99	-	ETA:	1:40	-	loss:	0.0207	-	accuracy:	0.99	-	ETA:
1:39	-	loss:	0.0205	-	accuracy:	0.99	-	ETA:	1:40	-	loss:	0.0203	-	accuracy:	0.99	-	ETA:	1:40	-	loss:	0.0201	-	accuracy:	0.99	-	ETA:
1:39	-	loss:	0.0199	-	accuracy:	0.99	-	ETA:	1:38	-	loss:	0.0197	-	accuracy:	0.99	-	ETA:	1:38	-	loss:	0.0195	-	accuracy:	0.99	-	ETA:
1:37	-	loss:	0.0215	-	accuracy:	0.99	-	ETA:	1:36	-	loss:	0.0213	-	accuracy:	0.99	-	ETA:	1:35	-	loss:	0.0212	-	accuracy:	0.99	-	ETA:
1:35	-	loss:	0.0210	-	accuracy:	0.99	-	ETA:	1:34	-	loss:	0.0208	-	accuracy:	0.99	-	ETA:	1:34	-	loss:	0.0206	-	accuracy:	0.99	-	ETA:
1:33	-	loss:	0.0225	-	accuracy:	0.99	-	ETA:	1:33	-	loss:	0.0223	-	accuracy:	0.99	-	ETA:	1:32	-	loss:	0.0221	-	accuracy:	0.99	-	ETA:
1:32	-	loss:	0.0219	-	accuracy:	0.99	-	ETA:	1:31	-	loss:	0.0235	-	accuracy:	0.99	-	ETA:	1:30	-	loss:	0.0234	-	accuracy:	0.99	-	ETA:
1:30	-	loss:	0.0232	-	accuracy:	0.99	-	ETA:	1:29	-	loss:	0.0230	-	accuracy:	0.99	-	ETA:	1:29	-	loss:	0.0229	-	accuracy:	0.99	-	ETA:
1:28	-	loss:	0.0228	-	accuracy:	0.99	-	ETA:	1:27	-	loss:	0.0226	-	accuracy:	0.99	-	ETA:	1:27	-	loss:	0.0225	-	accuracy:	0.99	-	ETA:
1:26	-	loss:	0.0223	-	accuracy:	0.99	-	ETA:	1:26	-	loss:	0.0222	-	accuracy:</												

```
accuracy: 0.99 - ETA: 1s - loss: 0.0201 - accuracy: 0.99 - ETA: 0s - loss: 0.0200 - accuracy: 0.99 - ETA: 0s - loss: 0.0200 - accur  
acy: 0.99 - ETA: 0s - loss: 0.0199 - accuracy: 0.99 - 126s 393ms/step - loss: 0.0199 - accuracy: 0.9975 - val_loss: 0.0241 - val_accu  
racy: 0.9968  
Epoch 46/50  
186/322 [=====>.....] - ETA: 0s - loss: 0.0033 - accuracy: 1.00 - ETA: 33s - loss: 0.1068 - accuracy: 0.984 - ETA:  
43s - loss: 0.0724 - accuracy: 0.989 - ETA: 48s - loss: 0.0553 - accuracy: 0.992 - ETA: 53s - loss: 0.0451 - accuracy: 0.993 - ETA: 5  
8s - loss: 0.0383 - accuracy: 0.994 - ETA: 59s - loss: 0.0596 - accuracy: 0.991 - ETA: 1:00 - loss: 0.0528 - accuracy: 0.99 - ETA: 1:0  
1 - loss: 0.0475 - accuracy: 0.99 - ETA: 1:03 - loss: 0.0432 - accuracy: 0.99 - ETA: 1:12 - loss: 0.0398 - accuracy: 0.99 - ETA: 1:21  
- loss: 0.0369 - accuracy: 0.99 - ETA: 1:50 - loss: 0.0345 - accuracy: 0.99 - ETA: 2:03 - loss: 0.0450 - accuracy: 0.99 - ETA: 2:03 -  
loss: 0.0423 - accuracy: 0.99 - ETA: 2:03 - loss: 0.0401 - accuracy: 0.99 - ETA: 2:05 - loss: 0.0381 - accuracy: 0.99 - ETA: 2:06 - lo  
ss: 0.0363 - accuracy: 0.99 - ETA: 2:05 - loss: 0.0347 - accuracy: 0.99 - ETA: 2:04 - loss: 0.0333 - accuracy: 0.99 - ETA: 2:05 - los  
s: 0.0320 - accuracy: 0.99 - ETA: 2:09 - loss: 0.0308 - accuracy: 0.99 - ETA: 2:10 - loss: 0.0297 - accuracy: 0.99 - ETA: 2:10 - loss:  
0.0287 - accuracy: 0.99 - ETA: 2:09 - loss: 0.0278 - accuracy: 0.99 - ETA: 2:07 - loss: 0.0269 - accuracy: 0.99 - ETA: 2:05 - loss: 0.  
0261 - accuracy: 0.99 - ETA: 2:03 - loss: 0.0315 - accuracy: 0.99 - ETA: 2:02 - loss: 0.0306 - accuracy: 0.99 - ETA: 2:03 - loss: 0.02  
98 - accuracy: 0.99 - ETA: 2:03 - loss: 0.0290 - accuracy: 0.99 - ETA: 2:02 - loss: 0.0283 - accuracy: 0.99 - ETA: 2:02 - loss: 0.0276  
- accuracy: 0.99 - ETA: 2:01 - loss: 0.0269 - accuracy: 0.99 - ETA: 2:00 - loss: 0.0263 - accuracy: 0.99 - ETA: 1:58 - loss: 0.0257 -  
accuracy: 0.99 - ETA: 1:56 - loss: 0.0251 - accuracy: 0.99 - ETA: 1:55 - loss: 0.0246 - accuracy: 0.99 - ETA: 1:54 - loss: 0.0241 - a  
ccuracy: 0.99 - ETA: 1:55 - loss: 0.0236 - accuracy: 0.99 - ETA: 1:55 - loss: 0.0231 - accuracy: 0.99 - ETA: 1:55 - loss: 0.0226 - acc  
uracy: 0.99 - ETA: 2:00 - loss: 0.0266 - accuracy: 0.99 - ETA: 1:59 - loss: 0.0261 - accuracy: 0.99 - ETA: 1:58 - loss: 0.0256 - accur  
acy: 0.99 - ETA: 1:58 - loss: 0.0251 - accuracy: 0.99 - ETA: 1:58 - loss: 0.0246 - accuracy: 0.99 - ETA: 1:57 - loss: 0.0242 - accurac  
y: 0.99 - ETA: 1:58 - loss: 0.0238 - accuracy: 0.99 - ETA: 1:58 - loss: 0.0234 - accuracy: 0.99 - ETA: 1:57 - loss: 0.0230 - accuracy:  
0.99 - ETA: 1:56 - loss: 0.0227 - accuracy: 0.99 - ETA: 1:56 - loss: 0.0223 - accuracy: 0.99 - ETA: 1:56 - loss: 0.0254 - accuracy: 0.  
99 - ETA: 1:55 - loss: 0.0250 - accuracy: 0.99 - ETA: 1:54 - loss: 0.0246 - accuracy: 0.99 - ETA: 1:55 - loss: 0.0242 - accuracy: 0.99  
- ETA: 1:54 - loss: 0.0271 - accuracy: 0.99 - ETA: 1:53 - loss: 0.0267 - accuracy: 0.99 - ETA: 1:52 - loss: 0.0264 - accuracy: 0.99 -  
ETA: 1:53 - loss: 0.0261 - accuracy: 0.99 - ETA: 1:52 - loss: 0.0257 - accuracy: 0.99 - ETA: 1:51 - loss: 0.0284 - accuracy: 0.99 - E  
TA: 1:51 - loss: 0.0280 - accuracy: 0.99 - ETA: 1:51 - loss: 0.0277 - accuracy: 0.99 - ETA: 1:51 - loss: 0.0274 - accuracy: 0.99 - ET  
A: 1:50 - loss: 0.0271 - accuracy: 0.99 - ETA: 1:49 - loss: 0.0268 - accuracy: 0.99 - ETA: 1:49 - loss: 0.0266 - accuracy: 0.99 - ETA:  
1:48 - loss: 0.0263 - accuracy: 0.99 - ETA: 1:48 - loss: 0.0260 - accuracy: 0.99 - ETA: 1:47 - loss: 0.0257 - accuracy: 0.99 - ETA: 1:  
47 - loss: 0.0254 - accuracy: 0.99 - ETA: 1:46 - loss: 0.0252 - accuracy: 0.99 - ETA: 1:45 - loss: 0.0249 - accuracy: 0.99 - ETA: 1:44  
- loss: 0.0246 - accuracy: 0.99 - ETA: 1:44 - loss: 0.0244 - accuracy: 0.99 - ETA: 1:43 - loss: 0.0241 - accuracy: 0.99 - ETA: 1:42 -  
loss: 0.0239 - accuracy: 0.99 - ETA: 1:42 - loss: 0.0236 - accuracy: 0.99 - ETA: 1:41 - loss: 0.0234 - accuracy: 0.99 - ETA: 1:40 - l  
oss: 0.0254 - accuracy: 0.99 - ETA: 1:40 - loss: 0.0251 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0249 - accuracy: 0.99 - ETA: 1:38 - los  
s: 0.0246 - accuracy: 0.99 - ETA: 1:38 - loss: 0.0244 - accuracy: 0.99 - ETA: 1:37 - loss: 0.0242 - accuracy: 0.99 - ETA: 1:37 - loss:  
0.0239 - accuracy: 0.99 - ETA: 1:36 - loss: 0.0237 - accuracy: 0.99 - ETA: 1:36 - loss: 0.0235 - accuracy: 0.99 - ETA: 1:35 - loss: 0.  
0233 - accuracy: 0.99 - ETA: 1:35 - loss: 0.0231 - accuracy: 0.99 - ETA: 1:34 - loss: 0.0229 - accuracy: 0.99 - ETA: 1:33 - loss: 0.02  
27 - accuracy: 0.99 - ETA: 1:32 - loss: 0.0225 - accuracy: 0.99 - ETA: 1:32 - loss: 0.0223 - accuracy: 0.99 - ETA: 1:32 - loss: 0.0221  
- accuracy: 0.99 - ETA: 1:32 - loss: 0.0219 - accuracy: 0.99 - ETA: 1:32 - loss: 0.0217 - accuracy: 0.99 - ETA: 1:32 - loss: 0.0215 -  
accuracy: 0.99 - ETA: 1:33 - loss: 0.0213 - accuracy: 0.99 - ETA: 1:34 - loss: 0.0211 - accuracy: 0.99 - ETA: 1:35 - loss: 0.0210 - a  
ccuracy: 0.99 - ETA: 1:36 - loss: 0.0208 - accuracy: 0.99 - ETA: 1:37 - loss: 0.0206 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0204 - acc  
uracy: 0.99 - ETA: 1:39 - loss: 0.0222 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0220 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0219 - accur  
acy: 0.99 - ETA: 1:39 - loss: 0.0217 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0215 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0231 - accurac  
y: 0.99 - ETA: 1:39 - loss: 0.0229 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0228 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0226 - accuracy:  
0.99 - ETA: 1:38 - loss: 0.0224 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0223 - accuracy: 0.99 - ETA: 1:39 - loss: 0.0221 - accuracy: 0.  
99 - ETA: 1:38 - loss: 0.0220 - accuracy: 0.99 - ETA: 1:37 - loss: 0.0219 - accuracy: 0.99 - ETA: 1:37 - loss: 0.0217 - accuracy: 0.99  
- ETA: 1:37 - loss: 0.0216 - accuracy: 0.99 - ETA: 1:37 - loss: 0.0214 - accuracy: 0.99 - ETA: 1:37 - loss: 0.0213 - accuracy: 0.99 -  
ETA: 1:37 - loss: 0.0211 - accuracy: 0.99 - ETA: 1:36 - loss: 0.0224 - accuracy: 0.99 - ETA: 1:36 - loss: 0.0223 - accuracy: 0.99 - E  
TA: 1:35 - loss: 0.0221 - accuracy: 0.99 - ETA: 1:36 - loss: 0.0220 - accuracy: 0.99 - ETA: 1:35 - loss: 0.0219 - accuracy: 0.99 - ET  
A: 1:35 - loss: 0.0217 - accuracy: 0.99 - ETA: 1:35 - loss: 0.0216 - accuracy: 0.99 - ETA: 1:34 - loss: 0.0215 - accuracy: 0.99 - ETA:  
1:33 - loss: 0.0213 - accuracy: 0.99 - ETA: 1:33 - loss: 0.0212 - accuracy: 0.99 - ETA: 1:33 - loss: 0.0211 - accuracy: 0.99 - ETA: 1:  
32 - loss: 0.0210 - accuracy: 0.99 - ETA: 1:32 - loss: 0.0208 - accuracy: 0.99 - ETA: 1:31 - loss: 0.0207 - accuracy: 0.99 - ETA: 1:32  
- loss: 0.0206 - accuracy: 0.99 - ETA: 1:31 - loss: 0.0205 - accuracy: 0.99 - ETA: 1:31 - loss: 0.0204 - accuracy: 0.99 - ETA: 1:30 -  
loss: 0.0202 - accuracy: 0.99 - ETA: 1:30 - loss: 0.0201 - accuracy: 0.99 - ETA: 1:30 - loss: 0.0200 - accuracy: 0.99 - ETA: 1:29 - l  
oss: 0.0199
```

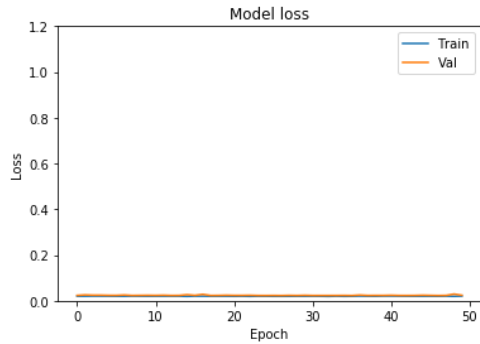

[illegible]

[illegible]

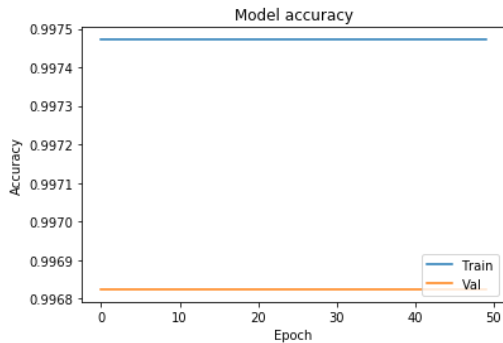
[illegible]

uracy: 0.99 - ETA: 3s - loss: 0.0212 - accuracy: 0.99 - ETA: 3s - loss: 0.0212 - accuracy: 0.99 - ETA: 2s - loss: 0.0211 - accuracy: 0.99 - ETA: 2s - loss: 0.0211 - accuracy: 0.99 - ETA: 2s - loss: 0.0210 - accuracy: 0.99 - ETA: 2s - loss: 0.0210 - accuracy: 0.99 - ETA: 2s - loss: 0.0209 - accuracy: 0.99 - ETA: 1s - loss: 0.0209 - accuracy: 0.99 - ETA: 1s - loss: 0.0208 - accuracy: 0.99 - ETA: 1s - loss: 0.0208 - accuracy: 0.99 - ETA: 1s - loss: 0.0207 - accuracy: 0.99 - ETA: 0s - loss: 0.0207 - accuracy: 0.99 - ETA: 0s - loss: 0.0206 - accuracy: 0.99 - ETA: 0s - loss: 0.0205 - accuracy: 0.99 - ETA: 0s - loss: 0.0205 - accuracy: 0.99 - 90s 281ms/step - loss: 0.0205 - accuracy: 0.9975 - val_loss: 0.0239 - val_accuracy: 0.9968

```
In [14]: plt.plot(hist_out.history['loss'])
plt.plot(hist_out.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Val'], loc='upper right')
plt.ylim(top=1.2, bottom=0)
plt.show()
```



```
In [15]: plt.plot(hist_out.history['accuracy'])
plt.plot(hist_out.history['val_accuracy'])
plt.title('Model accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train', 'Val'], loc='lower right')
plt.show()
```



```
In [58]: ftd = train_data[(train_data['Label'] == 'Normal') |
                        ((train_data['Label'] == 'Pneumonia') & (train_data['Label_2_Virus_category'] == 'COVID-19'))]

# Create a target attribute where value = positive if 'Pneumonia + COVID-19' or value = negative if 'Normal'
ftd['target'] = ['negative' if holder == 'Normal' else 'positive' for holder in ftd['Label']]

ftd= shuffle(ftd, random_state=1)

final_validation_data = ftd.iloc[1000:, :]
ftd = ftd.iloc[:1000, :]

print(f"Final train data shape : {ftd.shape}")
ftd.sample(10)
```

C:\Users\Micky\Anaconda3\lib\site-packages\ipykernel_launcher.py:6: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.

Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

Final train data shape : (1000, 7)

Out[58]:

	Unnamed: 0	X_ray_image_name	Label	Dataset_type	Label_2_Virus_category	Label_1_Virus_category	target
719	719	NORMAL2-IM-0564-0001.jpeg	Normal	TRAIN	NaN	NaN	negative
164	164	IM-0349-0001.jpeg	Normal	TRAIN	NaN	NaN	negative
561	561	IM-0722-0001.jpeg	Normal	TRAIN	NaN	NaN	negative
169	169	IM-0343-0001.jpeg	Normal	TRAIN	NaN	NaN	negative
1170	1170	NORMAL2-IM-1149-0001.jpeg	Normal	TRAIN	NaN	NaN	negative
558	558	IM-0728-0001.jpeg	Normal	TRAIN	NaN	NaN	negative
611	611	IM-0752-0001.jpeg	Normal	TRAIN	NaN	NaN	negative
256	256	IM-0467-0001.jpeg	Normal	TRAIN	NaN	NaN	negative
1286	1286	NORMAL2-IM-1311-0001.jpeg	Normal	TRAIN	NaN	NaN	negative
1102	1102	NORMAL2-IM-1064-0001.jpeg	Normal	TRAIN	NaN	NaN	negative

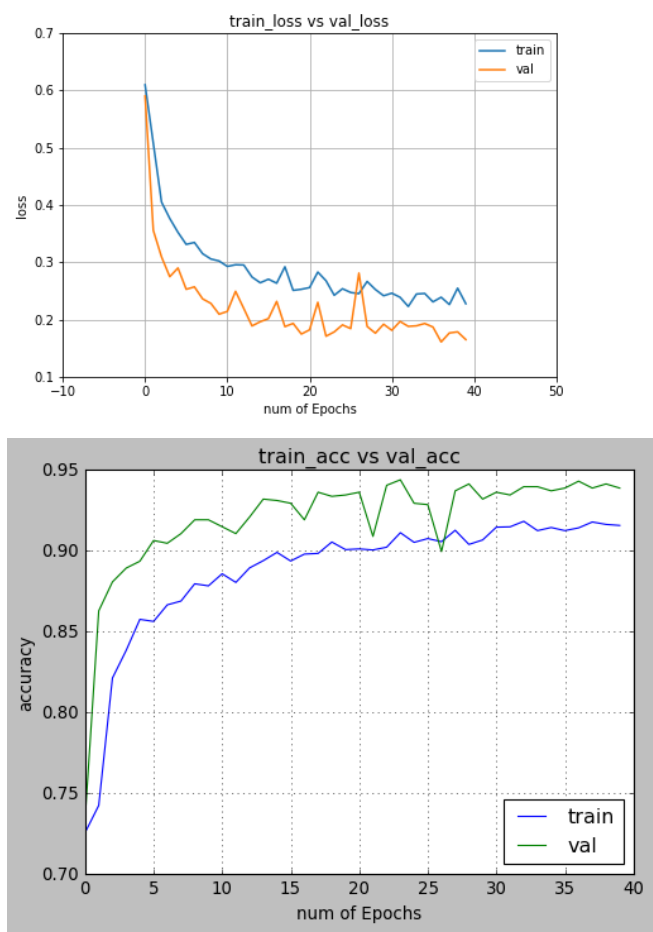
```
In [12]: model.compile(loss='categorical_crossentropy', optimizer='rmsprop', metrics=["accuracy"])
```

```
In [14]: hist = model.fit(X_train, y_train, batch_size=16, epochs=num_epoch, verbose=1, validation_data=(X_test, y_test))
```

```
Epoch 1/40
293/293 [=====] - 4s 14ms/step - loss: 0.6097 - accuracy: 0.7254 - val_loss: 0.5900 - val_accuracy: 0.7338
Epoch 2/40
293/293 [=====] - 4s 13ms/step - loss: 0.5076 - accuracy: 0.7423 - val_loss: 0.3554 - val_accuracy: 0.8626
Epoch 3/40
293/293 [=====] - 4s 14ms/step - loss: 0.4049 - accuracy: 0.8211 - val_loss: 0.3092 - val_accuracy: 0.8805
Epoch 4/40
293/293 [=====] - 4s 13ms/step - loss: 0.3763 - accuracy: 0.8382 - val_loss: 0.2746 - val_accuracy: 0.8891
Epoch 5/40
293/293 [=====] - 4s 13ms/step - loss: 0.3520 - accuracy: 0.8574 - val_loss: 0.2899 - val_accuracy: 0.8933
Epoch 6/40
293/293 [=====] - 4s 14ms/step - loss: 0.3311 - accuracy: 0.8561 - val_loss: 0.2526 - val_accuracy: 0.9061
Epoch 7/40
293/293 [=====] - 4s 14ms/step - loss: 0.3344 - accuracy: 0.8664 - val_loss: 0.2569 - val_accuracy: 0.9044
Epoch 8/40
293/293 [=====] - 4s 13ms/step - loss: 0.3148 - accuracy: 0.8687 - val_loss: 0.2359 - val_accuracy: 0.9104
Epoch 9/40
293/293 [=====] - 4s 13ms/step - loss: 0.3055 - accuracy: 0.8794 - val_loss: 0.2280 - val_accuracy: 0.9189
Epoch 10/40
293/293 [=====] - 4s 13ms/step - loss: 0.3021 - accuracy: 0.8781 - val_loss: 0.2092 - val_accuracy: 0.9189
Epoch 11/40
293/293 [=====] - 4s 14ms/step - loss: 0.2926 - accuracy: 0.8856 - val_loss: 0.2140 - val_accuracy: 0.9147
Epoch 12/40
293/293 [=====] - 4s 13ms/step - loss: 0.2955 - accuracy: 0.8802 - val_loss: 0.2491 - val_accuracy: 0.9104
Epoch 13/40
293/293 [=====] - 4s 13ms/step - loss: 0.2951 - accuracy: 0.8892 - val_loss: 0.2191 - val_accuracy: 0.9206
Epoch 14/40
293/293 [=====] - 4s 13ms/step - loss: 0.2740 - accuracy: 0.8937 - val_loss: 0.1886 - val_accuracy: 0.9317
Epoch 15/40
293/293 [=====] - 4s 14ms/step - loss: 0.2640 - accuracy: 0.8988 - val_loss: 0.1960 - val_accuracy: 0.9309
Epoch 16/40
293/293 [=====] - 4s 13ms/step - loss: 0.2702 - accuracy: 0.8935 - val_loss: 0.2015 - val_accuracy: 0.9292
Epoch 17/40
293/293 [=====] - 4s 14ms/step - loss: 0.2633 - accuracy: 0.8977 - val_loss: 0.2315 - val_accuracy: 0.9189
Epoch 18/40
293/293 [=====] - 4s 14ms/step - loss: 0.2920 - accuracy: 0.8982 - val_loss: 0.1874 - val_accuracy: 0.9360 -
accuracy - ETA
Epoch 19/40
293/293 [=====] - 4s 14ms/step - loss: 0.2508 - accuracy: 0.9052 - val_loss: 0.1931 - val_accuracy: 0.9334
Epoch 20/40
293/293 [=====] - 4s 13ms/step - loss: 0.2526 - accuracy: 0.9005 - val_loss: 0.1745 - val_accuracy: 0.9343
Epoch 21/40
293/293 [=====] - 4s 13ms/step - loss: 0.2556 - accuracy: 0.9009 - val_loss: 0.1819 - val_accuracy: 0.9360
Epoch 22/40
293/293 [=====] - 4s 14ms/step - loss: 0.2827 - accuracy: 0.9003 - val_loss: 0.2299 - val_accuracy: 0.9087
Epoch 23/40
293/293 [=====] - 4s 14ms/step - loss: 0.2675 - accuracy: 0.9020 - val_loss: 0.1707 - val_accuracy: 0.9403
Epoch 24/40
293/293 [=====] - 4s 13ms/step - loss: 0.2422 - accuracy: 0.9110 - val_loss: 0.1782 - val_accuracy: 0.9437
Epoch 25/40
293/293 [=====] - 4s 13ms/step - loss: 0.2536 - accuracy: 0.9050 - val_loss: 0.1905 - val_accuracy: 0.9292
Epoch 26/40
293/293 [=====] - 4s 13ms/step - loss: 0.2471 - accuracy: 0.9073 - val_loss: 0.1840 - val_accuracy: 0.9283
Epoch 27/40
293/293 [=====] - 4s 13ms/step - loss: 0.2450 - accuracy: 0.9054 - val_loss: 0.2809 - val_accuracy: 0.8993
Epoch 28/40
293/293 [=====] - 4s 13ms/step - loss: 0.2663 - accuracy: 0.9125 - val_loss: 0.1877 - val_accuracy: 0.9369
Epoch 29/40
293/293 [=====] - 4s 13ms/step - loss: 0.2523 - accuracy: 0.9037 - val_loss: 0.1760 - val_accuracy: 0.9411
Epoch 30/40
293/293 [=====] - 4s 13ms/step - loss: 0.2414 - accuracy: 0.9065 - val_loss: 0.1916 - val_accuracy: 0.9317
Epoch 31/40
293/293 [=====] - 4s 13ms/step - loss: 0.2458 - accuracy: 0.9144 - val_loss: 0.1810 - val_accuracy: 0.9360
Epoch 32/40
293/293 [=====] - 4s 13ms/step - loss: 0.2387 - accuracy: 0.9146 - val_loss: 0.1965 - val_accuracy: 0.9343
Epoch 33/40
293/293 [=====] - 4s 13ms/step - loss: 0.2226 - accuracy: 0.9180 - val_loss: 0.1879 - val_accuracy: 0.9394
Epoch 34/40
293/293 [=====] - 4s 13ms/step - loss: 0.2443 - accuracy: 0.9123 - val_loss: 0.1889 - val_accuracy: 0.9394
Epoch 35/40
293/293 [=====] - 4s 13ms/step - loss: 0.2453 - accuracy: 0.9142 - val_loss: 0.1929 - val_accuracy: 0.9369
Epoch 36/40
293/293 [=====] - 4s 13ms/step - loss: 0.2307 - accuracy: 0.9123 - val_loss: 0.1868 - val_accuracy: 0.9386
Epoch 37/40
293/293 [=====] - 4s 13ms/step - loss: 0.2388 - accuracy: 0.9140 - val_loss: 0.1607 - val_accuracy: 0.9428
Epoch 38/40
293/293 [=====] - 4s 13ms/step - loss: 0.2259 - accuracy: 0.9176 - val_loss: 0.1764 - val_accuracy: 0.9386
Epoch 39/40
293/293 [=====] - 4s 14ms/step - loss: 0.2546 - accuracy: 0.9161 - val_loss: 0.1783 - val_accuracy: 0.9411
Epoch 40/40
293/293 [=====] - 4s 13ms/step - loss: 0.2274 - accuracy: 0.9155 - val_loss: 0.1648 - val_accuracy: 0.9386
```

```
In [ ]: hist = model.fit(X_train, y_train, batch_size=16, epochs=num_epoch, verbose=1, validation_data=(X_test, y_test))
```

```
In [15]: display_loss_accuracy(hist)
```



Effective Net80 Training and testing

```
In [11]: image_input = Input(shape=(224, 224, 3))
model = EfficientNetB0(input_tensor=image_input, include_top=True, weights='imagenet')
model.summary()
last_layer = model.get_layer('avg_pool').output
x= Flatten(name='flatten')(last_layer)
out = Dense(num_classes, activation='softmax', name='output_layer')(x)
custom_resnet_model = Model(inputs=image_input, outputs= out)
custom_resnet_model.summary()
```

Model: "efficientnetb0"

Layer (type)	Output Shape	Param #	Connected to
=====			
input_1 (InputLayer)	[(None, 224, 224, 3)]	0	
rescaling (Rescaling)	(None, 224, 224, 3)	0	input_1[0][0]
normalization (Normalization)	(None, 224, 224, 3)	7	rescaling[0][0]
stem_conv_pad (ZeroPadding2D)	(None, 225, 225, 3)	0	normalization[0][0]
stem_conv (Conv2D)	(None, 112, 112, 32)	864	stem_conv_pad[0][0]
stem_bn (BatchNormalization)	(None, 112, 112, 32)	128	stem_conv[0][0]
stem_activation (Activation)	(None, 112, 112, 32)	0	stem_bn[0][0]
block1a_dwconv (DepthwiseConv2D)	(None, 112, 112, 32)	288	stem_activation[0][0]
block1a_bn (BatchNormalization)	(None, 112, 112, 32)	128	block1a_dwconv[0][0]
block1a_activation (Activation)	(None, 112, 112, 32)	0	block1a_bn[0][0]
block1a_se_squeeze (GlobalAveragePooling2D)	(None, 32)	0	block1a_activation[0][0]
block1a_se_reshape (Reshape)	(None, 1, 1, 32)	0	block1a_se_squeeze[0][0]
block1a_se_reduce (Conv2D)	(None, 1, 1, 8)	264	block1a_se_reshape[0][0]
block1a_se_expand (Conv2D)	(None, 1, 1, 32)	288	block1a_se_reduce[0][0]
block1a_se_excite (Multiply)	(None, 112, 112, 32)	0	block1a_activation[0][0] block1a_se_expand[0][0]
block1a_project_conv (Conv2D)	(None, 112, 112, 16)	512	block1a_se_excite[0][0]
block1a_project_bn (BatchNormalization)	(None, 112, 112, 16)	64	block1a_project_conv[0][0]
block2a_expand_conv (Conv2D)	(None, 112, 112, 96)	1536	block1a_project_bn[0][0]
block2a_expand_bn (BatchNormalization)	(None, 112, 112, 96)	384	block2a_expand_conv[0][0]
block2a_expand_activation (Activation)	(None, 112, 112, 96)	0	block2a_expand_bn[0][0]
block2a_dwconv_pad (ZeroPadding2D)	(None, 113, 113, 96)	0	block2a_expand_activation[0][0]
block2a_dwconv (DepthwiseConv2D)	(None, 56, 56, 96)	864	block2a_dwconv_pad[0][0]
block2a_bn (BatchNormalization)	(None, 56, 56, 96)	384	block2a_dwconv[0][0]
block2a_activation (Activation)	(None, 56, 56, 96)	0	block2a_bn[0][0]
block2a_se_squeeze (GlobalAveragePooling2D)	(None, 96)	0	block2a_activation[0][0]
block2a_se_reshape (Reshape)	(None, 1, 1, 96)	0	block2a_se_squeeze[0][0]
block2a_se_reduce (Conv2D)	(None, 1, 1, 4)	388	block2a_se_reshape[0][0]
block2a_se_expand (Conv2D)	(None, 1, 1, 96)	480	block2a_se_reduce[0][0]
block2a_se_excite (Multiply)	(None, 56, 56, 96)	0	block2a_activation[0][0] block2a_se_expand[0][0]
block2a_project_conv (Conv2D)	(None, 56, 56, 24)	2304	block2a_se_excite[0][0]
block2a_project_bn (BatchNormalization)	(None, 56, 56, 24)	96	block2a_project_conv[0][0]
block2b_expand_conv (Conv2D)	(None, 56, 56, 144)	3456	block2a_project_bn[0][0]
block2b_expand_bn (BatchNormalization)	(None, 56, 56, 144)	576	block2b_expand_conv[0][0]
block2b_expand_activation (Activation)	(None, 56, 56, 144)	0	block2b_expand_bn[0][0]
block2b_dwconv (DepthwiseConv2D)	(None, 56, 56, 144)	1296	block2b_expand_activation[0][0]
block2b_bn (BatchNormalization)	(None, 56, 56, 144)	576	block2b_dwconv[0][0]
block2b_activation (Activation)	(None, 56, 56, 144)	0	block2b_bn[0][0]
block2b_se_squeeze (GlobalAveragePooling2D)	(None, 144)	0	block2b_activation[0][0]
block2b_se_reshape (Reshape)	(None, 1, 1, 144)	0	block2b_se_squeeze[0][0]
block2b_se_reduce (Conv2D)	(None, 1, 1, 6)	870	block2b_se_reshape[0][0]
block2b_se_expand (Conv2D)	(None, 1, 1, 144)	1008	block2b_se_reduce[0][0]
block2b_se_excite (Multiply)	(None, 56, 56, 144)	0	block2b_activation[0][0] block2b_se_expand[0][0]

block2b_project_conv (Conv2D)	(None, 56, 56, 24)	3456	block2b_se_excite[0][0]
block2b_project_bn (BatchNormal	(None, 56, 56, 24)	96	block2b_project_conv[0][0]
block2b_drop (Dropout)	(None, 56, 56, 24)	0	block2b_project_bn[0][0]
block2b_add (Add)	(None, 56, 56, 24)	0	block2b_drop[0][0] block2a_project_bn[0][0]
block3a_expand_conv (Conv2D)	(None, 56, 56, 144)	3456	block2b_add[0][0]
block3a_expand_bn (BatchNormali	(None, 56, 56, 144)	576	block3a_expand_conv[0][0]
block3a_expand_activation (Acti	(None, 56, 56, 144)	0	block3a_expand_bn[0][0]
block3a_dwconv_pad (ZeroPadding	(None, 59, 59, 144)	0	block3a_expand_activation[0][0]
block3a_dwconv (DepthwiseConv2D	(None, 28, 28, 144)	3600	block3a_dwconv_pad[0][0]
block3a_bn (BatchNormalization)	(None, 28, 28, 144)	576	block3a_dwconv[0][0]
block3a_activation (Activation)	(None, 28, 28, 144)	0	block3a_bn[0][0]
block3a_se_squeeze (GlobalAvera	(None, 144)	0	block3a_activation[0][0]
block3a_se_reshape (Reshape)	(None, 1, 1, 144)	0	block3a_se_squeeze[0][0]
block3a_se_reduce (Conv2D)	(None, 1, 1, 6)	870	block3a_se_reshape[0][0]
block3a_se_expand (Conv2D)	(None, 1, 1, 144)	1008	block3a_se_reduce[0][0]
block3a_se_excite (Multiply)	(None, 28, 28, 144)	0	block3a_activation[0][0] block3a_se_expand[0][0]
block3a_project_conv (Conv2D)	(None, 28, 28, 40)	5760	block3a_se_excite[0][0]
block3a_project_bn (BatchNormal	(None, 28, 28, 40)	160	block3a_project_conv[0][0]
block3b_expand_conv (Conv2D)	(None, 28, 28, 240)	9600	block3a_project_bn[0][0]
block3b_expand_bn (BatchNormali	(None, 28, 28, 240)	960	block3b_expand_conv[0][0]
block3b_expand_activation (Acti	(None, 28, 28, 240)	0	block3b_expand_bn[0][0]
block3b_dwconv (DepthwiseConv2D	(None, 28, 28, 240)	6000	block3b_expand_activation[0][0]
block3b_bn (BatchNormalization)	(None, 28, 28, 240)	960	block3b_dwconv[0][0]
block3b_activation (Activation)	(None, 28, 28, 240)	0	block3b_bn[0][0]
block3b_se_squeeze (GlobalAvera	(None, 240)	0	block3b_activation[0][0]
block3b_se_reshape (Reshape)	(None, 1, 1, 240)	0	block3b_se_squeeze[0][0]
block3b_se_reduce (Conv2D)	(None, 1, 1, 10)	2410	block3b_se_reshape[0][0]
block3b_se_expand (Conv2D)	(None, 1, 1, 240)	2640	block3b_se_reduce[0][0]
block3b_se_excite (Multiply)	(None, 28, 28, 240)	0	block3b_activation[0][0] block3b_se_expand[0][0]
block3b_project_conv (Conv2D)	(None, 28, 28, 40)	9600	block3b_se_excite[0][0]
block3b_project_bn (BatchNormal	(None, 28, 28, 40)	160	block3b_project_conv[0][0]
block3b_drop (Dropout)	(None, 28, 28, 40)	0	block3b_project_bn[0][0]
block3b_add (Add)	(None, 28, 28, 40)	0	block3b_drop[0][0] block3a_project_bn[0][0]
block4a_expand_conv (Conv2D)	(None, 28, 28, 240)	9600	block3b_add[0][0]
block4a_expand_bn (BatchNormali	(None, 28, 28, 240)	960	block4a_expand_conv[0][0]
block4a_expand_activation (Acti	(None, 28, 28, 240)	0	block4a_expand_bn[0][0]
block4a_dwconv_pad (ZeroPadding	(None, 29, 29, 240)	0	block4a_expand_activation[0][0]
block4a_dwconv (DepthwiseConv2D	(None, 14, 14, 240)	2160	block4a_dwconv_pad[0][0]
block4a_bn (BatchNormalization)	(None, 14, 14, 240)	960	block4a_dwconv[0][0]
block4a_activation (Activation)	(None, 14, 14, 240)	0	block4a_bn[0][0]
block4a_se_squeeze (GlobalAvera	(None, 240)	0	block4a_activation[0][0]
block4a_se_reshape (Reshape)	(None, 1, 1, 240)	0	block4a_se_squeeze[0][0]
block4a_se_reduce (Conv2D)	(None, 1, 1, 10)	2410	block4a_se_reshape[0][0]
block4a_se_expand (Conv2D)	(None, 1, 1, 240)	2640	block4a_se_reduce[0][0]

block4a_se_excite (Multiply)	(None, 14, 14, 240)	0	block4a_activation[0][0] block4a_se_expand[0][0]
block4a_project_conv (Conv2D)	(None, 14, 14, 80)	19200	block4a_se_excite[0][0]
block4a_project_bn (BatchNormal	(None, 14, 14, 80)	320	block4a_project_conv[0][0]
block4b_expand_conv (Conv2D)	(None, 14, 14, 480)	38400	block4a_project_bn[0][0]
block4b_expand_bn (BatchNormali	(None, 14, 14, 480)	1920	block4b_expand_conv[0][0]
block4b_expand_activation (Acti	(None, 14, 14, 480)	0	block4b_expand_bn[0][0]
block4b_dwconv (DepthwiseConv2D	(None, 14, 14, 480)	4320	block4b_expand_activation[0][0]
block4b_bn (BatchNormalization)	(None, 14, 14, 480)	1920	block4b_dwconv[0][0]
block4b_activation (Activation)	(None, 14, 14, 480)	0	block4b_bn[0][0]
block4b_se_squeeze (GlobalAvera	(None, 480)	0	block4b_activation[0][0]
block4b_se_reshape (Reshape)	(None, 1, 1, 480)	0	block4b_se_squeeze[0][0]
block4b_se_reduce (Conv2D)	(None, 1, 1, 20)	9620	block4b_se_reshape[0][0]
block4b_se_expand (Conv2D)	(None, 1, 1, 480)	10080	block4b_se_reduce[0][0]
block4b_se_excite (Multiply)	(None, 14, 14, 480)	0	block4b_activation[0][0] block4b_se_expand[0][0]
block4b_project_conv (Conv2D)	(None, 14, 14, 80)	38400	block4b_se_excite[0][0]
block4b_project_bn (BatchNormal	(None, 14, 14, 80)	320	block4b_project_conv[0][0]
block4b_drop (Dropout)	(None, 14, 14, 80)	0	block4b_project_bn[0][0]
block4b_add (Add)	(None, 14, 14, 80)	0	block4b_drop[0][0] block4a_project_bn[0][0]
block4c_expand_conv (Conv2D)	(None, 14, 14, 480)	38400	block4b_add[0][0]
block4c_expand_bn (BatchNormali	(None, 14, 14, 480)	1920	block4c_expand_conv[0][0]
block4c_expand_activation (Acti	(None, 14, 14, 480)	0	block4c_expand_bn[0][0]
block4c_dwconv (DepthwiseConv2D	(None, 14, 14, 480)	4320	block4c_expand_activation[0][0]
block4c_bn (BatchNormalization)	(None, 14, 14, 480)	1920	block4c_dwconv[0][0]
block4c_activation (Activation)	(None, 14, 14, 480)	0	block4c_bn[0][0]
block4c_se_squeeze (GlobalAvera	(None, 480)	0	block4c_activation[0][0]
block4c_se_reshape (Reshape)	(None, 1, 1, 480)	0	block4c_se_squeeze[0][0]
block4c_se_reduce (Conv2D)	(None, 1, 1, 20)	9620	block4c_se_reshape[0][0]
block4c_se_expand (Conv2D)	(None, 1, 1, 480)	10080	block4c_se_reduce[0][0]
block4c_se_excite (Multiply)	(None, 14, 14, 480)	0	block4c_activation[0][0] block4c_se_expand[0][0]
block4c_project_conv (Conv2D)	(None, 14, 14, 80)	38400	block4c_se_excite[0][0]
block4c_project_bn (BatchNormal	(None, 14, 14, 80)	320	block4c_project_conv[0][0]
block4c_drop (Dropout)	(None, 14, 14, 80)	0	block4c_project_bn[0][0]
block4c_add (Add)	(None, 14, 14, 80)	0	block4c_drop[0][0] block4b_add[0][0]
block5a_expand_conv (Conv2D)	(None, 14, 14, 480)	38400	block4c_add[0][0]
block5a_expand_bn (BatchNormali	(None, 14, 14, 480)	1920	block5a_expand_conv[0][0]
block5a_expand_activation (Acti	(None, 14, 14, 480)	0	block5a_expand_bn[0][0]
block5a_dwconv (DepthwiseConv2D	(None, 14, 14, 480)	12000	block5a_expand_activation[0][0]
block5a_bn (BatchNormalization)	(None, 14, 14, 480)	1920	block5a_dwconv[0][0]
block5a_activation (Activation)	(None, 14, 14, 480)	0	block5a_bn[0][0]
block5a_se_squeeze (GlobalAvera	(None, 480)	0	block5a_activation[0][0]
block5a_se_reshape (Reshape)	(None, 1, 1, 480)	0	block5a_se_squeeze[0][0]
block5a_se_reduce (Conv2D)	(None, 1, 1, 20)	9620	block5a_se_reshape[0][0]
block5a_se_expand (Conv2D)	(None, 1, 1, 480)	10080	block5a_se_reduce[0][0]

block5a_se_excite (Multiply)	(None, 14, 14, 480)	0	block5a_activation[0][0] block5a_se_expand[0][0]
block5a_project_conv (Conv2D)	(None, 14, 14, 112)	53760	block5a_se_excite[0][0]
block5a_project_bn (BatchNormal	(None, 14, 14, 112)	448	block5a_project_conv[0][0]
block5b_expand_conv (Conv2D)	(None, 14, 14, 672)	75264	block5a_project_bn[0][0]
block5b_expand_bn (BatchNormali	(None, 14, 14, 672)	2688	block5b_expand_conv[0][0]
block5b_expand_activation (Acti	(None, 14, 14, 672)	0	block5b_expand_bn[0][0]
block5b_dwconv (DepthwiseConv2D	(None, 14, 14, 672)	16800	block5b_expand_activation[0][0]
block5b_bn (BatchNormalization)	(None, 14, 14, 672)	2688	block5b_dwconv[0][0]
block5b_activation (Activation)	(None, 14, 14, 672)	0	block5b_bn[0][0]
block5b_se_squeeze (GlobalAvera	(None, 672)	0	block5b_activation[0][0]
block5b_se_reshape (Reshape)	(None, 1, 1, 672)	0	block5b_se_squeeze[0][0]
block5b_se_reduce (Conv2D)	(None, 1, 1, 28)	18844	block5b_se_reshape[0][0]
block5b_se_expand (Conv2D)	(None, 1, 1, 672)	19488	block5b_se_reduce[0][0]
block5b_se_excite (Multiply)	(None, 14, 14, 672)	0	block5b_activation[0][0] block5b_se_expand[0][0]
block5b_project_conv (Conv2D)	(None, 14, 14, 112)	75264	block5b_se_excite[0][0]
block5b_project_bn (BatchNormal	(None, 14, 14, 112)	448	block5b_project_conv[0][0]
block5b_drop (Dropout)	(None, 14, 14, 112)	0	block5b_project_bn[0][0]
block5b_add (Add)	(None, 14, 14, 112)	0	block5b_drop[0][0] block5a_project_bn[0][0]
block5c_expand_conv (Conv2D)	(None, 14, 14, 672)	75264	block5b_add[0][0]
block5c_expand_bn (BatchNormali	(None, 14, 14, 672)	2688	block5c_expand_conv[0][0]
block5c_expand_activation (Acti	(None, 14, 14, 672)	0	block5c_expand_bn[0][0]
block5c_dwconv (DepthwiseConv2D	(None, 14, 14, 672)	16800	block5c_expand_activation[0][0]
block5c_bn (BatchNormalization)	(None, 14, 14, 672)	2688	block5c_dwconv[0][0]
block5c_activation (Activation)	(None, 14, 14, 672)	0	block5c_bn[0][0]
block5c_se_squeeze (GlobalAvera	(None, 672)	0	block5c_activation[0][0]
block5c_se_reshape (Reshape)	(None, 1, 1, 672)	0	block5c_se_squeeze[0][0]
block5c_se_reduce (Conv2D)	(None, 1, 1, 28)	18844	block5c_se_reshape[0][0]
block5c_se_expand (Conv2D)	(None, 1, 1, 672)	19488	block5c_se_reduce[0][0]
block5c_se_excite (Multiply)	(None, 14, 14, 672)	0	block5c_activation[0][0] block5c_se_expand[0][0]
block5c_project_conv (Conv2D)	(None, 14, 14, 112)	75264	block5c_se_excite[0][0]
block5c_project_bn (BatchNormal	(None, 14, 14, 112)	448	block5c_project_conv[0][0]
block5c_drop (Dropout)	(None, 14, 14, 112)	0	block5c_project_bn[0][0]
block5c_add (Add)	(None, 14, 14, 112)	0	block5c_drop[0][0] block5b_add[0][0]
block6a_expand_conv (Conv2D)	(None, 14, 14, 672)	75264	block5c_add[0][0]
block6a_expand_bn (BatchNormali	(None, 14, 14, 672)	2688	block6a_expand_conv[0][0]
block6a_expand_activation (Acti	(None, 14, 14, 672)	0	block6a_expand_bn[0][0]
block6a_dwconv_pad (ZeroPadding	(None, 17, 17, 672)	0	block6a_expand_activation[0][0]
block6a_dwconv (DepthwiseConv2D	(None, 7, 7, 672)	16800	block6a_dwconv_pad[0][0]
block6a_bn (BatchNormalization)	(None, 7, 7, 672)	2688	block6a_dwconv[0][0]
block6a_activation (Activation)	(None, 7, 7, 672)	0	block6a_bn[0][0]
block6a_se_squeeze (GlobalAvera	(None, 672)	0	block6a_activation[0][0]
block6a_se_reshape (Reshape)	(None, 1, 1, 672)	0	block6a_se_squeeze[0][0]
block6a_se_reduce (Conv2D)	(None, 1, 1, 28)	18844	block6a_se_reshape[0][0]

block6a_se_expand (Conv2D)	(None, 1, 1, 672)	19488	block6a_se_reduce[0][0]
block6a_se_excite (Multiply)	(None, 7, 7, 672)	0	block6a_activation[0][0] block6a_se_expand[0][0]
block6a_project_conv (Conv2D)	(None, 7, 7, 192)	129024	block6a_se_excite[0][0]
block6a_project_bn (BatchNormal	(None, 7, 7, 192)	768	block6a_project_conv[0][0]
block6b_expand_conv (Conv2D)	(None, 7, 7, 1152)	221184	block6a_project_bn[0][0]
block6b_expand_bn (BatchNormali	(None, 7, 7, 1152)	4608	block6b_expand_conv[0][0]
block6b_expand_activation (Acti	(None, 7, 7, 1152)	0	block6b_expand_bn[0][0]
block6b_dwconv (DepthwiseConv2D	(None, 7, 7, 1152)	28800	block6b_expand_activation[0][0]
block6b_bn (BatchNormalization)	(None, 7, 7, 1152)	4608	block6b_dwconv[0][0]
block6b_activation (Activation)	(None, 7, 7, 1152)	0	block6b_bn[0][0]
block6b_se_squeeze (GlobalAvera	(None, 1152)	0	block6b_activation[0][0]
block6b_se_reshape (Reshape)	(None, 1, 1, 1152)	0	block6b_se_squeeze[0][0]
block6b_se_reduce (Conv2D)	(None, 1, 1, 48)	55344	block6b_se_reshape[0][0]
block6b_se_expand (Conv2D)	(None, 1, 1, 1152)	56448	block6b_se_reduce[0][0]
block6b_se_excite (Multiply)	(None, 7, 7, 1152)	0	block6b_activation[0][0] block6b_se_expand[0][0]
block6b_project_conv (Conv2D)	(None, 7, 7, 192)	221184	block6b_se_excite[0][0]
block6b_project_bn (BatchNormal	(None, 7, 7, 192)	768	block6b_project_conv[0][0]
block6b_drop (Dropout)	(None, 7, 7, 192)	0	block6b_project_bn[0][0]
block6b_add (Add)	(None, 7, 7, 192)	0	block6b_drop[0][0] block6a_project_bn[0][0]
block6c_expand_conv (Conv2D)	(None, 7, 7, 1152)	221184	block6b_add[0][0]
block6c_expand_bn (BatchNormali	(None, 7, 7, 1152)	4608	block6c_expand_conv[0][0]
block6c_expand_activation (Acti	(None, 7, 7, 1152)	0	block6c_expand_bn[0][0]
block6c_dwconv (DepthwiseConv2D	(None, 7, 7, 1152)	28800	block6c_expand_activation[0][0]
block6c_bn (BatchNormalization)	(None, 7, 7, 1152)	4608	block6c_dwconv[0][0]
block6c_activation (Activation)	(None, 7, 7, 1152)	0	block6c_bn[0][0]
block6c_se_squeeze (GlobalAvera	(None, 1152)	0	block6c_activation[0][0]
block6c_se_reshape (Reshape)	(None, 1, 1, 1152)	0	block6c_se_squeeze[0][0]
block6c_se_reduce (Conv2D)	(None, 1, 1, 48)	55344	block6c_se_reshape[0][0]
block6c_se_expand (Conv2D)	(None, 1, 1, 1152)	56448	block6c_se_reduce[0][0]
block6c_se_excite (Multiply)	(None, 7, 7, 1152)	0	block6c_activation[0][0] block6c_se_expand[0][0]
block6c_project_conv (Conv2D)	(None, 7, 7, 192)	221184	block6c_se_excite[0][0]
block6c_project_bn (BatchNormal	(None, 7, 7, 192)	768	block6c_project_conv[0][0]
block6c_drop (Dropout)	(None, 7, 7, 192)	0	block6c_project_bn[0][0]
block6c_add (Add)	(None, 7, 7, 192)	0	block6c_drop[0][0] block6b_add[0][0]
block6d_expand_conv (Conv2D)	(None, 7, 7, 1152)	221184	block6c_add[0][0]
block6d_expand_bn (BatchNormali	(None, 7, 7, 1152)	4608	block6d_expand_conv[0][0]
block6d_expand_activation (Acti	(None, 7, 7, 1152)	0	block6d_expand_bn[0][0]
block6d_dwconv (DepthwiseConv2D	(None, 7, 7, 1152)	28800	block6d_expand_activation[0][0]
block6d_bn (BatchNormalization)	(None, 7, 7, 1152)	4608	block6d_dwconv[0][0]
block6d_activation (Activation)	(None, 7, 7, 1152)	0	block6d_bn[0][0]
block6d_se_squeeze (GlobalAvera	(None, 1152)	0	block6d_activation[0][0]
block6d_se_reshape (Reshape)	(None, 1, 1, 1152)	0	block6d_se_squeeze[0][0]
block6d_se_reduce (Conv2D)	(None, 1, 1, 48)	55344	block6d_se_reshape[0][0]

block6d_se_expand (Conv2D)	(None, 1, 1, 1152)	56448	block6d_se_reduce[0][0]
block6d_se_excite (Multiply)	(None, 7, 7, 1152)	0	block6d_activation[0][0] block6d_se_expand[0][0]
block6d_project_conv (Conv2D)	(None, 7, 7, 192)	221184	block6d_se_excite[0][0]
block6d_project_bn (BatchNormal	(None, 7, 7, 192)	768	block6d_project_conv[0][0]
block6d_drop (Dropout)	(None, 7, 7, 192)	0	block6d_project_bn[0][0]
block6d_add (Add)	(None, 7, 7, 192)	0	block6d_drop[0][0] block6c_add[0][0]
block7a_expand_conv (Conv2D)	(None, 7, 7, 1152)	221184	block6d_add[0][0]
block7a_expand_bn (BatchNormali	(None, 7, 7, 1152)	4608	block7a_expand_conv[0][0]
block7a_expand_activation (Acti	(None, 7, 7, 1152)	0	block7a_expand_bn[0][0]
block7a_dwconv (DepthwiseConv2D	(None, 7, 7, 1152)	10368	block7a_expand_activation[0][0]
block7a_bn (BatchNormalization)	(None, 7, 7, 1152)	4608	block7a_dwconv[0][0]
block7a_activation (Activation)	(None, 7, 7, 1152)	0	block7a_bn[0][0]
block7a_se_squeeze (GlobalAvera	(None, 1152)	0	block7a_activation[0][0]
block7a_se_reshape (Reshape)	(None, 1, 1, 1152)	0	block7a_se_squeeze[0][0]
block7a_se_reduce (Conv2D)	(None, 1, 1, 48)	55344	block7a_se_reshape[0][0]
block7a_se_expand (Conv2D)	(None, 1, 1, 1152)	56448	block7a_se_reduce[0][0]
block7a_se_excite (Multiply)	(None, 7, 7, 1152)	0	block7a_activation[0][0] block7a_se_expand[0][0]
block7a_project_conv (Conv2D)	(None, 7, 7, 320)	368640	block7a_se_excite[0][0]
block7a_project_bn (BatchNormal	(None, 7, 7, 320)	1280	block7a_project_conv[0][0]
top_conv (Conv2D)	(None, 7, 7, 1280)	409600	block7a_project_bn[0][0]
top_bn (BatchNormalization)	(None, 7, 7, 1280)	5120	top_conv[0][0]
top_activation (Activation)	(None, 7, 7, 1280)	0	top_bn[0][0]
avg_pool (GlobalAveragePooling2	(None, 1280)	0	top_activation[0][0]
top_dropout (Dropout)	(None, 1280)	0	avg_pool[0][0]
predictions (Dense)	(None, 1000)	1281000	top_dropout[0][0]
=====			
Total params: 5,330,571			
Trainable params: 5,288,548			
Non-trainable params: 42,023			

Model: "functional_1"

Layer (type)	Output Shape	Param #	Connected to
=====			
input_1 (InputLayer)	[(None, 224, 224, 3)]	0	
rescaling (Rescaling)	(None, 224, 224, 3)	0	input_1[0][0]
normalization (Normalization)	(None, 224, 224, 3)	7	rescaling[0][0]
stem_conv_pad (ZeroPadding2D)	(None, 225, 225, 3)	0	normalization[0][0]
stem_conv (Conv2D)	(None, 112, 112, 32)	864	stem_conv_pad[0][0]
stem_bn (BatchNormalization)	(None, 112, 112, 32)	128	stem_conv[0][0]
stem_activation (Activation)	(None, 112, 112, 32)	0	stem_bn[0][0]
block1a_dwconv (DepthwiseConv2D	(None, 112, 112, 32)	288	stem_activation[0][0]
block1a_bn (BatchNormalization)	(None, 112, 112, 32)	128	block1a_dwconv[0][0]
block1a_activation (Activation)	(None, 112, 112, 32)	0	block1a_bn[0][0]
block1a_se_squeeze (GlobalAvera	(None, 32)	0	block1a_activation[0][0]
block1a_se_reshape (Reshape)	(None, 1, 1, 32)	0	block1a_se_squeeze[0][0]
block1a_se_reduce (Conv2D)	(None, 1, 1, 8)	264	block1a_se_reshape[0][0]
block1a_se_expand (Conv2D)	(None, 1, 1, 32)	288	block1a_se_reduce[0][0]
block1a_se_excite (Multiply)	(None, 112, 112, 32)	0	block1a_activation[0][0]

			block1a_se_expand[0][0]
block1a_project_conv (Conv2D)	(None, 112, 112, 16)	512	block1a_se_excite[0][0]
block1a_project_bn (BatchNormal	(None, 112, 112, 16)	64	block1a_project_conv[0][0]
block2a_expand_conv (Conv2D)	(None, 112, 112, 96)	1536	block1a_project_bn[0][0]
block2a_expand_bn (BatchNormali	(None, 112, 112, 96)	384	block2a_expand_conv[0][0]
block2a_expand_activation (Acti	(None, 112, 112, 96)	0	block2a_expand_bn[0][0]
block2a_dwconv_pad (ZeroPadding	(None, 113, 113, 96)	0	block2a_expand_activation[0][0]
block2a_dwconv (DepthwiseConv2D	(None, 56, 56, 96)	864	block2a_dwconv_pad[0][0]
block2a_bn (BatchNormalization)	(None, 56, 56, 96)	384	block2a_dwconv[0][0]
block2a_activation (Activation)	(None, 56, 56, 96)	0	block2a_bn[0][0]
block2a_se_squeeze (GlobalAvera	(None, 96)	0	block2a_activation[0][0]
block2a_se_reshape (Reshape)	(None, 1, 1, 96)	0	block2a_se_squeeze[0][0]
block2a_se_reduce (Conv2D)	(None, 1, 1, 4)	388	block2a_se_reshape[0][0]
block2a_se_expand (Conv2D)	(None, 1, 1, 96)	480	block2a_se_reduce[0][0]
block2a_se_excite (Multiply)	(None, 56, 56, 96)	0	block2a_activation[0][0] block2a_se_expand[0][0]
block2a_project_conv (Conv2D)	(None, 56, 56, 24)	2304	block2a_se_excite[0][0]
block2a_project_bn (BatchNormal	(None, 56, 56, 24)	96	block2a_project_conv[0][0]
block2b_expand_conv (Conv2D)	(None, 56, 56, 144)	3456	block2a_project_bn[0][0]
block2b_expand_bn (BatchNormali	(None, 56, 56, 144)	576	block2b_expand_conv[0][0]
block2b_expand_activation (Acti	(None, 56, 56, 144)	0	block2b_expand_bn[0][0]
block2b_dwconv (DepthwiseConv2D	(None, 56, 56, 144)	1296	block2b_expand_activation[0][0]
block2b_bn (BatchNormalization)	(None, 56, 56, 144)	576	block2b_dwconv[0][0]
block2b_activation (Activation)	(None, 56, 56, 144)	0	block2b_bn[0][0]
block2b_se_squeeze (GlobalAvera	(None, 144)	0	block2b_activation[0][0]
block2b_se_reshape (Reshape)	(None, 1, 1, 144)	0	block2b_se_squeeze[0][0]
block2b_se_reduce (Conv2D)	(None, 1, 1, 6)	870	block2b_se_reshape[0][0]
block2b_se_expand (Conv2D)	(None, 1, 1, 144)	1008	block2b_se_reduce[0][0]
block2b_se_excite (Multiply)	(None, 56, 56, 144)	0	block2b_activation[0][0] block2b_se_expand[0][0]
block2b_project_conv (Conv2D)	(None, 56, 56, 24)	3456	block2b_se_excite[0][0]
block2b_project_bn (BatchNormal	(None, 56, 56, 24)	96	block2b_project_conv[0][0]
block2b_drop (Dropout)	(None, 56, 56, 24)	0	block2b_project_bn[0][0]
block2b_add (Add)	(None, 56, 56, 24)	0	block2b_drop[0][0] block2a_project_bn[0][0]
block3a_expand_conv (Conv2D)	(None, 56, 56, 144)	3456	block2b_add[0][0]
block3a_expand_bn (BatchNormali	(None, 56, 56, 144)	576	block3a_expand_conv[0][0]
block3a_expand_activation (Acti	(None, 56, 56, 144)	0	block3a_expand_bn[0][0]
block3a_dwconv_pad (ZeroPadding	(None, 59, 59, 144)	0	block3a_expand_activation[0][0]
block3a_dwconv (DepthwiseConv2D	(None, 28, 28, 144)	3600	block3a_dwconv_pad[0][0]
block3a_bn (BatchNormalization)	(None, 28, 28, 144)	576	block3a_dwconv[0][0]
block3a_activation (Activation)	(None, 28, 28, 144)	0	block3a_bn[0][0]
block3a_se_squeeze (GlobalAvera	(None, 144)	0	block3a_activation[0][0]
block3a_se_reshape (Reshape)	(None, 1, 1, 144)	0	block3a_se_squeeze[0][0]
block3a_se_reduce (Conv2D)	(None, 1, 1, 6)	870	block3a_se_reshape[0][0]
block3a_se_expand (Conv2D)	(None, 1, 1, 144)	1008	block3a_se_reduce[0][0]
block3a_se_excite (Multiply)	(None, 28, 28, 144)	0	block3a_activation[0][0] block3a_se_expand[0][0]

block3a_project_conv (Conv2D)	(None, 28, 28, 40)	5760	block3a_se_excite[0][0]
block3a_project_bn (BatchNormal	(None, 28, 28, 40)	160	block3a_project_conv[0][0]
block3b_expand_conv (Conv2D)	(None, 28, 28, 240)	9600	block3a_project_bn[0][0]
block3b_expand_bn (BatchNormali	(None, 28, 28, 240)	960	block3b_expand_conv[0][0]
block3b_expand_activation (Acti	(None, 28, 28, 240)	0	block3b_expand_bn[0][0]
block3b_dwconv (DepthwiseConv2D	(None, 28, 28, 240)	6000	block3b_expand_activation[0][0]
block3b_bn (BatchNormalization)	(None, 28, 28, 240)	960	block3b_dwconv[0][0]
block3b_activation (Activation)	(None, 28, 28, 240)	0	block3b_bn[0][0]
block3b_se_squeeze (GlobalAvera	(None, 240)	0	block3b_activation[0][0]
block3b_se_reshape (Reshape)	(None, 1, 1, 240)	0	block3b_se_squeeze[0][0]
block3b_se_reduce (Conv2D)	(None, 1, 1, 10)	2410	block3b_se_reshape[0][0]
block3b_se_expand (Conv2D)	(None, 1, 1, 240)	2640	block3b_se_reduce[0][0]
block3b_se_excite (Multiply)	(None, 28, 28, 240)	0	block3b_activation[0][0] block3b_se_expand[0][0]
block3b_project_conv (Conv2D)	(None, 28, 28, 40)	9600	block3b_se_excite[0][0]
block3b_project_bn (BatchNormal	(None, 28, 28, 40)	160	block3b_project_conv[0][0]
block3b_drop (Dropout)	(None, 28, 28, 40)	0	block3b_project_bn[0][0]
block3b_add (Add)	(None, 28, 28, 40)	0	block3b_drop[0][0] block3a_project_bn[0][0]
block4a_expand_conv (Conv2D)	(None, 28, 28, 240)	9600	block3b_add[0][0]
block4a_expand_bn (BatchNormali	(None, 28, 28, 240)	960	block4a_expand_conv[0][0]
block4a_expand_activation (Acti	(None, 28, 28, 240)	0	block4a_expand_bn[0][0]
block4a_dwconv_pad (ZeroPadding	(None, 29, 29, 240)	0	block4a_expand_activation[0][0]
block4a_dwconv (DepthwiseConv2D	(None, 14, 14, 240)	2160	block4a_dwconv_pad[0][0]
block4a_bn (BatchNormalization)	(None, 14, 14, 240)	960	block4a_dwconv[0][0]
block4a_activation (Activation)	(None, 14, 14, 240)	0	block4a_bn[0][0]
block4a_se_squeeze (GlobalAvera	(None, 240)	0	block4a_activation[0][0]
block4a_se_reshape (Reshape)	(None, 1, 1, 240)	0	block4a_se_squeeze[0][0]
block4a_se_reduce (Conv2D)	(None, 1, 1, 10)	2410	block4a_se_reshape[0][0]
block4a_se_expand (Conv2D)	(None, 1, 1, 240)	2640	block4a_se_reduce[0][0]
block4a_se_excite (Multiply)	(None, 14, 14, 240)	0	block4a_activation[0][0] block4a_se_expand[0][0]
block4a_project_conv (Conv2D)	(None, 14, 14, 80)	19200	block4a_se_excite[0][0]
block4a_project_bn (BatchNormal	(None, 14, 14, 80)	320	block4a_project_conv[0][0]
block4b_expand_conv (Conv2D)	(None, 14, 14, 480)	38400	block4a_project_bn[0][0]
block4b_expand_bn (BatchNormali	(None, 14, 14, 480)	1920	block4b_expand_conv[0][0]
block4b_expand_activation (Acti	(None, 14, 14, 480)	0	block4b_expand_bn[0][0]
block4b_dwconv (DepthwiseConv2D	(None, 14, 14, 480)	4320	block4b_expand_activation[0][0]
block4b_bn (BatchNormalization)	(None, 14, 14, 480)	1920	block4b_dwconv[0][0]
block4b_activation (Activation)	(None, 14, 14, 480)	0	block4b_bn[0][0]
block4b_se_squeeze (GlobalAvera	(None, 480)	0	block4b_activation[0][0]
block4b_se_reshape (Reshape)	(None, 1, 1, 480)	0	block4b_se_squeeze[0][0]
block4b_se_reduce (Conv2D)	(None, 1, 1, 20)	9620	block4b_se_reshape[0][0]
block4b_se_expand (Conv2D)	(None, 1, 1, 480)	10080	block4b_se_reduce[0][0]
block4b_se_excite (Multiply)	(None, 14, 14, 480)	0	block4b_activation[0][0] block4b_se_expand[0][0]
block4b_project_conv (Conv2D)	(None, 14, 14, 80)	38400	block4b_se_excite[0][0]

block4b_project_bn (BatchNormal	(None, 14, 14, 80)	320	block4b_project_conv[0][0]
block4b_drop (Dropout)	(None, 14, 14, 80)	0	block4b_project_bn[0][0]
block4b_add (Add)	(None, 14, 14, 80)	0	block4b_drop[0][0] block4a_project_bn[0][0]
block4c_expand_conv (Conv2D)	(None, 14, 14, 480)	38400	block4b_add[0][0]
block4c_expand_bn (BatchNormali	(None, 14, 14, 480)	1920	block4c_expand_conv[0][0]
block4c_expand_activation (Acti	(None, 14, 14, 480)	0	block4c_expand_bn[0][0]
block4c_dwconv (DepthwiseConv2D	(None, 14, 14, 480)	4320	block4c_expand_activation[0][0]
block4c_bn (BatchNormalization)	(None, 14, 14, 480)	1920	block4c_dwconv[0][0]
block4c_activation (Activation)	(None, 14, 14, 480)	0	block4c_bn[0][0]
block4c_se_squeeze (GlobalAvera	(None, 480)	0	block4c_activation[0][0]
block4c_se_reshape (Reshape)	(None, 1, 1, 480)	0	block4c_se_squeeze[0][0]
block4c_se_reduce (Conv2D)	(None, 1, 1, 20)	9620	block4c_se_reshape[0][0]
block4c_se_expand (Conv2D)	(None, 1, 1, 480)	10080	block4c_se_reduce[0][0]
block4c_se_excite (Multiply)	(None, 14, 14, 480)	0	block4c_activation[0][0] block4c_se_expand[0][0]
block4c_project_conv (Conv2D)	(None, 14, 14, 80)	38400	block4c_se_excite[0][0]
block4c_project_bn (BatchNormal	(None, 14, 14, 80)	320	block4c_project_conv[0][0]
block4c_drop (Dropout)	(None, 14, 14, 80)	0	block4c_project_bn[0][0]
block4c_add (Add)	(None, 14, 14, 80)	0	block4c_drop[0][0] block4b_add[0][0]
block5a_expand_conv (Conv2D)	(None, 14, 14, 480)	38400	block4c_add[0][0]
block5a_expand_bn (BatchNormali	(None, 14, 14, 480)	1920	block5a_expand_conv[0][0]
block5a_expand_activation (Acti	(None, 14, 14, 480)	0	block5a_expand_bn[0][0]
block5a_dwconv (DepthwiseConv2D	(None, 14, 14, 480)	12000	block5a_expand_activation[0][0]
block5a_bn (BatchNormalization)	(None, 14, 14, 480)	1920	block5a_dwconv[0][0]
block5a_activation (Activation)	(None, 14, 14, 480)	0	block5a_bn[0][0]
block5a_se_squeeze (GlobalAvera	(None, 480)	0	block5a_activation[0][0]
block5a_se_reshape (Reshape)	(None, 1, 1, 480)	0	block5a_se_squeeze[0][0]
block5a_se_reduce (Conv2D)	(None, 1, 1, 20)	9620	block5a_se_reshape[0][0]
block5a_se_expand (Conv2D)	(None, 1, 1, 480)	10080	block5a_se_reduce[0][0]
block5a_se_excite (Multiply)	(None, 14, 14, 480)	0	block5a_activation[0][0] block5a_se_expand[0][0]
block5a_project_conv (Conv2D)	(None, 14, 14, 112)	53760	block5a_se_excite[0][0]
block5a_project_bn (BatchNormal	(None, 14, 14, 112)	448	block5a_project_conv[0][0]
block5b_expand_conv (Conv2D)	(None, 14, 14, 672)	75264	block5a_project_bn[0][0]
block5b_expand_bn (BatchNormali	(None, 14, 14, 672)	2688	block5b_expand_conv[0][0]
block5b_expand_activation (Acti	(None, 14, 14, 672)	0	block5b_expand_bn[0][0]
block5b_dwconv (DepthwiseConv2D	(None, 14, 14, 672)	16800	block5b_expand_activation[0][0]
block5b_bn (BatchNormalization)	(None, 14, 14, 672)	2688	block5b_dwconv[0][0]
block5b_activation (Activation)	(None, 14, 14, 672)	0	block5b_bn[0][0]
block5b_se_squeeze (GlobalAvera	(None, 672)	0	block5b_activation[0][0]
block5b_se_reshape (Reshape)	(None, 1, 1, 672)	0	block5b_se_squeeze[0][0]
block5b_se_reduce (Conv2D)	(None, 1, 1, 28)	18844	block5b_se_reshape[0][0]
block5b_se_expand (Conv2D)	(None, 1, 1, 672)	19488	block5b_se_reduce[0][0]
block5b_se_excite (Multiply)	(None, 14, 14, 672)	0	block5b_activation[0][0] block5b_se_expand[0][0]
block5b_project_conv (Conv2D)	(None, 14, 14, 112)	75264	block5b_se_excite[0][0]

block5b_project_bn (BatchNormal	(None, 14, 14, 112)	448	block5b_project_conv[0][0]
block5b_drop (Dropout)	(None, 14, 14, 112)	0	block5b_project_bn[0][0]
block5b_add (Add)	(None, 14, 14, 112)	0	block5b_drop[0][0] block5a_project_bn[0][0]
block5c_expand_conv (Conv2D)	(None, 14, 14, 672)	75264	block5b_add[0][0]
block5c_expand_bn (BatchNormali	(None, 14, 14, 672)	2688	block5c_expand_conv[0][0]
block5c_expand_activation (Acti	(None, 14, 14, 672)	0	block5c_expand_bn[0][0]
block5c_dwconv (DepthwiseConv2D	(None, 14, 14, 672)	16800	block5c_expand_activation[0][0]
block5c_bn (BatchNormalization)	(None, 14, 14, 672)	2688	block5c_dwconv[0][0]
block5c_activation (Activation)	(None, 14, 14, 672)	0	block5c_bn[0][0]
block5c_se_squeeze (GlobalAvera	(None, 672)	0	block5c_activation[0][0]
block5c_se_reshape (Reshape)	(None, 1, 1, 672)	0	block5c_se_squeeze[0][0]
block5c_se_reduce (Conv2D)	(None, 1, 1, 28)	18844	block5c_se_reshape[0][0]
block5c_se_expand (Conv2D)	(None, 1, 1, 672)	19488	block5c_se_reduce[0][0]
block5c_se_excite (Multiply)	(None, 14, 14, 672)	0	block5c_activation[0][0] block5c_se_expand[0][0]
block5c_project_conv (Conv2D)	(None, 14, 14, 112)	75264	block5c_se_excite[0][0]
block5c_project_bn (BatchNormal	(None, 14, 14, 112)	448	block5c_project_conv[0][0]
block5c_drop (Dropout)	(None, 14, 14, 112)	0	block5c_project_bn[0][0]
block5c_add (Add)	(None, 14, 14, 112)	0	block5c_drop[0][0] block5b_add[0][0]
block6a_expand_conv (Conv2D)	(None, 14, 14, 672)	75264	block5c_add[0][0]
block6a_expand_bn (BatchNormali	(None, 14, 14, 672)	2688	block6a_expand_conv[0][0]
block6a_expand_activation (Acti	(None, 14, 14, 672)	0	block6a_expand_bn[0][0]
block6a_dwconv_pad (ZeroPadding	(None, 17, 17, 672)	0	block6a_expand_activation[0][0]
block6a_dwconv (DepthwiseConv2D	(None, 7, 7, 672)	16800	block6a_dwconv_pad[0][0]
block6a_bn (BatchNormalization)	(None, 7, 7, 672)	2688	block6a_dwconv[0][0]
block6a_activation (Activation)	(None, 7, 7, 672)	0	block6a_bn[0][0]
block6a_se_squeeze (GlobalAvera	(None, 672)	0	block6a_activation[0][0]
block6a_se_reshape (Reshape)	(None, 1, 1, 672)	0	block6a_se_squeeze[0][0]
block6a_se_reduce (Conv2D)	(None, 1, 1, 28)	18844	block6a_se_reshape[0][0]
block6a_se_expand (Conv2D)	(None, 1, 1, 672)	19488	block6a_se_reduce[0][0]
block6a_se_excite (Multiply)	(None, 7, 7, 672)	0	block6a_activation[0][0] block6a_se_expand[0][0]
block6a_project_conv (Conv2D)	(None, 7, 7, 192)	129024	block6a_se_excite[0][0]
block6a_project_bn (BatchNormal	(None, 7, 7, 192)	768	block6a_project_conv[0][0]
block6b_expand_conv (Conv2D)	(None, 7, 7, 1152)	221184	block6a_project_bn[0][0]
block6b_expand_bn (BatchNormali	(None, 7, 7, 1152)	4608	block6b_expand_conv[0][0]
block6b_expand_activation (Acti	(None, 7, 7, 1152)	0	block6b_expand_bn[0][0]
block6b_dwconv (DepthwiseConv2D	(None, 7, 7, 1152)	28800	block6b_expand_activation[0][0]
block6b_bn (BatchNormalization)	(None, 7, 7, 1152)	4608	block6b_dwconv[0][0]
block6b_activation (Activation)	(None, 7, 7, 1152)	0	block6b_bn[0][0]
block6b_se_squeeze (GlobalAvera	(None, 1152)	0	block6b_activation[0][0]
block6b_se_reshape (Reshape)	(None, 1, 1, 1152)	0	block6b_se_squeeze[0][0]
block6b_se_reduce (Conv2D)	(None, 1, 1, 48)	55344	block6b_se_reshape[0][0]
block6b_se_expand (Conv2D)	(None, 1, 1, 1152)	56448	block6b_se_reduce[0][0]
block6b_se_excite (Multiply)	(None, 7, 7, 1152)	0	block6b_activation[0][0] block6b_se_expand[0][0]

block6b_project_conv (Conv2D)	(None, 7, 7, 192)	221184	block6b_se_excite[0][0]
block6b_project_bn (BatchNormal	(None, 7, 7, 192)	768	block6b_project_conv[0][0]
block6b_drop (Dropout)	(None, 7, 7, 192)	0	block6b_project_bn[0][0]
block6b_add (Add)	(None, 7, 7, 192)	0	block6b_drop[0][0] block6a_project_bn[0][0]
block6c_expand_conv (Conv2D)	(None, 7, 7, 1152)	221184	block6b_add[0][0]
block6c_expand_bn (BatchNormali	(None, 7, 7, 1152)	4608	block6c_expand_conv[0][0]
block6c_expand_activation (Acti	(None, 7, 7, 1152)	0	block6c_expand_bn[0][0]
block6c_dwconv (DepthwiseConv2D	(None, 7, 7, 1152)	28800	block6c_expand_activation[0][0]
block6c_bn (BatchNormalization)	(None, 7, 7, 1152)	4608	block6c_dwconv[0][0]
block6c_activation (Activation)	(None, 7, 7, 1152)	0	block6c_bn[0][0]
block6c_se_squeeze (GlobalAvera	(None, 1152)	0	block6c_activation[0][0]
block6c_se_reshape (Reshape)	(None, 1, 1, 1152)	0	block6c_se_squeeze[0][0]
block6c_se_reduce (Conv2D)	(None, 1, 1, 48)	55344	block6c_se_reshape[0][0]
block6c_se_expand (Conv2D)	(None, 1, 1, 1152)	56448	block6c_se_reduce[0][0]
block6c_se_excite (Multiply)	(None, 7, 7, 1152)	0	block6c_activation[0][0] block6c_se_expand[0][0]
block6c_project_conv (Conv2D)	(None, 7, 7, 192)	221184	block6c_se_excite[0][0]
block6c_project_bn (BatchNormal	(None, 7, 7, 192)	768	block6c_project_conv[0][0]
block6c_drop (Dropout)	(None, 7, 7, 192)	0	block6c_project_bn[0][0]
block6c_add (Add)	(None, 7, 7, 192)	0	block6c_drop[0][0] block6b_add[0][0]
block6d_expand_conv (Conv2D)	(None, 7, 7, 1152)	221184	block6c_add[0][0]
block6d_expand_bn (BatchNormali	(None, 7, 7, 1152)	4608	block6d_expand_conv[0][0]
block6d_expand_activation (Acti	(None, 7, 7, 1152)	0	block6d_expand_bn[0][0]
block6d_dwconv (DepthwiseConv2D	(None, 7, 7, 1152)	28800	block6d_expand_activation[0][0]
block6d_bn (BatchNormalization)	(None, 7, 7, 1152)	4608	block6d_dwconv[0][0]
block6d_activation (Activation)	(None, 7, 7, 1152)	0	block6d_bn[0][0]
block6d_se_squeeze (GlobalAvera	(None, 1152)	0	block6d_activation[0][0]
block6d_se_reshape (Reshape)	(None, 1, 1, 1152)	0	block6d_se_squeeze[0][0]
block6d_se_reduce (Conv2D)	(None, 1, 1, 48)	55344	block6d_se_reshape[0][0]
block6d_se_expand (Conv2D)	(None, 1, 1, 1152)	56448	block6d_se_reduce[0][0]
block6d_se_excite (Multiply)	(None, 7, 7, 1152)	0	block6d_activation[0][0] block6d_se_expand[0][0]
block6d_project_conv (Conv2D)	(None, 7, 7, 192)	221184	block6d_se_excite[0][0]
block6d_project_bn (BatchNormal	(None, 7, 7, 192)	768	block6d_project_conv[0][0]
block6d_drop (Dropout)	(None, 7, 7, 192)	0	block6d_project_bn[0][0]
block6d_add (Add)	(None, 7, 7, 192)	0	block6d_drop[0][0] block6c_add[0][0]
block7a_expand_conv (Conv2D)	(None, 7, 7, 1152)	221184	block6d_add[0][0]
block7a_expand_bn (BatchNormali	(None, 7, 7, 1152)	4608	block7a_expand_conv[0][0]
block7a_expand_activation (Acti	(None, 7, 7, 1152)	0	block7a_expand_bn[0][0]
block7a_dwconv (DepthwiseConv2D	(None, 7, 7, 1152)	10368	block7a_expand_activation[0][0]
block7a_bn (BatchNormalization)	(None, 7, 7, 1152)	4608	block7a_dwconv[0][0]
block7a_activation (Activation)	(None, 7, 7, 1152)	0	block7a_bn[0][0]
block7a_se_squeeze (GlobalAvera	(None, 1152)	0	block7a_activation[0][0]
block7a_se_reshape (Reshape)	(None, 1, 1, 1152)	0	block7a_se_squeeze[0][0]
block7a_se_reduce (Conv2D)	(None, 1, 1, 48)	55344	block7a_se_reshape[0][0]

block7a_se_expand (Conv2D)	(None, 1, 1, 1152)	56448	block7a_se_reduce[0][0]
block7a_se_excite (Multiply)	(None, 7, 7, 1152)	0	block7a_activation[0][0] block7a_se_expand[0][0]
block7a_project_conv (Conv2D)	(None, 7, 7, 320)	368640	block7a_se_excite[0][0]
block7a_project_bn (BatchNormal	(None, 7, 7, 320)	1280	block7a_project_conv[0][0]
top_conv (Conv2D)	(None, 7, 7, 1280)	409600	block7a_project_bn[0][0]
top_bn (BatchNormalization)	(None, 7, 7, 1280)	5120	top_conv[0][0]
top_activation (Activation)	(None, 7, 7, 1280)	0	top_bn[0][0]
avg_pool (GlobalAveragePooling2	(None, 1280)	0	top_activation[0][0]
flatten (Flatten)	(None, 1280)	0	avg_pool[0][0]
output_layer (Dense)	(None, 2)	2562	flatten[0][0]
=====			
Total params: 4,052,133			
Trainable params: 4,010,110			
Non-trainable params: 42,023			

```
In [12]: for layer in custom_resnet_model.layers[:-1]:
          layer.trainable = False

          custom_resnet_model.layers[-1].trainable
```

Out[12]: True

```
In [13]: custom_resnet_model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

```
In [14]: t=time.time()
hist = custom_resnet_model.fit(X_train, y_train, batch_size=32, epochs=num_epoch, verbose=1, validation_data=(X_test, y_test))
print('Training time: %s' % (t - time.time()))
(loss, accuracy) = custom_resnet_model.evaluate(X_test, y_test, batch_size=10, verbose=1)
print("[INFO] loss={:.4f}, accuracy: {:.4f}%".format(loss, accuracy * 100))
```

Epoch 1/100
147/147 [=====] - 10s 67ms/step - loss: 0.2773 - accuracy: 0.8845 - val_loss: 0.1604 - val_accuracy: 0.9403
Epoch 2/100
147/147 [=====] - 8s 56ms/step - loss: 0.1834 - accuracy: 0.9313 - val_loss: 0.1341 - val_accuracy: 0.9497
Epoch 3/100
147/147 [=====] - 8s 57ms/step - loss: 0.1541 - accuracy: 0.9443 - val_loss: 0.1338 - val_accuracy: 0.9514
Epoch 4/100
147/147 [=====] - 8s 57ms/step - loss: 0.1407 - accuracy: 0.9498 - val_loss: 0.1146 - val_accuracy: 0.9548
Epoch 5/100
147/147 [=====] - 8s 57ms/step - loss: 0.1303 - accuracy: 0.9522 - val_loss: 0.1142 - val_accuracy: 0.9590
Epoch 6/100
147/147 [=====] - 8s 57ms/step - loss: 0.1272 - accuracy: 0.9537 - val_loss: 0.1158 - val_accuracy: 0.9608
Epoch 7/100
147/147 [=====] - 8s 57ms/step - loss: 0.1209 - accuracy: 0.9573 - val_loss: 0.1075 - val_accuracy: 0.9608
Epoch 8/100
147/147 [=====] - 8s 57ms/step - loss: 0.1171 - accuracy: 0.9586 - val_loss: 0.1077 - val_accuracy: 0.9616
Epoch 9/100
147/147 [=====] - 8s 57ms/step - loss: 0.1102 - accuracy: 0.9571 - val_loss: 0.1019 - val_accuracy: 0.9616
Epoch 10/100
147/147 [=====] - 8s 57ms/step - loss: 0.1122 - accuracy: 0.9582 - val_loss: 0.1016 - val_accuracy: 0.9616
Epoch 11/100
147/147 [=====] - 8s 57ms/step - loss: 0.1082 - accuracy: 0.9596 - val_loss: 0.1034 - val_accuracy: 0.9625
Epoch 12/100
147/147 [=====] - 8s 57ms/step - loss: 0.1036 - accuracy: 0.9643 - val_loss: 0.0983 - val_accuracy: 0.9642
Epoch 13/100
147/147 [=====] - 8s 57ms/step - loss: 0.1028 - accuracy: 0.9607 - val_loss: 0.1088 - val_accuracy: 0.9573
Epoch 14/100
147/147 [=====] - 8s 57ms/step - loss: 0.1038 - accuracy: 0.9637 - val_loss: 0.1113 - val_accuracy: 0.9582
Epoch 15/100
147/147 [=====] - 8s 57ms/step - loss: 0.0960 - accuracy: 0.9667 - val_loss: 0.0955 - val_accuracy: 0.9650
Epoch 16/100
147/147 [=====] - 8s 57ms/step - loss: 0.0958 - accuracy: 0.9654 - val_loss: 0.0941 - val_accuracy: 0.96670.0
966 - accura
Epoch 17/100
147/147 [=====] - 8s 57ms/step - loss: 0.0946 - accuracy: 0.9654 - val_loss: 0.0990 - val_accuracy: 0.9616
Epoch 18/100
147/147 [=====] - 8s 56ms/step - loss: 0.0953 - accuracy: 0.9661 - val_loss: 0.0937 - val_accuracy: 0.9659
Epoch 19/100
147/147 [=====] - 8s 57ms/step - loss: 0.0896 - accuracy: 0.9686 - val_loss: 0.0941 - val_accuracy: 0.9684
Epoch 20/100
147/147 [=====] - 8s 57ms/step - loss: 0.0922 - accuracy: 0.9637 - val_loss: 0.0929 - val_accuracy: 0.9667
Epoch 21/100
147/147 [=====] - 8s 57ms/step - loss: 0.0914 - accuracy: 0.9641 - val_loss: 0.0918 - val_accuracy: 0.9684
Epoch 22/100
147/147 [=====] - 8s 57ms/step - loss: 0.0900 - accuracy: 0.9690 - val_loss: 0.0917 - val_accuracy: 0.9684
Epoch 23/100
147/147 [=====] - 8s 57ms/step - loss: 0.0860 - accuracy: 0.9673 - val_loss: 0.0922 - val_accuracy: 0.9676
Epoch 24/100
147/147 [=====] - 8s 57ms/step - loss: 0.0893 - accuracy: 0.9673 - val_loss: 0.0908 - val_accuracy: 0.9659
Epoch 25/100
147/147 [=====] - 8s 57ms/step - loss: 0.0885 - accuracy: 0.9684 - val_loss: 0.0911 - val_accuracy: 0.9676
Epoch 26/100
147/147 [=====] - 8s 57ms/step - loss: 0.0859 - accuracy: 0.9675 - val_loss: 0.0901 - val_accuracy: 0.9659
Epoch 27/100
147/147 [=====] - 8s 57ms/step - loss: 0.0849 - accuracy: 0.9695 - val_loss: 0.0925 - val_accuracy: 0.9676
Epoch 28/100
147/147 [=====] - 8s 57ms/step - loss: 0.0816 - accuracy: 0.9695 - val_loss: 0.0898 - val_accuracy: 0.9701
Epoch 29/100
147/147 [=====] - 8s 57ms/step - loss: 0.0813 - accuracy: 0.9688 - val_loss: 0.0890 - val_accuracy: 0.9650
Epoch 30/100
147/147 [=====] - 8s 57ms/step - loss: 0.0780 - accuracy: 0.9725 - val_loss: 0.0886 - val_accuracy: 0.9684
Epoch 31/100
147/147 [=====] - 8s 57ms/step - loss: 0.0832 - accuracy: 0.9684 - val_loss: 0.0878 - val_accuracy: 0.9676
Epoch 32/100
147/147 [=====] - 8s 57ms/step - loss: 0.0806 - accuracy: 0.9729 - val_loss: 0.0886 - val_accuracy: 0.9667
Epoch 33/100
147/147 [=====] - 8s 57ms/step - loss: 0.0810 - accuracy: 0.9699 - val_loss: 0.0874 - val_accuracy: 0.9676
Epoch 34/100
147/147 [=====] - 8s 57ms/step - loss: 0.0794 - accuracy: 0.9686 - val_loss: 0.0884 - val_accuracy: 0.9667
Epoch 35/100
147/147 [=====] - 8s 57ms/step - loss: 0.0787 - accuracy: 0.9701 - val_loss: 0.0955 - val_accuracy: 0.9667
Epoch 36/100
147/147 [=====] - 8s 57ms/step - loss: 0.0741 - accuracy: 0.9729 - val_loss: 0.0895 - val_accuracy: 0.9684
Epoch 37/100
147/147 [=====] - 8s 57ms/step - loss: 0.0763 - accuracy: 0.9688 - val_loss: 0.0881 - val_accuracy: 0.9693
Epoch 38/100
147/147 [=====] - 8s 57ms/step - loss: 0.0788 - accuracy: 0.9720 - val_loss: 0.0945 - val_accuracy: 0.9676
Epoch 39/100
147/147 [=====] - 8s 57ms/step - loss: 0.0775 - accuracy: 0.9725 - val_loss: 0.0891 - val_accuracy: 0.9693
Epoch 40/100
147/147 [=====] - 8s 57ms/step - loss: 0.0775 - accuracy: 0.9708 - val_loss: 0.0882 - val_accuracy: 0.9684
Epoch 41/100
147/147 [=====] - 8s 57ms/step - loss: 0.0786 - accuracy: 0.9710 - val_loss: 0.0867 - val_accuracy: 0.9667
Epoch 42/100
147/147 [=====] - 8s 56ms/step - loss: 0.0738 - accuracy: 0.9752 - val_loss: 0.0855 - val_accuracy: 0.9676
Epoch 43/100
147/147 [=====] - 8s 57ms/step - loss: 0.0762 - accuracy: 0.9714 - val_loss: 0.0879 - val_accuracy: 0.9693
Epoch 44/100
147/147 [=====] - 8s 57ms/step - loss: 0.0764 - accuracy: 0.9710 - val_loss: 0.0871 - val_accuracy: 0.9667
Epoch 45/100
147/147 [=====] - 8s 57ms/step - loss: 0.0710 - accuracy: 0.9761 - val_loss: 0.0897 - val_accuracy: 0.9710

Epoch 46/100
147/147 [=====] - 8s 57ms/step - loss: 0.0732 - accuracy: 0.9725 - val_loss: 0.0878 - val_accuracy: 0.9676
Epoch 47/100
147/147 [=====] - 8s 57ms/step - loss: 0.0690 - accuracy: 0.9720 - val_loss: 0.0878 - val_accuracy: 0.9676
Epoch 48/100
147/147 [=====] - 8s 56ms/step - loss: 0.0708 - accuracy: 0.9746 - val_loss: 0.0866 - val_accuracy: 0.9676
Epoch 49/100
147/147 [=====] - 8s 57ms/step - loss: 0.0711 - accuracy: 0.9746 - val_loss: 0.0878 - val_accuracy: 0.9684
Epoch 50/100
147/147 [=====] - 8s 57ms/step - loss: 0.0692 - accuracy: 0.9754 - val_loss: 0.0878 - val_accuracy: 0.9667
Epoch 51/100
147/147 [=====] - 8s 57ms/step - loss: 0.0705 - accuracy: 0.9765 - val_loss: 0.0877 - val_accuracy: 0.9684
Epoch 52/100
147/147 [=====] - 8s 57ms/step - loss: 0.0695 - accuracy: 0.9729 - val_loss: 0.0917 - val_accuracy: 0.9684
Epoch 53/100
147/147 [=====] - 8s 57ms/step - loss: 0.0711 - accuracy: 0.9740 - val_loss: 0.0931 - val_accuracy: 0.9676
Epoch 54/100
147/147 [=====] - 8s 57ms/step - loss: 0.0681 - accuracy: 0.9735 - val_loss: 0.0887 - val_accuracy: 0.9693
Epoch 55/100
147/147 [=====] - 8s 57ms/step - loss: 0.0703 - accuracy: 0.9733 - val_loss: 0.0948 - val_accuracy: 0.9667
Epoch 56/100
147/147 [=====] - 8s 57ms/step - loss: 0.0659 - accuracy: 0.9750 - val_loss: 0.0872 - val_accuracy: 0.9676
Epoch 57/100
147/147 [=====] - 8s 57ms/step - loss: 0.0700 - accuracy: 0.9740 - val_loss: 0.0922 - val_accuracy: 0.9676
Epoch 58/100
147/147 [=====] - 8s 56ms/step - loss: 0.0691 - accuracy: 0.9735 - val_loss: 0.0884 - val_accuracy: 0.9676
Epoch 59/100
147/147 [=====] - 8s 57ms/step - loss: 0.0685 - accuracy: 0.9737 - val_loss: 0.0883 - val_accuracy: 0.9693
Epoch 60/100
147/147 [=====] - 8s 57ms/step - loss: 0.0670 - accuracy: 0.9761 - val_loss: 0.0884 - val_accuracy: 0.9693
Epoch 61/100
147/147 [=====] - 8s 57ms/step - loss: 0.0713 - accuracy: 0.9750 - val_loss: 0.0867 - val_accuracy: 0.9693
Epoch 62/100
147/147 [=====] - 8s 57ms/step - loss: 0.0664 - accuracy: 0.9750 - val_loss: 0.0859 - val_accuracy: 0.9693
Epoch 63/100
147/147 [=====] - 8s 57ms/step - loss: 0.0686 - accuracy: 0.9744 - val_loss: 0.0912 - val_accuracy: 0.9693
Epoch 64/100
147/147 [=====] - 8s 57ms/step - loss: 0.0666 - accuracy: 0.9737 - val_loss: 0.0882 - val_accuracy: 0.9693
Epoch 65/100
147/147 [=====] - 8s 56ms/step - loss: 0.0654 - accuracy: 0.9767 - val_loss: 0.0869 - val_accuracy: 0.9693
Epoch 66/100
147/147 [=====] - 8s 57ms/step - loss: 0.0653 - accuracy: 0.9757 - val_loss: 0.0887 - val_accuracy: 0.9676
Epoch 67/100
147/147 [=====] - 8s 57ms/step - loss: 0.0650 - accuracy: 0.9761 - val_loss: 0.0861 - val_accuracy: 0.9693
Epoch 68/100
147/147 [=====] - 8s 56ms/step - loss: 0.0632 - accuracy: 0.9754 - val_loss: 0.0858 - val_accuracy: 0.9693
Epoch 69/100
147/147 [=====] - 8s 57ms/step - loss: 0.0640 - accuracy: 0.9763 - val_loss: 0.0855 - val_accuracy: 0.9693
Epoch 70/100
147/147 [=====] - 8s 57ms/step - loss: 0.0638 - accuracy: 0.9787 - val_loss: 0.0867 - val_accuracy: 0.9693
Epoch 71/100
147/147 [=====] - 8s 57ms/step - loss: 0.0659 - accuracy: 0.9757 - val_loss: 0.0881 - val_accuracy: 0.9701
Epoch 72/100
147/147 [=====] - 8s 57ms/step - loss: 0.0646 - accuracy: 0.9765 - val_loss: 0.0877 - val_accuracy: 0.9676
Epoch 73/100
147/147 [=====] - 8s 56ms/step - loss: 0.0662 - accuracy: 0.9759 - val_loss: 0.0867 - val_accuracy: 0.9676
Epoch 74/100
147/147 [=====] - 8s 57ms/step - loss: 0.0638 - accuracy: 0.9767 - val_loss: 0.0866 - val_accuracy: 0.9684
Epoch 75/100
147/147 [=====] - 8s 57ms/step - loss: 0.0615 - accuracy: 0.9776 - val_loss: 0.0862 - val_accuracy: 0.9701
Epoch 76/100
147/147 [=====] - 8s 57ms/step - loss: 0.0630 - accuracy: 0.9761 - val_loss: 0.0871 - val_accuracy: 0.9693
Epoch 77/100
147/147 [=====] - 8s 56ms/step - loss: 0.0627 - accuracy: 0.9757 - val_loss: 0.0865 - val_accuracy: 0.9710
Epoch 78/100
147/147 [=====] - 8s 57ms/step - loss: 0.0596 - accuracy: 0.9784 - val_loss: 0.0871 - val_accuracy: 0.9693
Epoch 79/100
147/147 [=====] - 8s 56ms/step - loss: 0.0629 - accuracy: 0.9776 - val_loss: 0.0907 - val_accuracy: 0.9710
Epoch 80/100
147/147 [=====] - 8s 57ms/step - loss: 0.0580 - accuracy: 0.9784 - val_loss: 0.0882 - val_accuracy: 0.9684
Epoch 81/100
147/147 [=====] - 8s 57ms/step - loss: 0.0616 - accuracy: 0.9778 - val_loss: 0.0867 - val_accuracy: 0.9693
Epoch 82/100
147/147 [=====] - 8s 57ms/step - loss: 0.0600 - accuracy: 0.9791 - val_loss: 0.0879 - val_accuracy: 0.9701
Epoch 83/100
147/147 [=====] - 8s 57ms/step - loss: 0.0642 - accuracy: 0.9746 - val_loss: 0.0984 - val_accuracy: 0.9693
Epoch 84/100
147/147 [=====] - 8s 57ms/step - loss: 0.0605 - accuracy: 0.9772 - val_loss: 0.0921 - val_accuracy: 0.9676
Epoch 85/100
147/147 [=====] - 8s 57ms/step - loss: 0.0628 - accuracy: 0.9744 - val_loss: 0.0873 - val_accuracy: 0.9701
Epoch 86/100
147/147 [=====] - 8s 57ms/step - loss: 0.0612 - accuracy: 0.9759 - val_loss: 0.0881 - val_accuracy: 0.9701
Epoch 87/100
147/147 [=====] - 8s 57ms/step - loss: 0.0602 - accuracy: 0.9784 - val_loss: 0.0871 - val_accuracy: 0.9684
Epoch 88/100
147/147 [=====] - 8s 57ms/step - loss: 0.0629 - accuracy: 0.9754 - val_loss: 0.0866 - val_accuracy: 0.9693
Epoch 89/100
147/147 [=====] - 8s 57ms/step - loss: 0.0575 - accuracy: 0.9793 - val_loss: 0.0867 - val_accuracy: 0.9701
Epoch 90/100
147/147 [=====] - 8s 57ms/step - loss: 0.0597 - accuracy: 0.9782 - val_loss: 0.0901 - val_accuracy: 0.9710
Epoch 91/100

```

147/147 [=====] - 8s 57ms/step - loss: 0.0622 - accuracy: 0.9778 - val_loss: 0.0878 - val_accuracy: 0.9693
Epoch 92/100
147/147 [=====] - 8s 57ms/step - loss: 0.0613 - accuracy: 0.9782 - val_loss: 0.0872 - val_accuracy: 0.9701
Epoch 93/100
147/147 [=====] - 8s 57ms/step - loss: 0.0587 - accuracy: 0.9774 - val_loss: 0.0872 - val_accuracy: 0.9684
Epoch 94/100
147/147 [=====] - 8s 57ms/step - loss: 0.0577 - accuracy: 0.9782 - val_loss: 0.0911 - val_accuracy: 0.9718
Epoch 95/100
147/147 [=====] - 8s 57ms/step - loss: 0.0626 - accuracy: 0.9787 - val_loss: 0.0897 - val_accuracy: 0.9710
Epoch 96/100
147/147 [=====] - 8s 57ms/step - loss: 0.0537 - accuracy: 0.9808 - val_loss: 0.0885 - val_accuracy: 0.9701
Epoch 97/100
147/147 [=====] - 8s 57ms/step - loss: 0.0577 - accuracy: 0.9780 - val_loss: 0.0902 - val_accuracy: 0.9684
Epoch 98/100
147/147 [=====] - 8s 57ms/step - loss: 0.0599 - accuracy: 0.9765 - val_loss: 0.0895 - val_accuracy: 0.9710
Epoch 99/100
147/147 [=====] - 8s 57ms/step - loss: 0.0576 - accuracy: 0.9789 - val_loss: 0.0933 - val_accuracy: 0.9676
Epoch 100/100
147/147 [=====] - 8s 57ms/step - loss: 0.0555 - accuracy: 0.9791 - val_loss: 0.0901 - val_accuracy: 0.9693
Training time: -848.8196833133698
118/118 [=====] - 3s 25ms/step - loss: 0.0901 - accuracy: 0.9693
[INFO] loss=0.0901, accuracy: 96.9283%

```

```

In [15]: (loss, accuracy) = custom_resnet_model.evaluate(X_test, y_test, batch_size=10, verbose=1)

print("[INFO] loss={:.4f}, accuracy: {:.4f}%".format(loss, accuracy * 100))

118/118 [=====] - 3s 22ms/step - loss: 0.0901 - accuracy: 0.9693
[INFO] loss=0.0901, accuracy: 96.9283%

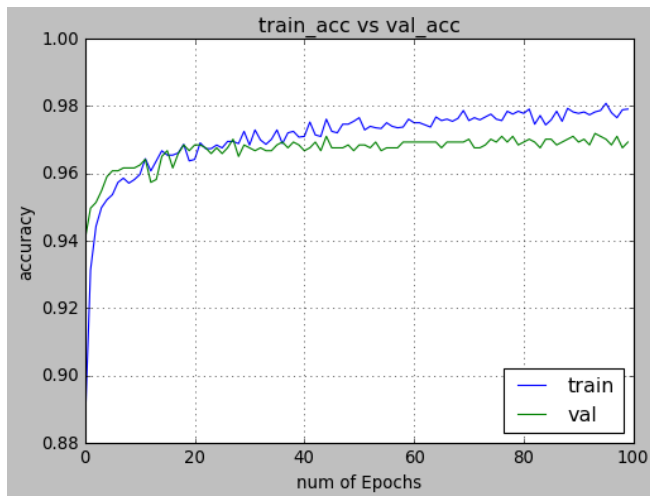
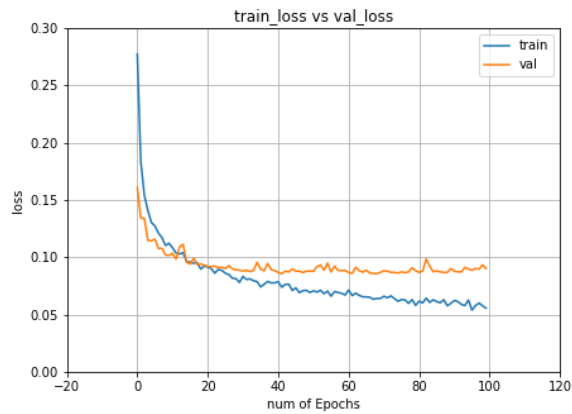
```

visualizing losses and accuracy

```

In [16]: display_loss_accuracy(hist)

```



```
In [11]: image_input = Input(shape=(224, 224, 3))
model = MobileNetV2(input_tensor=image_input, include_top=True, weights='imagenet')
model.summary()
output_layer = model(image_input)

# Last_Layer = model.get_layer('avg_pool').output
x= Flatten(name='flatten')(output_layer)
out = Dense(num_classes, activation='softmax', name='output_layer')(x)
custom_resnet_model = Model(inputs=image_input, outputs= out)
custom_resnet_model.summary()
```


Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/mobilenet_v2/mobilenet_v2_weights_tf_dim_ordering_tf_kernels_1.0_224.h5
 14540800/14536120 [=====] - 1s 0us/step
 Model: "mobilenetv2_1.00_224"

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 224, 224, 3)]	0	
Conv1_pad (ZeroPadding2D)	(None, 225, 225, 3)	0	input_1[0][0]
Conv1 (Conv2D)	(None, 112, 112, 32)	864	Conv1_pad[0][0]
bn_Conv1 (BatchNormalization)	(None, 112, 112, 32)	128	Conv1[0][0]
Conv1_relu (ReLU)	(None, 112, 112, 32)	0	bn_Conv1[0][0]
expanded_conv_depthwise (DepthwiseConv2D)	(None, 112, 112, 32)	288	Conv1_relu[0][0]
expanded_conv_depthwise_BN (BatchNormalization)	(None, 112, 112, 32)	128	expanded_conv_depthwise[0][0]
expanded_conv_depthwise_relu (ReLU)	(None, 112, 112, 32)	0	expanded_conv_depthwise_BN[0][0]
expanded_conv_project (Conv2D)	(None, 112, 112, 16)	512	expanded_conv_depthwise_relu[0][0]
expanded_conv_project_BN (BatchNormalization)	(None, 112, 112, 16)	64	expanded_conv_project[0][0]
block_1_expand (Conv2D)	(None, 112, 112, 96)	1536	expanded_conv_project_BN[0][0]
block_1_expand_BN (BatchNormalization)	(None, 112, 112, 96)	384	block_1_expand[0][0]
block_1_expand_relu (ReLU)	(None, 112, 112, 96)	0	block_1_expand_BN[0][0]
block_1_pad (ZeroPadding2D)	(None, 113, 113, 96)	0	block_1_expand_relu[0][0]
block_1_depthwise (DepthwiseConv2D)	(None, 56, 56, 96)	864	block_1_pad[0][0]
block_1_depthwise_BN (BatchNormalization)	(None, 56, 56, 96)	384	block_1_depthwise[0][0]
block_1_depthwise_relu (ReLU)	(None, 56, 56, 96)	0	block_1_depthwise_BN[0][0]
block_1_project (Conv2D)	(None, 56, 56, 24)	2304	block_1_depthwise_relu[0][0]
block_1_project_BN (BatchNormalization)	(None, 56, 56, 24)	96	block_1_project[0][0]
block_2_expand (Conv2D)	(None, 56, 56, 144)	3456	block_1_project_BN[0][0]
block_2_expand_BN (BatchNormalization)	(None, 56, 56, 144)	576	block_2_expand[0][0]
block_2_expand_relu (ReLU)	(None, 56, 56, 144)	0	block_2_expand_BN[0][0]
block_2_depthwise (DepthwiseConv2D)	(None, 56, 56, 144)	1296	block_2_expand_relu[0][0]
block_2_depthwise_BN (BatchNormalization)	(None, 56, 56, 144)	576	block_2_depthwise[0][0]
block_2_depthwise_relu (ReLU)	(None, 56, 56, 144)	0	block_2_depthwise_BN[0][0]
block_2_project (Conv2D)	(None, 56, 56, 24)	3456	block_2_depthwise_relu[0][0]
block_2_project_BN (BatchNormalization)	(None, 56, 56, 24)	96	block_2_project[0][0]
block_2_add (Add)	(None, 56, 56, 24)	0	block_1_project_BN[0][0] block_2_project_BN[0][0]
block_3_expand (Conv2D)	(None, 56, 56, 144)	3456	block_2_add[0][0]
block_3_expand_BN (BatchNormalization)	(None, 56, 56, 144)	576	block_3_expand[0][0]
block_3_expand_relu (ReLU)	(None, 56, 56, 144)	0	block_3_expand_BN[0][0]
block_3_pad (ZeroPadding2D)	(None, 57, 57, 144)	0	block_3_expand_relu[0][0]
block_3_depthwise (DepthwiseConv2D)	(None, 28, 28, 144)	1296	block_3_pad[0][0]
block_3_depthwise_BN (BatchNormalization)	(None, 28, 28, 144)	576	block_3_depthwise[0][0]
block_3_depthwise_relu (ReLU)	(None, 28, 28, 144)	0	block_3_depthwise_BN[0][0]
block_3_project (Conv2D)	(None, 28, 28, 32)	4608	block_3_depthwise_relu[0][0]
block_3_project_BN (BatchNormalization)	(None, 28, 28, 32)	128	block_3_project[0][0]
block_4_expand (Conv2D)	(None, 28, 28, 192)	6144	block_3_project_BN[0][0]
block_4_expand_BN (BatchNormalization)	(None, 28, 28, 192)	768	block_4_expand[0][0]
block_4_expand_relu (ReLU)	(None, 28, 28, 192)	0	block_4_expand_BN[0][0]
block_4_depthwise (DepthwiseConv2D)	(None, 28, 28, 192)	1728	block_4_expand_relu[0][0]
block_4_depthwise_BN (BatchNormalization)	(None, 28, 28, 192)	768	block_4_depthwise[0][0]

block_4_depthwise_relu (ReLU)	(None, 28, 28, 192)	0	block_4_depthwise_BN[0][0]
block_4_project (Conv2D)	(None, 28, 28, 32)	6144	block_4_depthwise_relu[0][0]
block_4_project_BN (BatchNormal	(None, 28, 28, 32)	128	block_4_project[0][0]
block_4_add (Add)	(None, 28, 28, 32)	0	block_3_project_BN[0][0] block_4_project_BN[0][0]
block_5_expand (Conv2D)	(None, 28, 28, 192)	6144	block_4_add[0][0]
block_5_expand_BN (BatchNormali	(None, 28, 28, 192)	768	block_5_expand[0][0]
block_5_expand_relu (ReLU)	(None, 28, 28, 192)	0	block_5_expand_BN[0][0]
block_5_depthwise (DepthwiseCon	(None, 28, 28, 192)	1728	block_5_expand_relu[0][0]
block_5_depthwise_BN (BatchNorm	(None, 28, 28, 192)	768	block_5_depthwise[0][0]
block_5_depthwise_relu (ReLU)	(None, 28, 28, 192)	0	block_5_depthwise_BN[0][0]
block_5_project (Conv2D)	(None, 28, 28, 32)	6144	block_5_depthwise_relu[0][0]
block_5_project_BN (BatchNormal	(None, 28, 28, 32)	128	block_5_project[0][0]
block_5_add (Add)	(None, 28, 28, 32)	0	block_4_add[0][0] block_5_project_BN[0][0]
block_6_expand (Conv2D)	(None, 28, 28, 192)	6144	block_5_add[0][0]
block_6_expand_BN (BatchNormali	(None, 28, 28, 192)	768	block_6_expand[0][0]
block_6_expand_relu (ReLU)	(None, 28, 28, 192)	0	block_6_expand_BN[0][0]
block_6_pad (ZeroPadding2D)	(None, 29, 29, 192)	0	block_6_expand_relu[0][0]
block_6_depthwise (DepthwiseCon	(None, 14, 14, 192)	1728	block_6_pad[0][0]
block_6_depthwise_BN (BatchNorm	(None, 14, 14, 192)	768	block_6_depthwise[0][0]
block_6_depthwise_relu (ReLU)	(None, 14, 14, 192)	0	block_6_depthwise_BN[0][0]
block_6_project (Conv2D)	(None, 14, 14, 64)	12288	block_6_depthwise_relu[0][0]
block_6_project_BN (BatchNormal	(None, 14, 14, 64)	256	block_6_project[0][0]
block_7_expand (Conv2D)	(None, 14, 14, 384)	24576	block_6_project_BN[0][0]
block_7_expand_BN (BatchNormali	(None, 14, 14, 384)	1536	block_7_expand[0][0]
block_7_expand_relu (ReLU)	(None, 14, 14, 384)	0	block_7_expand_BN[0][0]
block_7_depthwise (DepthwiseCon	(None, 14, 14, 384)	3456	block_7_expand_relu[0][0]
block_7_depthwise_BN (BatchNorm	(None, 14, 14, 384)	1536	block_7_depthwise[0][0]
block_7_depthwise_relu (ReLU)	(None, 14, 14, 384)	0	block_7_depthwise_BN[0][0]
block_7_project (Conv2D)	(None, 14, 14, 64)	24576	block_7_depthwise_relu[0][0]
block_7_project_BN (BatchNormal	(None, 14, 14, 64)	256	block_7_project[0][0]
block_7_add (Add)	(None, 14, 14, 64)	0	block_6_project_BN[0][0] block_7_project_BN[0][0]
block_8_expand (Conv2D)	(None, 14, 14, 384)	24576	block_7_add[0][0]
block_8_expand_BN (BatchNormali	(None, 14, 14, 384)	1536	block_8_expand[0][0]
block_8_expand_relu (ReLU)	(None, 14, 14, 384)	0	block_8_expand_BN[0][0]
block_8_depthwise (DepthwiseCon	(None, 14, 14, 384)	3456	block_8_expand_relu[0][0]
block_8_depthwise_BN (BatchNorm	(None, 14, 14, 384)	1536	block_8_depthwise[0][0]
block_8_depthwise_relu (ReLU)	(None, 14, 14, 384)	0	block_8_depthwise_BN[0][0]
block_8_project (Conv2D)	(None, 14, 14, 64)	24576	block_8_depthwise_relu[0][0]
block_8_project_BN (BatchNormal	(None, 14, 14, 64)	256	block_8_project[0][0]
block_8_add (Add)	(None, 14, 14, 64)	0	block_7_add[0][0] block_8_project_BN[0][0]
block_9_expand (Conv2D)	(None, 14, 14, 384)	24576	block_8_add[0][0]
block_9_expand_BN (BatchNormali	(None, 14, 14, 384)	1536	block_9_expand[0][0]
block_9_expand_relu (ReLU)	(None, 14, 14, 384)	0	block_9_expand_BN[0][0]

block_9_depthwise (DepthwiseCon	(None, 14, 14, 384)	3456	block_9_expand_relu[0][0]
block_9_depthwise_BN (BatchNorm	(None, 14, 14, 384)	1536	block_9_depthwise[0][0]
block_9_depthwise_relu (ReLU)	(None, 14, 14, 384)	0	block_9_depthwise_BN[0][0]
block_9_project (Conv2D)	(None, 14, 14, 64)	24576	block_9_depthwise_relu[0][0]
block_9_project_BN (BatchNormal	(None, 14, 14, 64)	256	block_9_project[0][0]
block_9_add (Add)	(None, 14, 14, 64)	0	block_8_add[0][0] block_9_project_BN[0][0]
block_10_expand (Conv2D)	(None, 14, 14, 384)	24576	block_9_add[0][0]
block_10_expand_BN (BatchNormal	(None, 14, 14, 384)	1536	block_10_expand[0][0]
block_10_expand_relu (ReLU)	(None, 14, 14, 384)	0	block_10_expand_BN[0][0]
block_10_depthwise (DepthwiseCo	(None, 14, 14, 384)	3456	block_10_expand_relu[0][0]
block_10_depthwise_BN (BatchNor	(None, 14, 14, 384)	1536	block_10_depthwise[0][0]
block_10_depthwise_relu (ReLU)	(None, 14, 14, 384)	0	block_10_depthwise_BN[0][0]
block_10_project (Conv2D)	(None, 14, 14, 96)	36864	block_10_depthwise_relu[0][0]
block_10_project_BN (BatchNorma	(None, 14, 14, 96)	384	block_10_project[0][0]
block_11_expand (Conv2D)	(None, 14, 14, 576)	55296	block_10_project_BN[0][0]
block_11_expand_BN (BatchNormal	(None, 14, 14, 576)	2304	block_11_expand[0][0]
block_11_expand_relu (ReLU)	(None, 14, 14, 576)	0	block_11_expand_BN[0][0]
block_11_depthwise (DepthwiseCo	(None, 14, 14, 576)	5184	block_11_expand_relu[0][0]
block_11_depthwise_BN (BatchNor	(None, 14, 14, 576)	2304	block_11_depthwise[0][0]
block_11_depthwise_relu (ReLU)	(None, 14, 14, 576)	0	block_11_depthwise_BN[0][0]
block_11_project (Conv2D)	(None, 14, 14, 96)	55296	block_11_depthwise_relu[0][0]
block_11_project_BN (BatchNorma	(None, 14, 14, 96)	384	block_11_project[0][0]
block_11_add (Add)	(None, 14, 14, 96)	0	block_10_project_BN[0][0] block_11_project_BN[0][0]
block_12_expand (Conv2D)	(None, 14, 14, 576)	55296	block_11_add[0][0]
block_12_expand_BN (BatchNormal	(None, 14, 14, 576)	2304	block_12_expand[0][0]
block_12_expand_relu (ReLU)	(None, 14, 14, 576)	0	block_12_expand_BN[0][0]
block_12_depthwise (DepthwiseCo	(None, 14, 14, 576)	5184	block_12_expand_relu[0][0]
block_12_depthwise_BN (BatchNor	(None, 14, 14, 576)	2304	block_12_depthwise[0][0]
block_12_depthwise_relu (ReLU)	(None, 14, 14, 576)	0	block_12_depthwise_BN[0][0]
block_12_project (Conv2D)	(None, 14, 14, 96)	55296	block_12_depthwise_relu[0][0]
block_12_project_BN (BatchNorma	(None, 14, 14, 96)	384	block_12_project[0][0]
block_12_add (Add)	(None, 14, 14, 96)	0	block_11_add[0][0] block_12_project_BN[0][0]
block_13_expand (Conv2D)	(None, 14, 14, 576)	55296	block_12_add[0][0]
block_13_expand_BN (BatchNormal	(None, 14, 14, 576)	2304	block_13_expand[0][0]
block_13_expand_relu (ReLU)	(None, 14, 14, 576)	0	block_13_expand_BN[0][0]
block_13_pad (ZeroPadding2D)	(None, 15, 15, 576)	0	block_13_expand_relu[0][0]
block_13_depthwise (DepthwiseCo	(None, 7, 7, 576)	5184	block_13_pad[0][0]
block_13_depthwise_BN (BatchNor	(None, 7, 7, 576)	2304	block_13_depthwise[0][0]
block_13_depthwise_relu (ReLU)	(None, 7, 7, 576)	0	block_13_depthwise_BN[0][0]
block_13_project (Conv2D)	(None, 7, 7, 160)	92160	block_13_depthwise_relu[0][0]
block_13_project_BN (BatchNorma	(None, 7, 7, 160)	640	block_13_project[0][0]
block_14_expand (Conv2D)	(None, 7, 7, 960)	153600	block_13_project_BN[0][0]
block_14_expand_BN (BatchNormal	(None, 7, 7, 960)	3840	block_14_expand[0][0]
block_14_expand_relu (ReLU)	(None, 7, 7, 960)	0	block_14_expand_BN[0][0]

block_14_depthwise (DepthwiseCo (None, 7, 7, 960)	8640	block_14_expand_relu[0][0]
block_14_depthwise_BN (BatchNor (None, 7, 7, 960)	3840	block_14_depthwise[0][0]
block_14_depthwise_relu (ReLU) (None, 7, 7, 960)	0	block_14_depthwise_BN[0][0]
block_14_project (Conv2D) (None, 7, 7, 160)	153600	block_14_depthwise_relu[0][0]
block_14_project_BN (BatchNorma (None, 7, 7, 160)	640	block_14_project[0][0]
block_14_add (Add) (None, 7, 7, 160)	0	block_13_project_BN[0][0] block_14_project_BN[0][0]
block_15_expand (Conv2D) (None, 7, 7, 960)	153600	block_14_add[0][0]
block_15_expand_BN (BatchNormal (None, 7, 7, 960)	3840	block_15_expand[0][0]
block_15_expand_relu (ReLU) (None, 7, 7, 960)	0	block_15_expand_BN[0][0]
block_15_depthwise (DepthwiseCo (None, 7, 7, 960)	8640	block_15_expand_relu[0][0]
block_15_depthwise_BN (BatchNor (None, 7, 7, 960)	3840	block_15_depthwise[0][0]
block_15_depthwise_relu (ReLU) (None, 7, 7, 960)	0	block_15_depthwise_BN[0][0]
block_15_project (Conv2D) (None, 7, 7, 160)	153600	block_15_depthwise_relu[0][0]
block_15_project_BN (BatchNorma (None, 7, 7, 160)	640	block_15_project[0][0]
block_15_add (Add) (None, 7, 7, 160)	0	block_14_add[0][0] block_15_project_BN[0][0]
block_16_expand (Conv2D) (None, 7, 7, 960)	153600	block_15_add[0][0]
block_16_expand_BN (BatchNormal (None, 7, 7, 960)	3840	block_16_expand[0][0]
block_16_expand_relu (ReLU) (None, 7, 7, 960)	0	block_16_expand_BN[0][0]
block_16_depthwise (DepthwiseCo (None, 7, 7, 960)	8640	block_16_expand_relu[0][0]
block_16_depthwise_BN (BatchNor (None, 7, 7, 960)	3840	block_16_depthwise[0][0]
block_16_depthwise_relu (ReLU) (None, 7, 7, 960)	0	block_16_depthwise_BN[0][0]
block_16_project (Conv2D) (None, 7, 7, 320)	307200	block_16_depthwise_relu[0][0]
block_16_project_BN (BatchNorma (None, 7, 7, 320)	1280	block_16_project[0][0]
Conv_1 (Conv2D) (None, 7, 7, 1280)	409600	block_16_project_BN[0][0]
Conv_1_bn (BatchNormalization) (None, 7, 7, 1280)	5120	Conv_1[0][0]
out_relu (ReLU) (None, 7, 7, 1280)	0	Conv_1_bn[0][0]
global_average_pooling2d (Globa (None, 1280)	0	out_relu[0][0]
predictions (Dense) (None, 1000)	1281000	global_average_pooling2d[0][0]

=====

Total params: 3,538,984
Trainable params: 3,504,872
Non-trainable params: 34,112

Model: "functional_1"

Layer (type)	Output Shape	Param #
=====		
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
=====		
mobilenetv2_1.00_224 (Functi	(None, 1000)	3538984
=====		
flatten (Flatten)	(None, 1000)	0
=====		
output_layer (Dense)	(None, 2)	2002
=====		

Total params: 3,540,986
Trainable params: 3,506,874
Non-trainable params: 34,112

```
In [12]: for layer in custom_resnet_model.layers[:-1]:
        layer.trainable = False

        custom_resnet_model.layers[-1].trainable
```

Out[12]: True

```
In [13]: custom_resnet_model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

```
In [14]: t = time.time()
hist = custom_resnet_model.fit(X_train, y_train, batch_size=32, epochs=num_epoch, verbose=1, validation_data=(X_test, y_test))
print('Training time: %s' % (t - time.time()))
(loss, accuracy) = custom_resnet_model.evaluate(X_test, y_test, batch_size=10, verbose=1)
print("[INFO] loss={:.4f}, accuracy: {:.4f}%".format(loss, accuracy * 100))
```

Epoch 1/100
147/147 [=====] - 7s 48ms/step - loss: 0.6437 - accuracy: 0.7252 - val_loss: 0.6028 - val_accuracy: 0.7338
Epoch 2/100
147/147 [=====] - 6s 41ms/step - loss: 0.5841 - accuracy: 0.7289 - val_loss: 0.5623 - val_accuracy: 0.7338
Epoch 3/100
147/147 [=====] - 6s 41ms/step - loss: 0.5552 - accuracy: 0.7289 - val_loss: 0.5391 - val_accuracy: 0.7338
Epoch 4/100
147/147 [=====] - 6s 41ms/step - loss: 0.5367 - accuracy: 0.7293 - val_loss: 0.5222 - val_accuracy: 0.7338
Epoch 5/100
147/147 [=====] - 6s 41ms/step - loss: 0.5217 - accuracy: 0.7312 - val_loss: 0.5073 - val_accuracy: 0.7398
Epoch 6/100
147/147 [=====] - 6s 41ms/step - loss: 0.5084 - accuracy: 0.7340 - val_loss: 0.4940 - val_accuracy: 0.7406
Epoch 7/100
147/147 [=====] - 6s 41ms/step - loss: 0.4961 - accuracy: 0.7376 - val_loss: 0.4815 - val_accuracy: 0.7449
Epoch 8/100
147/147 [=====] - 6s 41ms/step - loss: 0.4846 - accuracy: 0.7427 - val_loss: 0.4699 - val_accuracy: 0.7517
Epoch 9/100
147/147 [=====] - 6s 41ms/step - loss: 0.4740 - accuracy: 0.7487 - val_loss: 0.4591 - val_accuracy: 0.7568
Epoch 10/100
147/147 [=====] - 6s 41ms/step - loss: 0.4642 - accuracy: 0.7556 - val_loss: 0.4488 - val_accuracy: 0.7568
Epoch 11/100
147/147 [=====] - 6s 41ms/step - loss: 0.4549 - accuracy: 0.7626 - val_loss: 0.4394 - val_accuracy: 0.7619
Epoch 12/100
147/147 [=====] - 6s 41ms/step - loss: 0.4463 - accuracy: 0.7690 - val_loss: 0.4305 - val_accuracy: 0.7628
Epoch 13/100
147/147 [=====] - 6s 41ms/step - loss: 0.4381 - accuracy: 0.7763 - val_loss: 0.4221 - val_accuracy: 0.7713
Epoch 14/100
147/147 [=====] - 6s 41ms/step - loss: 0.4306 - accuracy: 0.7835 - val_loss: 0.4143 - val_accuracy: 0.7807
Epoch 15/100
147/147 [=====] - 6s 41ms/step - loss: 0.4235 - accuracy: 0.7912 - val_loss: 0.4069 - val_accuracy: 0.7850
Epoch 16/100
147/147 [=====] - 6s 41ms/step - loss: 0.4169 - accuracy: 0.7965 - val_loss: 0.4000 - val_accuracy: 0.7884
Epoch 17/100
147/147 [=====] - 6s 41ms/step - loss: 0.4106 - accuracy: 0.8019 - val_loss: 0.3934 - val_accuracy: 0.7986
Epoch 18/100
147/147 [=====] - 6s 41ms/step - loss: 0.4046 - accuracy: 0.8104 - val_loss: 0.3872 - val_accuracy: 0.8063
Epoch 19/100
147/147 [=====] - 6s 41ms/step - loss: 0.3991 - accuracy: 0.8164 - val_loss: 0.3813 - val_accuracy: 0.8217
Epoch 20/100
147/147 [=====] - 6s 41ms/step - loss: 0.3939 - accuracy: 0.8228 - val_loss: 0.3758 - val_accuracy: 0.8294
Epoch 21/100
147/147 [=====] - 6s 41ms/step - loss: 0.3888 - accuracy: 0.8256 - val_loss: 0.3705 - val_accuracy: 0.8345
Epoch 22/100
147/147 [=====] - 6s 41ms/step - loss: 0.3842 - accuracy: 0.8309 - val_loss: 0.3656 - val_accuracy: 0.8370
Epoch 23/100
147/147 [=====] - 6s 41ms/step - loss: 0.3797 - accuracy: 0.8373 - val_loss: 0.3608 - val_accuracy: 0.8413
Epoch 24/100
147/147 [=====] - 6s 41ms/step - loss: 0.3755 - accuracy: 0.8407 - val_loss: 0.3563 - val_accuracy: 0.8456
Epoch 25/100
147/147 [=====] - 6s 41ms/step - loss: 0.3715 - accuracy: 0.8448 - val_loss: 0.3520 - val_accuracy: 0.8473
Epoch 26/100
147/147 [=====] - 6s 41ms/step - loss: 0.3676 - accuracy: 0.8486 - val_loss: 0.3480 - val_accuracy: 0.8549
Epoch 27/100
147/147 [=====] - 6s 41ms/step - loss: 0.3639 - accuracy: 0.8499 - val_loss: 0.3441 - val_accuracy: 0.8592
Epoch 28/100
147/147 [=====] - 6s 41ms/step - loss: 0.3605 - accuracy: 0.8540 - val_loss: 0.3403 - val_accuracy: 0.8601
Epoch 29/100
147/147 [=====] - 6s 41ms/step - loss: 0.3572 - accuracy: 0.8559 - val_loss: 0.3368 - val_accuracy: 0.8626
Epoch 30/100
147/147 [=====] - 6s 41ms/step - loss: 0.3540 - accuracy: 0.8589 - val_loss: 0.3334 - val_accuracy: 0.8686
Epoch 31/100
147/147 [=====] - 6s 41ms/step - loss: 0.3509 - accuracy: 0.8610 - val_loss: 0.3301 - val_accuracy: 0.8712
Epoch 32/100
147/147 [=====] - 6s 41ms/step - loss: 0.3480 - accuracy: 0.8649 - val_loss: 0.3270 - val_accuracy: 0.8754
Epoch 33/100
147/147 [=====] - 6s 41ms/step - loss: 0.3452 - accuracy: 0.8678 - val_loss: 0.3240 - val_accuracy: 0.8763
Epoch 34/100
147/147 [=====] - 6s 41ms/step - loss: 0.3425 - accuracy: 0.8706 - val_loss: 0.3211 - val_accuracy: 0.8788
Epoch 35/100
147/147 [=====] - 6s 41ms/step - loss: 0.3399 - accuracy: 0.8711 - val_loss: 0.3183 - val_accuracy: 0.8823
Epoch 36/100
147/147 [=====] - 6s 41ms/step - loss: 0.3375 - accuracy: 0.8723 - val_loss: 0.3157 - val_accuracy: 0.8848
Epoch 37/100
147/147 [=====] - 6s 41ms/step - loss: 0.3351 - accuracy: 0.8762 - val_loss: 0.3131 - val_accuracy: 0.8874
Epoch 38/100
147/147 [=====] - 6s 41ms/step - loss: 0.3328 - accuracy: 0.8772 - val_loss: 0.3106 - val_accuracy: 0.8865
Epoch 39/100
147/147 [=====] - 6s 41ms/step - loss: 0.3306 - accuracy: 0.8787 - val_loss: 0.3083 - val_accuracy: 0.8891
Epoch 40/100
147/147 [=====] - 6s 41ms/step - loss: 0.3284 - accuracy: 0.8796 - val_loss: 0.3060 - val_accuracy: 0.8916
Epoch 41/100
147/147 [=====] - 6s 41ms/step - loss: 0.3264 - accuracy: 0.8798 - val_loss: 0.3038 - val_accuracy: 0.8916
Epoch 42/100
147/147 [=====] - 6s 41ms/step - loss: 0.3244 - accuracy: 0.8813 - val_loss: 0.3017 - val_accuracy: 0.8925
Epoch 43/100
147/147 [=====] - 6s 41ms/step - loss: 0.3225 - accuracy: 0.8817 - val_loss: 0.2996 - val_accuracy: 0.8933
Epoch 44/100
147/147 [=====] - 6s 41ms/step - loss: 0.3206 - accuracy: 0.8826 - val_loss: 0.2977 - val_accuracy: 0.8933
Epoch 45/100
147/147 [=====] - 6s 41ms/step - loss: 0.3188 - accuracy: 0.8830 - val_loss: 0.2958 - val_accuracy: 0.8951
Epoch 46/100

147/147 [=====]	- 6s	41ms/step	- loss: 0.3170	- accuracy: 0.8847	- val_loss: 0.2939	- val_accuracy: 0.8951
Epoch 47/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.3154	- accuracy: 0.8854	- val_loss: 0.2921	- val_accuracy: 0.8942
Epoch 48/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.3137	- accuracy: 0.8875	- val_loss: 0.2903	- val_accuracy: 0.8951
Epoch 49/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.3121	- accuracy: 0.8871	- val_loss: 0.2886	- val_accuracy: 0.8959
Epoch 50/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.3106	- accuracy: 0.8881	- val_loss: 0.2870	- val_accuracy: 0.8968
Epoch 51/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.3090	- accuracy: 0.8890	- val_loss: 0.2854	- val_accuracy: 0.8976
Epoch 52/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.3076	- accuracy: 0.8894	- val_loss: 0.2838	- val_accuracy: 0.8976
Epoch 53/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.3062	- accuracy: 0.8898	- val_loss: 0.2823	- val_accuracy: 0.8976
Epoch 54/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.3048	- accuracy: 0.8911	- val_loss: 0.2809	- val_accuracy: 0.8985
Epoch 55/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.3034	- accuracy: 0.8907	- val_loss: 0.2795	- val_accuracy: 0.8985
Epoch 56/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.3021	- accuracy: 0.8915	- val_loss: 0.2781	- val_accuracy: 0.9019
Epoch 57/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.3008	- accuracy: 0.8920	- val_loss: 0.2768	- val_accuracy: 0.9019
Epoch 58/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2995	- accuracy: 0.8922	- val_loss: 0.2755	- val_accuracy: 0.9019
Epoch 59/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2984	- accuracy: 0.8924	- val_loss: 0.2742	- val_accuracy: 0.9027
Epoch 60/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2971	- accuracy: 0.8930	- val_loss: 0.2730	- val_accuracy: 0.9036
Epoch 61/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2960	- accuracy: 0.8928	- val_loss: 0.2718	- val_accuracy: 0.9053
Epoch 62/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2949	- accuracy: 0.8922	- val_loss: 0.2706	- val_accuracy: 0.9061
Epoch 63/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2937	- accuracy: 0.8943	- val_loss: 0.2695	- val_accuracy: 0.9070
Epoch 64/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2927	- accuracy: 0.8950	- val_loss: 0.2684	- val_accuracy: 0.9070
Epoch 65/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2916	- accuracy: 0.8950	- val_loss: 0.2673	- val_accuracy: 0.9078
Epoch 66/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2906	- accuracy: 0.8952	- val_loss: 0.2663	- val_accuracy: 0.9078
Epoch 67/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2897	- accuracy: 0.8960	- val_loss: 0.2652	- val_accuracy: 0.9087
Epoch 68/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2887	- accuracy: 0.8954	- val_loss: 0.2642	- val_accuracy: 0.9096
Epoch 69/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2877	- accuracy: 0.8975	- val_loss: 0.2633	- val_accuracy: 0.9096
Epoch 70/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2867	- accuracy: 0.8975	- val_loss: 0.2623	- val_accuracy: 0.9096
Epoch 71/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2858	- accuracy: 0.8986	- val_loss: 0.2614	- val_accuracy: 0.9113
Epoch 72/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2849	- accuracy: 0.8986	- val_loss: 0.2605	- val_accuracy: 0.9104
Epoch 73/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2841	- accuracy: 0.8984	- val_loss: 0.2596	- val_accuracy: 0.9113
Epoch 74/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2832	- accuracy: 0.8984	- val_loss: 0.2588	- val_accuracy: 0.9113
Epoch 75/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.2823	- accuracy: 0.8982	- val_loss: 0.2579	- val_accuracy: 0.9113
Epoch 76/100						
147/147 [=====]	- 6s	41ms/step	- loss: 0.281			

```

Epoch 92/100
147/147 [=====] - 6s 41ms/step - loss: 0.2702 - accuracy: 0.9029 - val_loss: 0.2460 - val_accuracy: 0.9121
Epoch 93/100
147/147 [=====] - 6s 41ms/step - loss: 0.2696 - accuracy: 0.9026 - val_loss: 0.2454 - val_accuracy: 0.9121
Epoch 94/100
147/147 [=====] - 6s 41ms/step - loss: 0.2690 - accuracy: 0.9026 - val_loss: 0.2449 - val_accuracy: 0.9121
Epoch 95/100
147/147 [=====] - 6s 41ms/step - loss: 0.2684 - accuracy: 0.9033 - val_loss: 0.2443 - val_accuracy: 0.9121
Epoch 96/100
147/147 [=====] - 6s 41ms/step - loss: 0.2678 - accuracy: 0.9035 - val_loss: 0.2437 - val_accuracy: 0.9130
Epoch 97/100
147/147 [=====] - 6s 41ms/step - loss: 0.2672 - accuracy: 0.9033 - val_loss: 0.2432 - val_accuracy: 0.9130
Epoch 98/100
147/147 [=====] - 6s 41ms/step - loss: 0.2667 - accuracy: 0.9037 - val_loss: 0.2426 - val_accuracy: 0.9138
Epoch 99/100
147/147 [=====] - 6s 41ms/step - loss: 0.2661 - accuracy: 0.9037 - val_loss: 0.2421 - val_accuracy: 0.9138
Epoch 100/100
147/147 [=====] - 6s 41ms/step - loss: 0.2655 - accuracy: 0.9039 - val_loss: 0.2416 - val_accuracy: 0.9147
Training time: -609.8109776973724
118/118 [=====] - 2s 17ms/step - loss: 0.2416 - accuracy: 0.9147
[INFO] loss=0.2416, accuracy: 91.4676%

```

```

In [15]: (loss, accuracy) = custom_resnet_model.evaluate(X_test, y_test, batch_size=10, verbose=1)

print("[INFO] loss={:.4f}, accuracy: {:.4f}%".format(loss, accuracy * 100))

118/118 [=====] - 2s 15ms/step - loss: 0.2416 - accuracy: 0.9147
[INFO] loss=0.2416, accuracy: 91.4676%

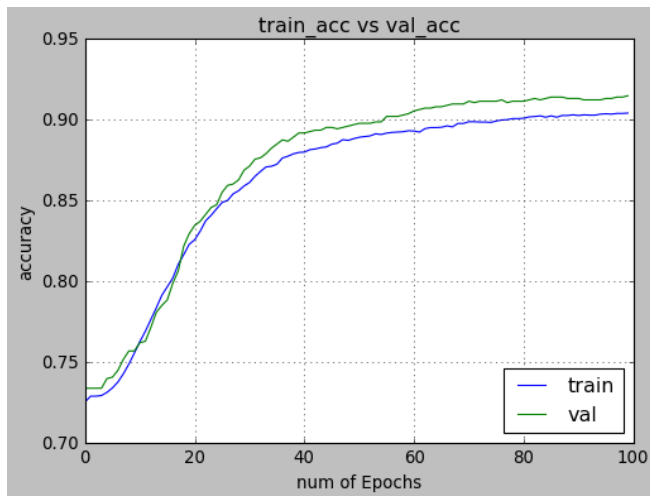
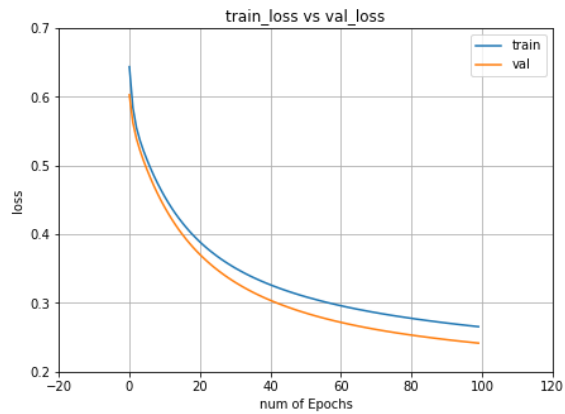
```

visualizing losses and accuracy

```

In [16]: display_loss_accuracy(hist)

```



Resnet50


```
In [11]: image_input = Input(shape=(224, 224, 3))
model = ResNet50(input_tensor=image_input, include_top=True, weights='imagenet')
model.summary()
last_layer = model.get_layer('avg_pool').output
x= Flatten(name='flatten')(last_layer)
out = Dense(num_classes, activation='softmax', name='output_layer')(x)
custom_resnet_model = Model(inputs=image_input, outputs= out)
custom_resnet_model.summary()
```

Model: "resnet50"

Layer (type)	Output Shape	Param #	Connected to
=====			
input_1 (InputLayer)	[(None, 224, 224, 3)]	0	
conv1_pad (ZeroPadding2D)	(None, 230, 230, 3)	0	input_1[0][0]
conv1_conv (Conv2D)	(None, 112, 112, 64)	9472	conv1_pad[0][0]
conv1_bn (BatchNormalization)	(None, 112, 112, 64)	256	conv1_conv[0][0]
conv1_relu (Activation)	(None, 112, 112, 64)	0	conv1_bn[0][0]
pool1_pad (ZeroPadding2D)	(None, 114, 114, 64)	0	conv1_relu[0][0]
pool1_pool (MaxPooling2D)	(None, 56, 56, 64)	0	pool1_pad[0][0]
conv2_block1_1_conv (Conv2D)	(None, 56, 56, 64)	4160	pool1_pool[0][0]
conv2_block1_1_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block1_1_conv[0][0]
conv2_block1_1_relu (Activation	(None, 56, 56, 64)	0	conv2_block1_1_bn[0][0]
conv2_block1_2_conv (Conv2D)	(None, 56, 56, 64)	36928	conv2_block1_1_relu[0][0]
conv2_block1_2_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block1_2_conv[0][0]
conv2_block1_2_relu (Activation	(None, 56, 56, 64)	0	conv2_block1_2_bn[0][0]
conv2_block1_0_conv (Conv2D)	(None, 56, 56, 256)	16640	pool1_pool[0][0]
conv2_block1_3_conv (Conv2D)	(None, 56, 56, 256)	16640	conv2_block1_2_relu[0][0]
conv2_block1_0_bn (BatchNormali	(None, 56, 56, 256)	1024	conv2_block1_0_conv[0][0]
conv2_block1_3_bn (BatchNormali	(None, 56, 56, 256)	1024	conv2_block1_3_conv[0][0]
conv2_block1_add (Add)	(None, 56, 56, 256)	0	conv2_block1_0_bn[0][0] conv2_block1_3_bn[0][0]
conv2_block1_out (Activation)	(None, 56, 56, 256)	0	conv2_block1_add[0][0]
conv2_block2_1_conv (Conv2D)	(None, 56, 56, 64)	16448	conv2_block1_out[0][0]
conv2_block2_1_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block2_1_conv[0][0]
conv2_block2_1_relu (Activation	(None, 56, 56, 64)	0	conv2_block2_1_bn[0][0]
conv2_block2_2_conv (Conv2D)	(None, 56, 56, 64)	36928	conv2_block2_1_relu[0][0]
conv2_block2_2_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block2_2_conv[0][0]
conv2_block2_2_relu (Activation	(None, 56, 56, 64)	0	conv2_block2_2_bn[0][0]
conv2_block2_3_conv (Conv2D)	(None, 56, 56, 256)	16640	conv2_block2_2_relu[0][0]
conv2_block2_3_bn (BatchNormali	(None, 56, 56, 256)	1024	conv2_block2_3_conv[0][0]
conv2_block2_add (Add)	(None, 56, 56, 256)	0	conv2_block1_out[0][0] conv2_block2_3_bn[0][0]
conv2_block2_out (Activation)	(None, 56, 56, 256)	0	conv2_block2_add[0][0]
conv2_block3_1_conv (Conv2D)	(None, 56, 56, 64)	16448	conv2_block2_out[0][0]
conv2_block3_1_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block3_1_conv[0][0]
conv2_block3_1_relu (Activation	(None, 56, 56, 64)	0	conv2_block3_1_bn[0][0]
conv2_block3_2_conv (Conv2D)	(None, 56, 56, 64)	36928	conv2_block3_1_relu[0][0]
conv2_block3_2_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block3_2_conv[0][0]
conv2_block3_2_relu (Activation	(None, 56, 56, 64)	0	conv2_block3_2_bn[0][0]
conv2_block3_3_conv (Conv2D)	(None, 56, 56, 256)	16640	conv2_block3_2_relu[0][0]
conv2_block3_3_bn (BatchNormali	(None, 56, 56, 256)	1024	conv2_block3_3_conv[0][0]
conv2_block3_add (Add)	(None, 56, 56, 256)	0	conv2_block2_out[0][0] conv2_block3_3_bn[0][0]
conv2_block3_out (Activation)	(None, 56, 56, 256)	0	conv2_block3_add[0][0]
conv3_block1_1_conv (Conv2D)	(None, 28, 28, 128)	32896	conv2_block3_out[0][0]
conv3_block1_1_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block1_1_conv[0][0]
conv3_block1_1_relu (Activation	(None, 28, 28, 128)	0	conv3_block1_1_bn[0][0]

conv3_block1_2_conv (Conv2D)	(None, 28, 28, 128)	147584	conv3_block1_1_relu[0][0]
conv3_block1_2_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block1_2_conv[0][0]
conv3_block1_2_relu (Activation	(None, 28, 28, 128)	0	conv3_block1_2_bn[0][0]
conv3_block1_0_conv (Conv2D)	(None, 28, 28, 512)	131584	conv2_block3_out[0][0]
conv3_block1_3_conv (Conv2D)	(None, 28, 28, 512)	66048	conv3_block1_2_relu[0][0]
conv3_block1_0_bn (BatchNormali	(None, 28, 28, 512)	2048	conv3_block1_0_conv[0][0]
conv3_block1_3_bn (BatchNormali	(None, 28, 28, 512)	2048	conv3_block1_3_conv[0][0]
conv3_block1_add (Add)	(None, 28, 28, 512)	0	conv3_block1_0_bn[0][0] conv3_block1_3_bn[0][0]
conv3_block1_out (Activation)	(None, 28, 28, 512)	0	conv3_block1_add[0][0]
conv3_block2_1_conv (Conv2D)	(None, 28, 28, 128)	65664	conv3_block1_out[0][0]
conv3_block2_1_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block2_1_conv[0][0]
conv3_block2_1_relu (Activation	(None, 28, 28, 128)	0	conv3_block2_1_bn[0][0]
conv3_block2_2_conv (Conv2D)	(None, 28, 28, 128)	147584	conv3_block2_1_relu[0][0]
conv3_block2_2_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block2_2_conv[0][0]
conv3_block2_2_relu (Activation	(None, 28, 28, 128)	0	conv3_block2_2_bn[0][0]
conv3_block2_3_conv (Conv2D)	(None, 28, 28, 512)	66048	conv3_block2_2_relu[0][0]
conv3_block2_3_bn (BatchNormali	(None, 28, 28, 512)	2048	conv3_block2_3_conv[0][0]
conv3_block2_add (Add)	(None, 28, 28, 512)	0	conv3_block1_out[0][0] conv3_block2_3_bn[0][0]
conv3_block2_out (Activation)	(None, 28, 28, 512)	0	conv3_block2_add[0][0]
conv3_block3_1_conv (Conv2D)	(None, 28, 28, 128)	65664	conv3_block2_out[0][0]
conv3_block3_1_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block3_1_conv[0][0]
conv3_block3_1_relu (Activation	(None, 28, 28, 128)	0	conv3_block3_1_bn[0][0]
conv3_block3_2_conv (Conv2D)	(None, 28, 28, 128)	147584	conv3_block3_1_relu[0][0]
conv3_block3_2_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block3_2_conv[0][0]
conv3_block3_2_relu (Activation	(None, 28, 28, 128)	0	conv3_block3_2_bn[0][0]
conv3_block3_3_conv (Conv2D)	(None, 28, 28, 512)	66048	conv3_block3_2_relu[0][0]
conv3_block3_3_bn (BatchNormali	(None, 28, 28, 512)	2048	conv3_block3_3_conv[0][0]
conv3_block3_add (Add)	(None, 28, 28, 512)	0	conv3_block2_out[0][0] conv3_block3_3_bn[0][0]
conv3_block3_out (Activation)	(None, 28, 28, 512)	0	conv3_block3_add[0][0]
conv3_block4_1_conv (Conv2D)	(None, 28, 28, 128)	65664	conv3_block3_out[0][0]
conv3_block4_1_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block4_1_conv[0][0]
conv3_block4_1_relu (Activation	(None, 28, 28, 128)	0	conv3_block4_1_bn[0][0]
conv3_block4_2_conv (Conv2D)	(None, 28, 28, 128)	147584	conv3_block4_1_relu[0][0]
conv3_block4_2_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block4_2_conv[0][0]
conv3_block4_2_relu (Activation	(None, 28, 28, 128)	0	conv3_block4_2_bn[0][0]
conv3_block4_3_conv (Conv2D)	(None, 28, 28, 512)	66048	conv3_block4_2_relu[0][0]
conv3_block4_3_bn (BatchNormali	(None, 28, 28, 512)	2048	conv3_block4_3_conv[0][0]
conv3_block4_add (Add)	(None, 28, 28, 512)	0	conv3_block3_out[0][0] conv3_block4_3_bn[0][0]
conv3_block4_out (Activation)	(None, 28, 28, 512)	0	conv3_block4_add[0][0]
conv4_block1_1_conv (Conv2D)	(None, 14, 14, 256)	131328	conv3_block4_out[0][0]
conv4_block1_1_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block1_1_conv[0][0]
conv4_block1_1_relu (Activation	(None, 14, 14, 256)	0	conv4_block1_1_bn[0][0]
conv4_block1_2_conv (Conv2D)	(None, 14, 14, 256)	590080	conv4_block1_1_relu[0][0]
conv4_block1_2_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block1_2_conv[0][0]

conv4_block1_2_relu (Activation (None, 14, 14, 256) 0	conv4_block1_2_bn[0][0]
conv4_block1_0_conv (Conv2D) (None, 14, 14, 1024) 525312	conv3_block4_out[0][0]
conv4_block1_3_conv (Conv2D) (None, 14, 14, 1024) 263168	conv4_block1_2_relu[0][0]
conv4_block1_0_bn (BatchNormali (None, 14, 14, 1024) 4096	conv4_block1_0_conv[0][0]
conv4_block1_3_bn (BatchNormali (None, 14, 14, 1024) 4096	conv4_block1_3_conv[0][0]
conv4_block1_add (Add) (None, 14, 14, 1024) 0	conv4_block1_0_bn[0][0] conv4_block1_3_bn[0][0]
conv4_block1_out (Activation) (None, 14, 14, 1024) 0	conv4_block1_add[0][0]
conv4_block2_1_conv (Conv2D) (None, 14, 14, 256) 262400	conv4_block1_out[0][0]
conv4_block2_1_bn (BatchNormali (None, 14, 14, 256) 1024	conv4_block2_1_conv[0][0]
conv4_block2_1_relu (Activation (None, 14, 14, 256) 0	conv4_block2_1_bn[0][0]
conv4_block2_2_conv (Conv2D) (None, 14, 14, 256) 590080	conv4_block2_1_relu[0][0]
conv4_block2_2_bn (BatchNormali (None, 14, 14, 256) 1024	conv4_block2_2_conv[0][0]
conv4_block2_2_relu (Activation (None, 14, 14, 256) 0	conv4_block2_2_bn[0][0]
conv4_block2_3_conv (Conv2D) (None, 14, 14, 1024) 263168	conv4_block2_2_relu[0][0]
conv4_block2_3_bn (BatchNormali (None, 14, 14, 1024) 4096	conv4_block2_3_conv[0][0]
conv4_block2_add (Add) (None, 14, 14, 1024) 0	conv4_block1_out[0][0] conv4_block2_3_bn[0][0]
conv4_block2_out (Activation) (None, 14, 14, 1024) 0	conv4_block2_add[0][0]
conv4_block3_1_conv (Conv2D) (None, 14, 14, 256) 262400	conv4_block2_out[0][0]
conv4_block3_1_bn (BatchNormali (None, 14, 14, 256) 1024	conv4_block3_1_conv[0][0]
conv4_block3_1_relu (Activation (None, 14, 14, 256) 0	conv4_block3_1_bn[0][0]
conv4_block3_2_conv (Conv2D) (None, 14, 14, 256) 590080	conv4_block3_1_relu[0][0]
conv4_block3_2_bn (BatchNormali (None, 14, 14, 256) 1024	conv4_block3_2_conv[0][0]
conv4_block3_2_relu (Activation (None, 14, 14, 256) 0	conv4_block3_2_bn[0][0]
conv4_block3_3_conv (Conv2D) (None, 14, 14, 1024) 263168	conv4_block3_2_relu[0][0]
conv4_block3_3_bn (BatchNormali (None, 14, 14, 1024) 4096	conv4_block3_3_conv[0][0]
conv4_block3_add (Add) (None, 14, 14, 1024) 0	conv4_block2_out[0][0] conv4_block3_3_bn[0][0]
conv4_block3_out (Activation) (None, 14, 14, 1024) 0	conv4_block3_add[0][0]
conv4_block4_1_conv (Conv2D) (None, 14, 14, 256) 262400	conv4_block3_out[0][0]
conv4_block4_1_bn (BatchNormali (None, 14, 14, 256) 1024	conv4_block4_1_conv[0][0]
conv4_block4_1_relu (Activation (None, 14, 14, 256) 0	conv4_block4_1_bn[0][0]
conv4_block4_2_conv (Conv2D) (None, 14, 14, 256) 590080	conv4_block4_1_relu[0][0]
conv4_block4_2_bn (BatchNormali (None, 14, 14, 256) 1024	conv4_block4_2_conv[0][0]
conv4_block4_2_relu (Activation (None, 14, 14, 256) 0	conv4_block4_2_bn[0][0]
conv4_block4_3_conv (Conv2D) (None, 14, 14, 1024) 263168	conv4_block4_2_relu[0][0]
conv4_block4_3_bn (BatchNormali (None, 14, 14, 1024) 4096	conv4_block4_3_conv[0][0]
conv4_block4_add (Add) (None, 14, 14, 1024) 0	conv4_block3_out[0][0] conv4_block4_3_bn[0][0]
conv4_block4_out (Activation) (None, 14, 14, 1024) 0	conv4_block4_add[0][0]
conv4_block5_1_conv (Conv2D) (None, 14, 14, 256) 262400	conv4_block4_out[0][0]
conv4_block5_1_bn (BatchNormali (None, 14, 14, 256) 1024	conv4_block5_1_conv[0][0]
conv4_block5_1_relu (Activation (None, 14, 14, 256) 0	conv4_block5_1_bn[0][0]
conv4_block5_2_conv (Conv2D) (None, 14, 14, 256) 590080	conv4_block5_1_relu[0][0]
conv4_block5_2_bn (BatchNormali (None, 14, 14, 256) 1024	conv4_block5_2_conv[0][0]
conv4_block5_2_relu (Activation (None, 14, 14, 256) 0	conv4_block5_2_bn[0][0]

conv4_block5_3_conv (Conv2D)	(None, 14, 14, 1024)	263168	conv4_block5_2_relu[0][0]
conv4_block5_3_bn (BatchNormali	(None, 14, 14, 1024)	4096	conv4_block5_3_conv[0][0]
conv4_block5_add (Add)	(None, 14, 14, 1024)	0	conv4_block4_out[0][0] conv4_block5_3_bn[0][0]
conv4_block5_out (Activation)	(None, 14, 14, 1024)	0	conv4_block5_add[0][0]
conv4_block6_1_conv (Conv2D)	(None, 14, 14, 256)	262400	conv4_block5_out[0][0]
conv4_block6_1_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block6_1_conv[0][0]
conv4_block6_1_relu (Activation	(None, 14, 14, 256)	0	conv4_block6_1_bn[0][0]
conv4_block6_2_conv (Conv2D)	(None, 14, 14, 256)	590080	conv4_block6_1_relu[0][0]
conv4_block6_2_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block6_2_conv[0][0]
conv4_block6_2_relu (Activation	(None, 14, 14, 256)	0	conv4_block6_2_bn[0][0]
conv4_block6_3_conv (Conv2D)	(None, 14, 14, 1024)	263168	conv4_block6_2_relu[0][0]
conv4_block6_3_bn (BatchNormali	(None, 14, 14, 1024)	4096	conv4_block6_3_conv[0][0]
conv4_block6_add (Add)	(None, 14, 14, 1024)	0	conv4_block5_out[0][0] conv4_block6_3_bn[0][0]
conv4_block6_out (Activation)	(None, 14, 14, 1024)	0	conv4_block6_add[0][0]
conv5_block1_1_conv (Conv2D)	(None, 7, 7, 512)	524800	conv4_block6_out[0][0]
conv5_block1_1_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block1_1_conv[0][0]
conv5_block1_1_relu (Activation	(None, 7, 7, 512)	0	conv5_block1_1_bn[0][0]
conv5_block1_2_conv (Conv2D)	(None, 7, 7, 512)	2359808	conv5_block1_1_relu[0][0]
conv5_block1_2_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block1_2_conv[0][0]
conv5_block1_2_relu (Activation	(None, 7, 7, 512)	0	conv5_block1_2_bn[0][0]
conv5_block1_0_conv (Conv2D)	(None, 7, 7, 2048)	2099200	conv4_block6_out[0][0]
conv5_block1_3_conv (Conv2D)	(None, 7, 7, 2048)	1050624	conv5_block1_2_relu[0][0]
conv5_block1_0_bn (BatchNormali	(None, 7, 7, 2048)	8192	conv5_block1_0_conv[0][0]
conv5_block1_3_bn (BatchNormali	(None, 7, 7, 2048)	8192	conv5_block1_3_conv[0][0]
conv5_block1_add (Add)	(None, 7, 7, 2048)	0	conv5_block1_0_bn[0][0] conv5_block1_3_bn[0][0]
conv5_block1_out (Activation)	(None, 7, 7, 2048)	0	conv5_block1_add[0][0]
conv5_block2_1_conv (Conv2D)	(None, 7, 7, 512)	1049088	conv5_block1_out[0][0]
conv5_block2_1_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block2_1_conv[0][0]
conv5_block2_1_relu (Activation	(None, 7, 7, 512)	0	conv5_block2_1_bn[0][0]
conv5_block2_2_conv (Conv2D)	(None, 7, 7, 512)	2359808	conv5_block2_1_relu[0][0]
conv5_block2_2_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block2_2_conv[0][0]
conv5_block2_2_relu (Activation	(None, 7, 7, 512)	0	conv5_block2_2_bn[0][0]
conv5_block2_3_conv (Conv2D)	(None, 7, 7, 2048)	1050624	conv5_block2_2_relu[0][0]
conv5_block2_3_bn (BatchNormali	(None, 7, 7, 2048)	8192	conv5_block2_3_conv[0][0]
conv5_block2_add (Add)	(None, 7, 7, 2048)	0	conv5_block1_out[0][0] conv5_block2_3_bn[0][0]
conv5_block2_out (Activation)	(None, 7, 7, 2048)	0	conv5_block2_add[0][0]
conv5_block3_1_conv (Conv2D)	(None, 7, 7, 512)	1049088	conv5_block2_out[0][0]
conv5_block3_1_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block3_1_conv[0][0]
conv5_block3_1_relu (Activation	(None, 7, 7, 512)	0	conv5_block3_1_bn[0][0]
conv5_block3_2_conv (Conv2D)	(None, 7, 7, 512)	2359808	conv5_block3_1_relu[0][0]
conv5_block3_2_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block3_2_conv[0][0]
conv5_block3_2_relu (Activation	(None, 7, 7, 512)	0	conv5_block3_2_bn[0][0]
conv5_block3_3_conv (Conv2D)	(None, 7, 7, 2048)	1050624	conv5_block3_2_relu[0][0]
conv5_block3_3_bn (BatchNormali	(None, 7, 7, 2048)	8192	conv5_block3_3_conv[0][0]

conv5_block3_add (Add)	(None, 7, 7, 2048)	0	conv5_block2_out[0][0] conv5_block3_3_bn[0][0]
conv5_block3_out (Activation)	(None, 7, 7, 2048)	0	conv5_block3_add[0][0]
avg_pool1 (GlobalAveragePooling2)	(None, 2048)	0	conv5_block3_out[0][0]
predictions (Dense)	(None, 1000)	2049000	avg_pool[0][0]
=====			
Total params: 25,636,712			
Trainable params: 25,583,592			
Non-trainable params: 53,120			
Model: "functional_1"			
Layer (type)	Output Shape	Param #	Connected to
=====			
input_1 (InputLayer)	[(None, 224, 224, 3)]	0	
conv1_pad (ZeroPadding2D)	(None, 230, 230, 3)	0	input_1[0][0]
conv1_conv (Conv2D)	(None, 112, 112, 64)	9472	conv1_pad[0][0]
conv1_bn (BatchNormalization)	(None, 112, 112, 64)	256	conv1_conv[0][0]
conv1_relu (Activation)	(None, 112, 112, 64)	0	conv1_bn[0][0]
pool1_pad (ZeroPadding2D)	(None, 114, 114, 64)	0	conv1_relu[0][0]
pool1_pool (MaxPooling2D)	(None, 56, 56, 64)	0	pool1_pad[0][0]
conv2_block1_1_conv (Conv2D)	(None, 56, 56, 64)	4160	pool1_pool[0][0]
conv2_block1_1_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block1_1_conv[0][0]
conv2_block1_1_relu (Activation	(None, 56, 56, 64)	0	conv2_block1_1_bn[0][0]
conv2_block1_2_conv (Conv2D)	(None, 56, 56, 64)	36928	conv2_block1_1_relu[0][0]
conv2_block1_2_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block1_2_conv[0][0]
conv2_block1_2_relu (Activation	(None, 56, 56, 64)	0	conv2_block1_2_bn[0][0]
conv2_block1_0_conv (Conv2D)	(None, 56, 56, 256)	16640	pool1_pool[0][0]
conv2_block1_3_conv (Conv2D)	(None, 56, 56, 256)	16640	conv2_block1_2_relu[0][0]
conv2_block1_0_bn (BatchNormali	(None, 56, 56, 256)	1024	conv2_block1_0_conv[0][0]
conv2_block1_3_bn (BatchNormali	(None, 56, 56, 256)	1024	conv2_block1_3_conv[0][0]
conv2_block1_add (Add)	(None, 56, 56, 256)	0	conv2_block1_0_bn[0][0] conv2_block1_3_bn[0][0]
conv2_block1_out (Activation)	(None, 56, 56, 256)	0	conv2_block1_add[0][0]
conv2_block2_1_conv (Conv2D)	(None, 56, 56, 64)	16448	conv2_block1_out[0][0]
conv2_block2_1_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block2_1_conv[0][0]
conv2_block2_1_relu (Activation	(None, 56, 56, 64)	0	conv2_block2_1_bn[0][0]
conv2_block2_2_conv (Conv2D)	(None, 56, 56, 64)	36928	conv2_block2_1_relu[0][0]
conv2_block2_2_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block2_2_conv[0][0]
conv2_block2_2_relu (Activation	(None, 56, 56, 64)	0	conv2_block2_2_bn[0][0]
conv2_block2_3_conv (Conv2D)	(None, 56, 56, 256)	16640	conv2_block2_2_relu[0][0]
conv2_block2_3_bn (BatchNormali	(None, 56, 56, 256)	1024	conv2_block2_3_conv[0][0]
conv2_block2_add (Add)	(None, 56, 56, 256)	0	conv2_block1_out[0][0] conv2_block2_3_bn[0][0]
conv2_block2_out (Activation)	(None, 56, 56, 256)	0	conv2_block2_add[0][0]
conv2_block3_1_conv (Conv2D)	(None, 56, 56, 64)	16448	conv2_block2_out[0][0]
conv2_block3_1_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block3_1_conv[0][0]
conv2_block3_1_relu (Activation	(None, 56, 56, 64)	0	conv2_block3_1_bn[0][0]
conv2_block3_2_conv (Conv2D)	(None, 56, 56, 64)	36928	conv2_block3_1_relu[0][0]
conv2_block3_2_bn (BatchNormali	(None, 56, 56, 64)	256	conv2_block3_2_conv[0][0]
conv2_block3_2_relu (Activation	(None, 56, 56, 64)	0	conv2_block3_2_bn[0][0]
conv2_block3_3_conv (Conv2D)	(None, 56, 56, 256)	16640	conv2_block3_2_relu[0][0]

conv2_block3_3_bn (BatchNormali	(None, 56, 56, 256)	1024	conv2_block3_3_conv[0][0]
conv2_block3_add (Add)	(None, 56, 56, 256)	0	conv2_block2_out[0][0] conv2_block3_3_bn[0][0]
conv2_block3_out (Activation)	(None, 56, 56, 256)	0	conv2_block3_add[0][0]
conv3_block1_1_conv (Conv2D)	(None, 28, 28, 128)	32896	conv2_block3_out[0][0]
conv3_block1_1_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block1_1_conv[0][0]
conv3_block1_1_relu (Activation	(None, 28, 28, 128)	0	conv3_block1_1_bn[0][0]
conv3_block1_2_conv (Conv2D)	(None, 28, 28, 128)	147584	conv3_block1_1_relu[0][0]
conv3_block1_2_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block1_2_conv[0][0]
conv3_block1_2_relu (Activation	(None, 28, 28, 128)	0	conv3_block1_2_bn[0][0]
conv3_block1_0_conv (Conv2D)	(None, 28, 28, 512)	131584	conv2_block3_out[0][0]
conv3_block1_3_conv (Conv2D)	(None, 28, 28, 512)	66048	conv3_block1_2_relu[0][0]
conv3_block1_0_bn (BatchNormali	(None, 28, 28, 512)	2048	conv3_block1_0_conv[0][0]
conv3_block1_3_bn (BatchNormali	(None, 28, 28, 512)	2048	conv3_block1_3_conv[0][0]
conv3_block1_add (Add)	(None, 28, 28, 512)	0	conv3_block1_0_bn[0][0] conv3_block1_3_bn[0][0]
conv3_block1_out (Activation)	(None, 28, 28, 512)	0	conv3_block1_add[0][0]
conv3_block2_1_conv (Conv2D)	(None, 28, 28, 128)	65664	conv3_block1_out[0][0]
conv3_block2_1_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block2_1_conv[0][0]
conv3_block2_1_relu (Activation	(None, 28, 28, 128)	0	conv3_block2_1_bn[0][0]
conv3_block2_2_conv (Conv2D)	(None, 28, 28, 128)	147584	conv3_block2_1_relu[0][0]
conv3_block2_2_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block2_2_conv[0][0]
conv3_block2_2_relu (Activation	(None, 28, 28, 128)	0	conv3_block2_2_bn[0][0]
conv3_block2_3_conv (Conv2D)	(None, 28, 28, 512)	66048	conv3_block2_2_relu[0][0]
conv3_block2_3_bn (BatchNormali	(None, 28, 28, 512)	2048	conv3_block2_3_conv[0][0]
conv3_block2_add (Add)	(None, 28, 28, 512)	0	conv3_block1_out[0][0] conv3_block2_3_bn[0][0]
conv3_block2_out (Activation)	(None, 28, 28, 512)	0	conv3_block2_add[0][0]
conv3_block3_1_conv (Conv2D)	(None, 28, 28, 128)	65664	conv3_block2_out[0][0]
conv3_block3_1_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block3_1_conv[0][0]
conv3_block3_1_relu (Activation	(None, 28, 28, 128)	0	conv3_block3_1_bn[0][0]
conv3_block3_2_conv (Conv2D)	(None, 28, 28, 128)	147584	conv3_block3_1_relu[0][0]
conv3_block3_2_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block3_2_conv[0][0]
conv3_block3_2_relu (Activation	(None, 28, 28, 128)	0	conv3_block3_2_bn[0][0]
conv3_block3_3_conv (Conv2D)	(None, 28, 28, 512)	66048	conv3_block3_2_relu[0][0]
conv3_block3_3_bn (BatchNormali	(None, 28, 28, 512)	2048	conv3_block3_3_conv[0][0]
conv3_block3_add (Add)	(None, 28, 28, 512)	0	conv3_block2_out[0][0] conv3_block3_3_bn[0][0]
conv3_block3_out (Activation)	(None, 28, 28, 512)	0	conv3_block3_add[0][0]
conv3_block4_1_conv (Conv2D)	(None, 28, 28, 128)	65664	conv3_block3_out[0][0]
conv3_block4_1_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block4_1_conv[0][0]
conv3_block4_1_relu (Activation	(None, 28, 28, 128)	0	conv3_block4_1_bn[0][0]
conv3_block4_2_conv (Conv2D)	(None, 28, 28, 128)	147584	conv3_block4_1_relu[0][0]
conv3_block4_2_bn (BatchNormali	(None, 28, 28, 128)	512	conv3_block4_2_conv[0][0]
conv3_block4_2_relu (Activation	(None, 28, 28, 128)	0	conv3_block4_2_bn[0][0]
conv3_block4_3_conv (Conv2D)	(None, 28, 28, 512)	66048	conv3_block4_2_relu[0][0]
conv3_block4_3_bn (BatchNormali	(None, 28, 28, 512)	2048	conv3_block4_3_conv[0][0]

conv3_block4_add (Add)	(None, 28, 28, 512)	0	conv3_block3_out[0][0] conv3_block4_3_bn[0][0]
conv3_block4_out (Activation)	(None, 28, 28, 512)	0	conv3_block4_add[0][0]
conv4_block1_1_conv (Conv2D)	(None, 14, 14, 256)	131328	conv3_block4_out[0][0]
conv4_block1_1_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block1_1_conv[0][0]
conv4_block1_1_relu (Activation	(None, 14, 14, 256)	0	conv4_block1_1_bn[0][0]
conv4_block1_2_conv (Conv2D)	(None, 14, 14, 256)	590080	conv4_block1_1_relu[0][0]
conv4_block1_2_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block1_2_conv[0][0]
conv4_block1_2_relu (Activation	(None, 14, 14, 256)	0	conv4_block1_2_bn[0][0]
conv4_block1_0_conv (Conv2D)	(None, 14, 14, 1024)	525312	conv3_block4_out[0][0]
conv4_block1_3_conv (Conv2D)	(None, 14, 14, 1024)	263168	conv4_block1_2_relu[0][0]
conv4_block1_0_bn (BatchNormali	(None, 14, 14, 1024)	4096	conv4_block1_0_conv[0][0]
conv4_block1_3_bn (BatchNormali	(None, 14, 14, 1024)	4096	conv4_block1_3_conv[0][0]
conv4_block1_add (Add)	(None, 14, 14, 1024)	0	conv4_block1_0_bn[0][0] conv4_block1_3_bn[0][0]
conv4_block1_out (Activation)	(None, 14, 14, 1024)	0	conv4_block1_add[0][0]
conv4_block2_1_conv (Conv2D)	(None, 14, 14, 256)	262400	conv4_block1_out[0][0]
conv4_block2_1_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block2_1_conv[0][0]
conv4_block2_1_relu (Activation	(None, 14, 14, 256)	0	conv4_block2_1_bn[0][0]
conv4_block2_2_conv (Conv2D)	(None, 14, 14, 256)	590080	conv4_block2_1_relu[0][0]
conv4_block2_2_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block2_2_conv[0][0]
conv4_block2_2_relu (Activation	(None, 14, 14, 256)	0	conv4_block2_2_bn[0][0]
conv4_block2_3_conv (Conv2D)	(None, 14, 14, 1024)	263168	conv4_block2_2_relu[0][0]
conv4_block2_3_bn (BatchNormali	(None, 14, 14, 1024)	4096	conv4_block2_3_conv[0][0]
conv4_block2_add (Add)	(None, 14, 14, 1024)	0	conv4_block1_out[0][0] conv4_block2_3_bn[0][0]
conv4_block2_out (Activation)	(None, 14, 14, 1024)	0	conv4_block2_add[0][0]
conv4_block3_1_conv (Conv2D)	(None, 14, 14, 256)	262400	conv4_block2_out[0][0]
conv4_block3_1_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block3_1_conv[0][0]
conv4_block3_1_relu (Activation	(None, 14, 14, 256)	0	conv4_block3_1_bn[0][0]
conv4_block3_2_conv (Conv2D)	(None, 14, 14, 256)	590080	conv4_block3_1_relu[0][0]
conv4_block3_2_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block3_2_conv[0][0]
conv4_block3_2_relu (Activation	(None, 14, 14, 256)	0	conv4_block3_2_bn[0][0]
conv4_block3_3_conv (Conv2D)	(None, 14, 14, 1024)	263168	conv4_block3_2_relu[0][0]
conv4_block3_3_bn (BatchNormali	(None, 14, 14, 1024)	4096	conv4_block3_3_conv[0][0]
conv4_block3_add (Add)	(None, 14, 14, 1024)	0	conv4_block2_out[0][0] conv4_block3_3_bn[0][0]
conv4_block3_out (Activation)	(None, 14, 14, 1024)	0	conv4_block3_add[0][0]
conv4_block4_1_conv (Conv2D)	(None, 14, 14, 256)	262400	conv4_block3_out[0][0]
conv4_block4_1_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block4_1_conv[0][0]
conv4_block4_1_relu (Activation	(None, 14, 14, 256)	0	conv4_block4_1_bn[0][0]
conv4_block4_2_conv (Conv2D)	(None, 14, 14, 256)	590080	conv4_block4_1_relu[0][0]
conv4_block4_2_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block4_2_conv[0][0]
conv4_block4_2_relu (Activation	(None, 14, 14, 256)	0	conv4_block4_2_bn[0][0]
conv4_block4_3_conv (Conv2D)	(None, 14, 14, 1024)	263168	conv4_block4_2_relu[0][0]
conv4_block4_3_bn (BatchNormali	(None, 14, 14, 1024)	4096	conv4_block4_3_conv[0][0]
conv4_block4_add (Add)	(None, 14, 14, 1024)	0	conv4_block3_out[0][0] conv4_block4_3_bn[0][0]

conv4_block4_out (Activation)	(None, 14, 14, 1024)	0	conv4_block4_add[0][0]
conv4_block5_1_conv (Conv2D)	(None, 14, 14, 256)	262400	conv4_block4_out[0][0]
conv4_block5_1_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block5_1_conv[0][0]
conv4_block5_1_relu (Activation	(None, 14, 14, 256)	0	conv4_block5_1_bn[0][0]
conv4_block5_2_conv (Conv2D)	(None, 14, 14, 256)	590080	conv4_block5_1_relu[0][0]
conv4_block5_2_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block5_2_conv[0][0]
conv4_block5_2_relu (Activation	(None, 14, 14, 256)	0	conv4_block5_2_bn[0][0]
conv4_block5_3_conv (Conv2D)	(None, 14, 14, 1024)	263168	conv4_block5_2_relu[0][0]
conv4_block5_3_bn (BatchNormali	(None, 14, 14, 1024)	4096	conv4_block5_3_conv[0][0]
conv4_block5_add (Add)	(None, 14, 14, 1024)	0	conv4_block4_out[0][0] conv4_block5_3_bn[0][0]
conv4_block5_out (Activation)	(None, 14, 14, 1024)	0	conv4_block5_add[0][0]
conv4_block6_1_conv (Conv2D)	(None, 14, 14, 256)	262400	conv4_block5_out[0][0]
conv4_block6_1_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block6_1_conv[0][0]
conv4_block6_1_relu (Activation	(None, 14, 14, 256)	0	conv4_block6_1_bn[0][0]
conv4_block6_2_conv (Conv2D)	(None, 14, 14, 256)	590080	conv4_block6_1_relu[0][0]
conv4_block6_2_bn (BatchNormali	(None, 14, 14, 256)	1024	conv4_block6_2_conv[0][0]
conv4_block6_2_relu (Activation	(None, 14, 14, 256)	0	conv4_block6_2_bn[0][0]
conv4_block6_3_conv (Conv2D)	(None, 14, 14, 1024)	263168	conv4_block6_2_relu[0][0]
conv4_block6_3_bn (BatchNormali	(None, 14, 14, 1024)	4096	conv4_block6_3_conv[0][0]
conv4_block6_add (Add)	(None, 14, 14, 1024)	0	conv4_block5_out[0][0] conv4_block6_3_bn[0][0]
conv4_block6_out (Activation)	(None, 14, 14, 1024)	0	conv4_block6_add[0][0]
conv5_block1_1_conv (Conv2D)	(None, 7, 7, 512)	524800	conv4_block6_out[0][0]
conv5_block1_1_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block1_1_conv[0][0]
conv5_block1_1_relu (Activation	(None, 7, 7, 512)	0	conv5_block1_1_bn[0][0]
conv5_block1_2_conv (Conv2D)	(None, 7, 7, 512)	2359808	conv5_block1_1_relu[0][0]
conv5_block1_2_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block1_2_conv[0][0]
conv5_block1_2_relu (Activation	(None, 7, 7, 512)	0	conv5_block1_2_bn[0][0]
conv5_block1_0_conv (Conv2D)	(None, 7, 7, 2048)	2099200	conv4_block6_out[0][0]
conv5_block1_3_conv (Conv2D)	(None, 7, 7, 2048)	1050624	conv5_block1_2_relu[0][0]
conv5_block1_0_bn (BatchNormali	(None, 7, 7, 2048)	8192	conv5_block1_0_conv[0][0]
conv5_block1_3_bn (BatchNormali	(None, 7, 7, 2048)	8192	conv5_block1_3_conv[0][0]
conv5_block1_add (Add)	(None, 7, 7, 2048)	0	conv5_block1_0_bn[0][0] conv5_block1_3_bn[0][0]
conv5_block1_out (Activation)	(None, 7, 7, 2048)	0	conv5_block1_add[0][0]
conv5_block2_1_conv (Conv2D)	(None, 7, 7, 512)	1049088	conv5_block1_out[0][0]
conv5_block2_1_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block2_1_conv[0][0]
conv5_block2_1_relu (Activation	(None, 7, 7, 512)	0	conv5_block2_1_bn[0][0]
conv5_block2_2_conv (Conv2D)	(None, 7, 7, 512)	2359808	conv5_block2_1_relu[0][0]
conv5_block2_2_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block2_2_conv[0][0]
conv5_block2_2_relu (Activation	(None, 7, 7, 512)	0	conv5_block2_2_bn[0][0]
conv5_block2_3_conv (Conv2D)	(None, 7, 7, 2048)	1050624	conv5_block2_2_relu[0][0]
conv5_block2_3_bn (BatchNormali	(None, 7, 7, 2048)	8192	conv5_block2_3_conv[0][0]
conv5_block2_add (Add)	(None, 7, 7, 2048)	0	conv5_block1_out[0][0] conv5_block2_3_bn[0][0]
conv5_block2_out (Activation)	(None, 7, 7, 2048)	0	conv5_block2_add[0][0]
conv5_block3_1_conv (Conv2D)	(None, 7, 7, 512)	1049088	conv5_block2_out[0][0]

conv5_block3_1_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block3_1_conv[0][0]
conv5_block3_1_relu (Activation	(None, 7, 7, 512)	0	conv5_block3_1_bn[0][0]
conv5_block3_2_conv (Conv2D)	(None, 7, 7, 512)	2359808	conv5_block3_1_relu[0][0]
conv5_block3_2_bn (BatchNormali	(None, 7, 7, 512)	2048	conv5_block3_2_conv[0][0]
conv5_block3_2_relu (Activation	(None, 7, 7, 512)	0	conv5_block3_2_bn[0][0]
conv5_block3_3_conv (Conv2D)	(None, 7, 7, 2048)	1050624	conv5_block3_2_relu[0][0]
conv5_block3_3_bn (BatchNormali	(None, 7, 7, 2048)	8192	conv5_block3_3_conv[0][0]
conv5_block3_add (Add)	(None, 7, 7, 2048)	0	conv5_block2_out[0][0] conv5_block3_3_bn[0][0]
conv5_block3_out (Activation)	(None, 7, 7, 2048)	0	conv5_block3_add[0][0]
avg_pool (GlobalAveragePooling2	(None, 2048)	0	conv5_block3_out[0][0]
flatten (Flatten)	(None, 2048)	0	avg_pool[0][0]
output_layer (Dense)	(None, 2)	4098	flatten[0][0]
=====			
Total params: 23,591,810			
Trainable params: 23,538,690			
Non-trainable params: 53,120			

```
In [12]: for layer in custom_resnet_model.layers[:-1]:
        layer.trainable = False

        custom_resnet_model.layers[-1].trainable
```

Out[12]: True

```
In [13]: custom_resnet_model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

```
In [14]: t=time.time()
hist = custom_resnet_model.fit(X_train, y_train, batch_size=32, epochs=num_epoch, verbose=1, validation_data=(X_test, y_test))
print('Training time: %s' % (t - time.time()))
(loss, accuracy) = custom_resnet_model.evaluate(X_test, y_test, batch_size=10, verbose=1)
print("[INFO] loss={:.4f}, accuracy: {:.4f}%".format(loss, accuracy * 100))
```

Epoch 1/100
147/147 [=====] - 11s 73ms/step - loss: 0.2067 - accuracy: 0.9142 - val_loss: 0.1091 - val_accuracy: 0.9616
Epoch 2/100
147/147 [=====] - 10s 65ms/step - loss: 0.1197 - accuracy: 0.9554 - val_loss: 0.0955 - val_accuracy: 0.9676
Epoch 3/100
147/147 [=====] - 10s 65ms/step - loss: 0.0949 - accuracy: 0.9661 - val_loss: 0.1064 - val_accuracy: 0.9582
Epoch 4/100
147/147 [=====] - 10s 65ms/step - loss: 0.0924 - accuracy: 0.9654 - val_loss: 0.0828 - val_accuracy: 0.9718
Epoch 5/100
147/147 [=====] - 10s 65ms/step - loss: 0.0836 - accuracy: 0.9686 - val_loss: 0.0833 - val_accuracy: 0.9753
Epoch 6/100
147/147 [=====] - 10s 65ms/step - loss: 0.0743 - accuracy: 0.9733 - val_loss: 0.0775 - val_accuracy: 0.9761
Epoch 7/100
147/147 [=====] - 10s 65ms/step - loss: 0.0727 - accuracy: 0.9757 - val_loss: 0.0867 - val_accuracy: 0.9744
Epoch 8/100
147/147 [=====] - 10s 65ms/step - loss: 0.0664 - accuracy: 0.9765 - val_loss: 0.0931 - val_accuracy: 0.9701
Epoch 9/100
147/147 [=====] - 10s 65ms/step - loss: 0.0622 - accuracy: 0.9782 - val_loss: 0.0788 - val_accuracy: 0.9727
Epoch 10/100
147/147 [=====] - 10s 65ms/step - loss: 0.0598 - accuracy: 0.9793 - val_loss: 0.0762 - val_accuracy: 0.9770
Epoch 11/100
147/147 [=====] - 10s 65ms/step - loss: 0.0532 - accuracy: 0.9827 - val_loss: 0.0738 - val_accuracy: 0.9753
Epoch 12/100
147/147 [=====] - 10s 65ms/step - loss: 0.0503 - accuracy: 0.9821 - val_loss: 0.0948 - val_accuracy: 0.9710
Epoch 13/100
147/147 [=====] - 10s 65ms/step - loss: 0.0492 - accuracy: 0.9819 - val_loss: 0.0755 - val_accuracy: 0.9753
Epoch 14/100
147/147 [=====] - 10s 65ms/step - loss: 0.0486 - accuracy: 0.9836 - val_loss: 0.0798 - val_accuracy: 0.9770
Epoch 15/100
147/147 [=====] - 10s 65ms/step - loss: 0.0464 - accuracy: 0.9842 - val_loss: 0.0955 - val_accuracy: 0.9625
Epoch 16/100
147/147 [=====] - 10s 65ms/step - loss: 0.0418 - accuracy: 0.9861 - val_loss: 0.0794 - val_accuracy: 0.9744
Epoch 17/100
147/147 [=====] - 10s 65ms/step - loss: 0.0412 - accuracy: 0.9878 - val_loss: 0.0813 - val_accuracy: 0.9744
Epoch 18/100
147/147 [=====] - 10s 65ms/step - loss: 0.0438 - accuracy: 0.9855 - val_loss: 0.0734 - val_accuracy: 0.9787
Epoch 19/100
147/147 [=====] - 10s 65ms/step - loss: 0.0393 - accuracy: 0.9872 - val_loss: 0.0721 - val_accuracy: 0.9787
Epoch 20/100
147/147 [=====] - 10s 65ms/step - loss: 0.0358 - accuracy: 0.9883 - val_loss: 0.0855 - val_accuracy: 0.9684
Epoch 21/100
147/147 [=====] - 10s 65ms/step - loss: 0.0341 - accuracy: 0.9908 - val_loss: 0.0864 - val_accuracy: 0.9735
Epoch 22/100
147/147 [=====] - 10s 65ms/step - loss: 0.0340 - accuracy: 0.9898 - val_loss: 0.0735 - val_accuracy: 0.9778
Epoch 23/100
147/147 [=====] - 10s 65ms/step - loss: 0.0353 - accuracy: 0.9887 - val_loss: 0.0731 - val_accuracy: 0.9761
Epoch 24/100
147/147 [=====] - 10s 65ms/step - loss: 0.0286 - accuracy: 0.9921 - val_loss: 0.0750 - val_accuracy: 0.9770
Epoch 25/100
147/147 [=====] - 10s 65ms/step - loss: 0.0283 - accuracy: 0.9912 - val_loss: 0.0749 - val_accuracy: 0.9787
Epoch 26/100
147/147 [=====] - 10s 65ms/step - loss: 0.0277 - accuracy: 0.9923 - val_loss: 0.0863 - val_accuracy: 0.9727
Epoch 27/100
147/147 [=====] - 10s 65ms/step - loss: 0.0267 - accuracy: 0.9927 - val_loss: 0.0818 - val_accuracy: 0.9727
Epoch 28/100
147/147 [=====] - 10s 66ms/step - loss: 0.0250 - accuracy: 0.9940 - val_loss: 0.0814 - val_accuracy: 0.9735
Epoch 29/100
147/147 [=====] - 10s 66ms/step - loss: 0.0243 - accuracy: 0.9936 - val_loss: 0.0795 - val_accuracy: 0.9753
Epoch 30/100
147/147 [=====] - 10s 65ms/step - loss: 0.0240 - accuracy: 0.9936 - val_loss: 0.0763 - val_accuracy: 0.9778
Epoch 31/100
147/147 [=====] - 10s 66ms/step - loss: 0.0265 - accuracy: 0.9925 - val_loss: 0.0915 - val_accuracy: 0.9727
Epoch 32/100
147/147 [=====] - 10s 66ms/step - loss: 0.0206 - accuracy: 0.9951 - val_loss: 0.0808 - val_accuracy: 0.9735
Epoch 33/100
147/147 [=====] - 10s 65ms/step - loss: 0.0198 - accuracy: 0.9951 - val_loss: 0.0782 - val_accuracy: 0.9761
Epoch 34/100
147/147 [=====] - 10s 65ms/step - loss: 0.0194 - accuracy: 0.9964 - val_loss: 0.0955 - val_accuracy: 0.9693
Epoch 35/100
147/147 [=====] - 10s 65ms/step - loss: 0.0182 - accuracy: 0.9959 - val_loss: 0.0821 - val_accuracy: 0.9753
Epoch 36/100
147/147 [=====] - 10s 66ms/step - loss: 0.0179 - accuracy: 0.9966 - val_loss: 0.0887 - val_accuracy: 0.9735
Epoch 37/100
147/147 [=====] - 10s 65ms/step - loss: 0.0186 - accuracy: 0.9964 - val_loss: 0.0857 - val_accuracy: 0.9761
Epoch 38/100
147/147 [=====] - 10s 65ms/step - loss: 0.0165 - accuracy: 0.9968 - val_loss: 0.0872 - val_accuracy: 0.9735
Epoch 39/100
147/147 [=====] - 10s 65ms/step - loss: 0.0150 - accuracy: 0.9981 - val_loss: 0.0834 - val_accuracy: 0.9753
Epoch 40/100
147/147 [=====] - 10s 65ms/step - loss: 0.0180 - accuracy: 0.9944 - val_loss: 0.0843 - val_accuracy: 0.9761
Epoch 41/100
147/147 [=====] - 10s 66ms/step - loss: 0.0170 - accuracy: 0.9962 - val_loss: 0.0986 - val_accuracy: 0.9735
Epoch 42/100
147/147 [=====] - 10s 66ms/step - loss: 0.0151 - accuracy: 0.9972 - val_loss: 0.0936 - val_accuracy: 0.9735
Epoch 43/100
147/147 [=====] - 10s 65ms/step - loss: 0.0168 - accuracy: 0.9962 - val_loss: 0.0853 - val_accuracy: 0.9761
Epoch 44/100
147/147 [=====] - 10s 65ms/step - loss: 0.0127 - accuracy: 0.9983 - val_loss: 0.1224 - val_accuracy: 0.9684
Epoch 45/100
147/147 [=====] - 10s 66ms/step - loss: 0.0167 - accuracy: 0.9957 - val_loss: 0.1033 - val_accuracy: 0.9710
Epoch 46/100

147/147 [=====] - 10s 66ms/step - loss: 0.0117 - accuracy: 0.9987 - val_loss: 0.0952 - val_accuracy: 0.9727
Epoch 47/100
147/147 [=====] - 10s 66ms/step - loss: 0.0111 - accuracy: 0.9983 - val_loss: 0.0926 - val_accuracy: 0.9744
Epoch 48/100
147/147 [=====] - 10s 65ms/step - loss: 0.0141 - accuracy: 0.9981 - val_loss: 0.0909 - val_accuracy: 0.9753
Epoch 49/100
147/147 [=====] - 10s 65ms/step - loss: 0.0113 - accuracy: 0.9983 - val_loss: 0.0916 - val_accuracy: 0.9744
Epoch 50/100
147/147 [=====] - 10s 66ms/step - loss: 0.0101 - accuracy: 0.9991 - val_loss: 0.0862 - val_accuracy: 0.9778
Epoch 51/100
147/147 [=====] - 10s 66ms/step - loss: 0.0101 - accuracy: 0.9989 - val_loss: 0.0902 - val_accuracy: 0.9753
Epoch 52/100
147/147 [=====] - 10s 66ms/step - loss: 0.0094 - accuracy: 0.9994 - val_loss: 0.0990 - val_accuracy: 0.9753
Epoch 53/100
147/147 [=====] - 10s 65ms/step - loss: 0.0090 - accuracy: 0.9996 - val_loss: 0.1023 - val_accuracy: 0.9718
Epoch 54/100
147/147 [=====] - 10s 65ms/step - loss: 0.0096 - accuracy: 0.9991 - val_loss: 0.0929 - val_accuracy: 0.9761
Epoch 55/100
147/147 [=====] - 10s 66ms/step - loss: 0.0090 - accuracy: 0.9991 - val_loss: 0.0948 - val_accuracy: 0.9727
Epoch 56/100
147/147 [=====] - 10s 65ms/step - loss: 0.0100 - accuracy: 0.9991 - val_loss: 0.1009 - val_accuracy: 0.9710
Epoch 57/100
147/147 [=====] - 10s 65ms/step - loss: 0.0083 - accuracy: 0.9996 - val_loss: 0.0982 - val_accuracy: 0.9727
Epoch 58/100
147/147 [=====] - 10s 66ms/step - loss: 0.0081 - accuracy: 0.9996 - val_loss: 0.0913 - val_accuracy: 0.9770
Epoch 59/100
147/147 [=====] - 10s 66ms/step - loss: 0.0100 - accuracy: 0.9985 - val_loss: 0.1036 - val_accuracy: 0.9744
Epoch 60/100
147/147 [=====] - 10s 65ms/step - loss: 0.0077 - accuracy: 0.9998 - val_loss: 0.1002 - val_accuracy: 0.9735
Epoch 61/100
147/147 [=====] - 10s 65ms/step - loss: 0.0068 - accuracy: 0.9998 - val_loss: 0.0951 - val_accuracy: 0.9770
Epoch 62/100
147/147 [=====] - 10s 66ms/step - loss: 0.0062 - accuracy: 1.0000 - val_loss: 0.1036 - val_accuracy: 0.9701
Epoch 63/100
147/147 [=====] - 10s 65ms/step - loss: 0.0062 - accuracy: 0.9998 - val_loss: 0.1074 - val_accuracy: 0.9727
Epoch 64/100
147/147 [=====] - 10s 66ms/step - loss: 0.0060 - accuracy: 0.9998 - val_loss: 0.0957 - val_accuracy: 0.9770
Epoch 65/100
147/147 [=====] - 10s 66ms/step - loss: 0.0057 - accuracy: 0.9998 - val_loss: 0.1313 - val_accuracy: 0.9659
Epoch 66/100
147/147 [=====] - 10s 66ms/step - loss: 0.0072 - accuracy: 0.9998 - val_loss: 0.1006 - val_accuracy: 0.9761
Epoch 67/100
147/147 [=====] - 10s 65ms/step - loss: 0.0058 - accuracy: 0.9998 - val_loss: 0.1022 - val_accuracy: 0.9753
Epoch 68/100
147/147 [=====] - 10s 65ms/step - loss: 0.0056 - accuracy: 1.0000 - val_loss: 0.1004 - val_accuracy: 0.9753
Epoch 69/100
147/147 [=====] - 10s 66ms/step - loss: 0.0047 - accuracy: 1.0000 - val_loss: 0.1055 - val_accuracy: 0.9710
Epoch 70/100
147/147 [=====] - 10s 66ms/step - loss: 0.0047 - accuracy: 1.0000 - val_loss: 0.1090 - val_accuracy: 0.9718
Epoch 71/100
147/147 [=====] - 10s 65ms/step - loss: 0.0047 - accuracy: 1.0000 - val_loss: 0.1058 - val_accuracy: 0.9744
Epoch 72/100
147/147 [=====] - 10s 66ms/step - loss: 0.0054 - accuracy: 1.0000 - val_loss: 0.1174 - val_accuracy: 0.9710
Epoch 73/100
147/147 [=====] - 10s 65ms/step - loss: 0.0053 - accuracy: 0.9998 - val_loss: 0.1061 - val_accuracy: 0.9761
Epoch 74/100
147/147 [=====] - 10s 65ms/step - loss: 0.0043 - accuracy: 1.0000 - val_loss: 0.1115 - val_accuracy: 0.9718
Epoch 75/100
147/147 [=====] - 10s 65ms/step - loss: 0.0045 - accuracy: 1.0000 - val_loss: 0.1042 - val_accuracy: 0.9753
Epoch 76/100
147/147 [=====] - 10s 65ms/step - loss: 0.0065 - accuracy: 0.9994 - val_loss: 0.1071 - val_accuracy: 0.9744
Epoch 77/100
147/147 [=====] - 10s 66ms/step - loss: 0.0043 - accuracy: 1.0000 - val_loss: 0.1119 - val_accuracy: 0.9718
Epoch 78/100
147/147 [=====] - 10s 65ms/step - loss: 0.0037 - accuracy: 1.0000 - val_loss: 0.1121 - val_accuracy: 0.9735
Epoch 79/100
147/147 [=====] - 10s 66ms/step - loss: 0.0065 - accuracy: 0.9989 - val_loss: 0.1177 - val_accuracy: 0.9735
Epoch 80/100
147/147 [=====] - 10s 65ms/step - loss: 0.0038 - accuracy: 0.9998 - val_loss: 0.1113 - val_accuracy: 0.9761
Epoch 81/100
147/147 [=====] - 10s 65ms/step - loss: 0.0043 - accuracy: 1.0000 - val_loss: 0.1083 - val_accuracy: 0.9770
Epoch 82/100
147/147 [=====] - 10s 65ms/step - loss: 0.0032 - accuracy: 1.0000 - val_loss: 0.1102 - val_accuracy: 0.9761
Epoch 83/100
147/147 [=====] - 10s 65ms/step - loss: 0.0036 - accuracy: 1.0000 - val_loss: 0.1150 - val_accuracy: 0.9744
Epoch 84/100
147/147 [=====] - 10s 66ms/step - loss: 0.0030 - accuracy: 1.0000 - val_loss: 0.1154 - val_accuracy: 0.9753
Epoch 85/100
147/147 [=====] - 10s 66ms/step - loss: 0.0026 - accuracy: 1.0000 - val_loss: 0.1183 - val_accuracy: 0.9735
Epoch 86/100
147/147 [=====] - 10s 65ms/step - loss: 0.0028 - accuracy: 1.0000 - val_loss: 0.1158 - val_accuracy: 0.9727
Epoch 87/100
147/147 [=====] - 10s 66ms/step - loss: 0.0027 - accuracy: 1.0000 - val_loss: 0.1133 - val_accuracy: 0.9753
Epoch 88/100
147/147 [=====] - 10s 66ms/step - loss: 0.0025 - accuracy: 1.0000 - val_loss: 0.1141 - val_accuracy: 0.9761
Epoch 89/100
147/147 [=====] - 10s 65ms/step - loss: 0.0024 - accuracy: 1.0000 - val_loss: 0.1178 - val_accuracy: 0.9744
Epoch 90/100
147/147 [=====] - 10s 65ms/step - loss: 0.0023 - accuracy: 1.0000 - val_loss: 0.1189 - val_accuracy: 0.9735
Epoch 91/100
147/147 [=====] - 10s 65ms/step - loss: 0.0026 - accuracy: 1.0000 - val_loss: 0.1188 - val_accuracy: 0.9735

```

Epoch 92/100
147/147 [=====] - 10s 65ms/step - loss: 0.0029 - accuracy: 1.0000 - val_loss: 0.1139 - val_accuracy: 0.9744
Epoch 93/100
147/147 [=====] - 10s 66ms/step - loss: 0.0026 - accuracy: 1.0000 - val_loss: 0.1202 - val_accuracy: 0.9735
Epoch 94/100
147/147 [=====] - 10s 66ms/step - loss: 0.0022 - accuracy: 1.0000 - val_loss: 0.1155 - val_accuracy: 0.9761
Epoch 95/100
147/147 [=====] - 10s 66ms/step - loss: 0.0022 - accuracy: 1.0000 - val_loss: 0.1174 - val_accuracy: 0.9761
Epoch 96/100
147/147 [=====] - 10s 66ms/step - loss: 0.0028 - accuracy: 1.0000 - val_loss: 0.1203 - val_accuracy: 0.9761
Epoch 97/100
147/147 [=====] - 10s 66ms/step - loss: 0.0021 - accuracy: 1.0000 - val_loss: 0.1201 - val_accuracy: 0.9753
Epoch 98/100
147/147 [=====] - 10s 66ms/step - loss: 0.0034 - accuracy: 0.9996 - val_loss: 0.2196 - val_accuracy: 0.9488
Epoch 99/100
147/147 [=====] - 10s 66ms/step - loss: 0.0162 - accuracy: 0.9934 - val_loss: 0.1327 - val_accuracy: 0.9735
Epoch 100/100
147/147 [=====] - 10s 65ms/step - loss: 0.0020 - accuracy: 1.0000 - val_loss: 0.1323 - val_accuracy: 0.9744
Training time: -975.6852314472198
118/118 [=====] - 3s 25ms/step - loss: 0.1323 - accuracy: 0.9744
[INFO] loss=0.1323, accuracy: 97.4403%

```

```

In [15]: (loss, accuracy) = custom_resnet_model.evaluate(X_test, y_test, batch_size=10, verbose=1)

print("[INFO] loss={:.4f}, accuracy: {:.4f}%".format(loss, accuracy * 100))

118/118 [=====] - 3s 23ms/step - loss: 0.1323 - accuracy: 0.9744
[INFO] loss=0.1323, accuracy: 97.4403%

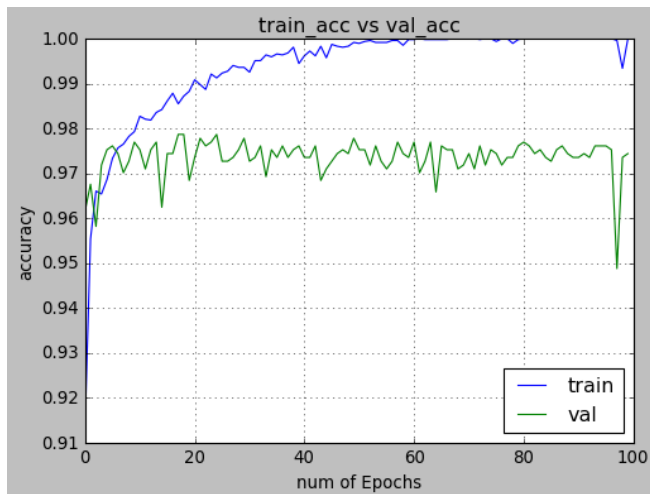
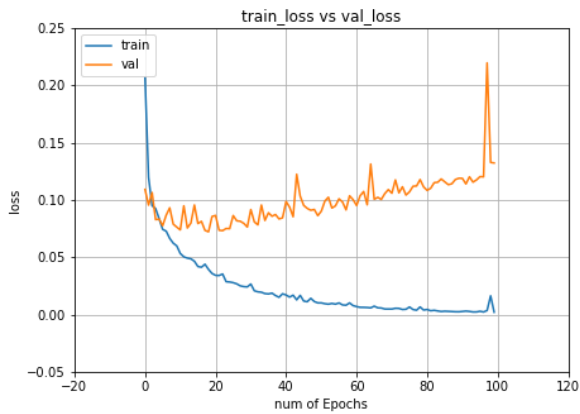
```

visualizing losses and accuracy

```

In [16]: display_loss_accuracy(hist)

```



Selection of appropriate layers

```
In [50]: train_image_generator = ImageDataGenerator(
        rescale=1./255,
        validation_split = 0.2,
        shear_range=0.2,
        zoom_range=0.2,
    )

    train_generator = train_image_generator.flow_from_dataframe(
        dataframe=ftd,
        directory=train_img_dir,
        x_col='X_ray_image_name',
        y_col='target',
        target_size=(224, 224),
        batch_size=16,
        seed=2020,
        shuffle=True,
        class_mode='binary'
    )

    validation_generator = train_image_generator.flow_from_dataframe(
        dataframe=final_validation_data,
        directory=train_img_dir,
        x_col='X_ray_image_name',
        y_col='target',
        target_size=(224, 224),
        batch_size=16,
        seed=2020,
        shuffle=True,
        class_mode='binary'
    )
```

Found 0 validated image filenames belonging to 0 classes.
Found 0 validated image filenames belonging to 0 classes.

Measurement of models performance

Convolutional Neural Network

```
In [56]: model = Sequential()
        model.add(Conv2D(32, (3, 3), input_shape= (224,224,3)))
        model.add(Activation('relu'))
        model.add(MaxPooling2D(pool_size=(2, 2)))

        model.add(Conv2D(32, (3, 3)))
        model.add(Activation('relu'))
        model.add(MaxPooling2D(pool_size=(2, 2)))

        model.add(Conv2D(64, (3,3)))
        model.add(Activation("relu"))
        model.add(Conv2D(250,(3,3)))
        model.add(Dropout(0.5))
        model.add(Activation("relu"))

        model.add(Conv2D(128,(3,3)))
        model.add(Activation("relu"))
        model.add(AvgPool2D(2,2))
        model.add(Conv2D(64,(3,3)))
        model.add(Activation("relu"))
        model.add(AvgPool2D(2,2))

        model.add(Conv2D(256,(2,2)))
        model.add(Activation("relu"))
        model.add(MaxPool2D(2,2))

        model.add(Flatten())
        model.add(Dense(32))
        model.add(Dropout(0.5))
        model.add(Dense(1))
        model.add(Activation("sigmoid"))
```

```
In [67]: model.summary()
```

Model: "sequential_2"

Layer (type)	Output Shape	Param #
=====		
conv2d_1 (Conv2D)	(None, 222, 222, 32)	896
activation (Activation)	(None, 222, 222, 32)	0
max_pooling2d (MaxPooling2D)	(None, 111, 111, 32)	0
conv2d_2 (Conv2D)	(None, 109, 109, 32)	9248
activation_1 (Activation)	(None, 109, 109, 32)	0
max_pooling2d_1 (MaxPooling2D)	(None, 54, 54, 32)	0
conv2d_3 (Conv2D)	(None, 52, 52, 64)	18496
activation_2 (Activation)	(None, 52, 52, 64)	0
conv2d_4 (Conv2D)	(None, 50, 50, 250)	144250
dropout_4 (Dropout)	(None, 50, 50, 250)	0
activation_3 (Activation)	(None, 50, 50, 250)	0
conv2d_5 (Conv2D)	(None, 48, 48, 128)	288128
activation_4 (Activation)	(None, 48, 48, 128)	0
average_pooling2d (AveragePooling2D)	(None, 24, 24, 128)	0
conv2d_6 (Conv2D)	(None, 22, 22, 64)	73792
activation_5 (Activation)	(None, 22, 22, 64)	0
average_pooling2d_1 (AveragePooling2D)	(None, 11, 11, 64)	0
conv2d_7 (Conv2D)	(None, 10, 10, 256)	65792
activation_6 (Activation)	(None, 10, 10, 256)	0
max_pooling2d_2 (MaxPooling2D)	(None, 5, 5, 256)	0
flatten (Flatten)	(None, 6400)	0
dense_5 (Dense)	(None, 32)	204832
dropout_5 (Dropout)	(None, 32)	0
dense_6 (Dense)	(None, 1)	33
activation_7 (Activation)	(None, 1)	0
=====		
Total params: 805,467		
Trainable params: 805,467		
Non-trainable params: 0		


```
In [18]: test_image_path = 'C:/UOFT/3546_TermProject/covid/DL-final-project/Chest_xray_seperate/PNEUMONIA/person11_bacteria_45.jpeg'
test_image = cv2.imread(test_image_path)
if ( num_channel == 1):
    test_image=cv2.cvtColor(test_image, cv2.COLOR_BGR2GRAY)
test_image=cv2.resize(test_image,(128,128))
test_image = np.array(test_image)
test_image = test_image.astype('float32')
test_image /= 255
print (test_image.shape)
test_image_nd = np.expand_dims(test_image, axis=0)
print (test_image_nd.shape)
print (test_image.shape)
# if num_channel==1:
#     if (K.image_data_format() == 'channels_first'):
#         test_image= np.expand_dims(test_image, axis=0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=3)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
# else:
#     if (K.image_data_format() == 'channels_first'):
#         test_image=np.rollaxis(test_image,2,0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)

# Predicting the test image
print((model.predict(test_image_nd)))
print(model.predict_classes(test_image_nd))

(128, 128, 3)
(1, 128, 128, 3)
(128, 128, 3)
[[0.08644115 0.91355884]]
[1]
```

```
In [18]: test_image_path = 'C:/UOFT/3546_TermProject/covid/DL-final-project/Chest_xray_seperate/PNEUMONIA/person11_bacteria_45.jpeg'
test_image = cv2.imread(test_image_path)
if ( num_channel == 1):
    test_image=cv2.cvtColor(test_image, cv2.COLOR_BGR2GRAY)
test_image=cv2.resize(test_image,(128,128))
test_image = np.array(test_image)
test_image = test_image.astype('float32')
test_image /= 255
print (test_image.shape)
test_image_nd = np.expand_dims(test_image, axis=0)
print (test_image_nd.shape)
print (test_image.shape)
# if num_channel==1:
#     if (K.image_data_format() == 'channels_first'):
#         test_image= np.expand_dims(test_image, axis=0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=3)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
# else:
#     if (K.image_data_format() == 'channels_first'):
#         test_image=np.rollaxis(test_image,2,0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)

# Predicting the test image
print((model.predict(test_image_nd)))
print(model.predict_classes(test_image_nd))

(128, 128, 3)
(1, 128, 128, 3)
(128, 128, 3)
[[0.08644115 0.91355884]]
[1]
```

```

In [18]: test_image_path = 'C:/UOFT/3546_TermProject/covid/DL-final-project/Chest_xray_seperate/PNEUMONIA/person11_bacteria_45.jpeg'
test_image = cv2.imread(test_image_path)
if ( num_channel == 1):
    test_image=cv2.cvtColor(test_image, cv2.COLOR_BGR2GRAY)
test_image=cv2.resize(test_image,(128,128))
test_image = np.array(test_image)
test_image = test_image.astype('float32')
test_image /= 255
print (test_image.shape)
test_image_nd = np.expand_dims(test_image, axis=0)
print (test_image_nd.shape)
print (test_image.shape)
# if num_channel==1:
#     if (K.image_data_format() == 'channels_first'):
#         test_image= np.expand_dims(test_image, axis=0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=3)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
# else:
#     if (K.image_data_format() == 'channels_first'):
#         test_image=np.rollaxis(test_image,2,0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)

# Predicting the test image
print((model.predict(test_image_nd)))
print(model.predict_classes(test_image_nd))

(128, 128, 3)
(1, 128, 128, 3)
(128, 128, 3)
[[0.08644115 0.91355884]]
[1]

```

efficient Net80 model

```
In [17]: score = custom_resnet_model.evaluate(X_test, y_test, verbose=0)
print('Test Loss:', score[0])
print('Test accuracy:', score[1])

test_image = X_test[0:1]
print (test_image.shape)

print(model.predict(test_image))
print(model.predict_classes(test_image))
print(y_test[0:1])
```

Test Loss: 0.09007590264081955

Test accuracy: 0.9692832827568054

(1, 224, 224, 3)

[[1.14373943e-05 1.04924751e-04 2.51722900e-04 4.77186404e-04
1.18551354e-04 2.09018617e-05 4.94372798e-05 3.72421637e-05
2.81130688e-05 3.33237003e-05 5.39718130e-05 4.48332175e-05
1.52935299e-05 2.69304637e-05 9.17184298e-06 1.39459144e-05
3.57407189e-05 1.11406243e-05 9.04056869e-05 1.29782620e-05
4.97367182e-05 1.52026812e-04 1.29650245e-04 9.28172449e-05
5.12071965e-05 1.26058385e-05 2.34643376e-05 2.12379255e-05
1.80296520e-05 4.98396868e-04 2.47400421e-05 6.25148459e-05
1.78604005e-05 5.69082549e-05 7.87051613e-05 7.06591309e-05
2.68311251e-05 3.09490679e-05 3.14168792e-05 2.49381701e-05
3.04195401e-05 1.12252492e-05 7.05893126e-06 9.69405119e-06
2.52910850e-05 3.18797393e-05 5.48759344e-05 8.84422025e-06
5.99929308e-06 3.62554201e-05 4.74188018e-05 2.52091377e-05
1.06253538e-04 5.40399014e-05 7.13719346e-05 5.98135775e-05
2.53936560e-05 3.17523663e-05 3.84590785e-05 1.80896059e-05
1.09258735e-05 6.93523816e-06 3.70208436e-05 3.30621988e-05
3.41156119e-05 4.84199654e-06 4.28685780e-05 6.73844215e-06
3.28146380e-05 1.46650229e-04 3.31878364e-05 2.56108760e-04
2.61196430e-04 4.09581007e-05 7.56776790e-05 1.36237679e-04
9.20068051e-05 2.07817684e-05 1.77334645e-04 1.37647934e-04
4.13984781e-05 7.86930832e-05 4.24302016e-05 1.24560480e-04
2.35329808e-05 5.86002352e-06 2.18944788e-05 8.65192269e-05
8.45431496e-05 3.85376101e-04 9.83447353e-06 7.61144838e-05
3.42622370e-05 7.31325781e-05 1.02619080e-04 2.22032431e-05
5.87798240e-05 4.03036538e-05 6.36094046e-06 1.85568555e-04
1.04721134e-04 1.46556995e-03 2.70735891e-05 2.10290254e-05
9.79486140e-06 2.09725858e-05 4.52680506e-05 1.53035537e-04
6.04435445e-05 2.56030016e-05 1.71550928e-05 3.08110699e-04
1.67387456e-03 1.12096459e-04 2.55914565e-05 9.76902083e-06
1.16005620e-04 4.60985902e-04 2.48370834e-05 3.65114938e-05
3.35951845e-05 6.50190414e-05 1.97253947e-04 6.78287470e-05
8.86783164e-05 3.39203070e-05 7.83100375e-04 1.23239923e-04
1.50638793e-04 4.32174493e-05 9.05354536e-05 3.12414595e-05
5.16333908e-04 1.17185227e-04 1.94612760e-04 1.78226008e-04
2.17185643e-05 9.77008312e-06 3.03590878e-05 2.52094997e-05
1.36422619e-04 1.50249689e-05 5.26125150e-05 5.74600926e-05
3.14157369e-05 1.09026314e-05 9.86417872e-05 3.10784380e-04
1.04601844e-04 1.25339080e-04 4.14308270e-05 6.06628091e-05
5.27289485e-05 2.94673984e-04 2.61931302e-04 1.45342143e-04
1.05807107e-04 1.62024109e-04 1.02612219e-04 1.64236335e-04
1.54185636e-05 2.57844367e-04 4.59984876e-04 4.87738871e-04
4.82647192e-05 2.09362479e-04 3.35372133e-05 1.07843816e-05
1.17915581e-04 4.03424929e-05 4.12247355e-05 4.26680490e-04
3.09097348e-04 6.37048142e-05 1.60922136e-04 1.90144812e-04
2.21473274e-05 2.15525924e-05 7.02993711e-04 6.90434390e-05
7.28029408e-05 2.49030454e-05 7.60511393e-05 4.34630056e-05
9.78316893e-05 9.55160285e-05 4.04321967e-04 6.89308508e-04
1.97458630e-05 2.51068850e-05 1.48604340e-05 2.13776177e-04
2.38938854e-04 4.88295482e-05 3.06138027e-05 5.57878593e-05
1.87989892e-04 1.52984532e-04 6.18642298e-05 1.30006025e-04
1.00618592e-04 3.20101281e-05 7.87851386e-05 1.01110712e-03
4.70257364e-05 7.69894468e-05 6.71668895e-05 1.08402739e-04
1.48987558e-04 7.06426566e-04 4.69786319e-05 3.17751765e-05
7.18288866e-05 7.85148150e-05 3.43337670e-05 1.47799292e-04
4.15880677e-05 9.35140924e-05 1.99519072e-05 1.35406372e-04
5.93844561e-05 4.66298407e-05 3.48248577e-05 1.75946974e-04
4.10317414e-04 7.20948156e-05 1.07296561e-04 9.15680721e-05
2.65757753e-05 6.20490537e-05 3.57686280e-04 4.92307408e-05
7.88953694e-05 9.74631985e-05 9.77357340e-05 1.59143674e-04
8.07626202e-05 4.58606664e-05 8.00844646e-05 2.82172841e-05
3.24945031e-05 8.70945951e-05 2.81445507e-04 2.00479510e-04
2.22346243e-05 1.46532038e-04 8.37646658e-05 1.47363280e-05
9.21480416e-04 2.03847609e-04 3.35818448e-04 2.13845720e-04
2.01661867e-04 1.12781454e-04 3.24054534e-04 3.73305993e-05
1.21427081e-04 1.35603113e-04 3.97058117e-04 1.90519655e-04
3.32621072e-04 2.99248932e-05 1.27345229e-05 2.49134842e-04
2.39574438e-04 2.25008494e-04 3.33966855e-05 1.90712235e-05
1.51352024e-05 1.99000569e-04 6.48575297e-05 2.96788494e-05
2.58134296e-05 1.93665455e-05 7.48435268e-05 5.02380317e-05
1.33373513e-04 2.41820890e-05 1.29066058e-04 2.84416874e-05
3.16090336e-05 8.89335715e-05 8.93408433e-05 1.91875341e-04
6.35485980e-04 1.77661263e-04 4.06467661e-05 3.37279926e-05
8.29484998e-05 3.44925502e-05 5.00490896e-05 1.12600843e-04
4.83377298e-05 7.75073495e-05 1.90335577e-05 8.28797129e-05
1.28626212e-04 9.85138322e-05 8.34467792e-05 1.86040619e-04
4.66942365e-05 1.01368511e-04 5.72525059e-05 5.52038400e-05
1.83994271e-05 4.83073636e-05 1.90646770e-05 1.14161985e-05
4.23263737e-05 1.16538922e-05 9.98509058e-05 4.05804931e-05
1.06657615e-04 3.14800382e-05 6.28069160e-04 6.06823123e-05
2.66918050e-05 1.28088577e-04 4.45805636e-05 1.41036862e-05
4.66265519e-05 1.94223849e-05 1.66193277e-05 1.68263159e-05
9.03178734e-05 4.53277571e-05 2.43365394e-05 6.53736352e-05
3.84007653e-05 1.57341579e-04 9.59814570e-05 3.72657923e-05
5.87329232e-05 2.72484049e-05 2.13894164e-05 4.79126393e-05
1.06414780e-04 1.51311193e-04 7.75438675e-05 4.13377638e-05
5.77614446e-05 7.71465129e-05 3.19645551e-05 2.00346949e-05
3.26975896e-05 5.18276647e-05 1.59105966e-05 2.47692573e-04
7.79931652e-05 8.65568582e-05 7.72862404e-06 8.81144224e-05

3.00691881e-05 2.82963210e-05 2.39766759e-05 1.12137468e-05
1.36173767e-04 3.45422523e-05 5.21864567e-05 7.69211983e-05
1.09941693e-05 1.56334972e-05 1.81861269e-05 5.80719679e-05
1.18833368e-05 1.58374634e-04 3.19300343e-05 3.77782053e-05
4.08860615e-05 1.21223620e-05 8.06795506e-05 4.09699387e-05
3.04040703e-04 2.89627915e-05 3.15261823e-05 5.40487672e-05
4.86005119e-05 1.21615476e-05 3.92062429e-05 1.31830617e-04
3.85843887e-05 6.10611532e-05 4.50137995e-05 2.91373854e-05
2.22384824e-05 1.56864160e-04 9.57904267e-05 2.81273351e-05
8.35719220e-06 3.35304976e-05 7.48602979e-05 6.25614266e-05
1.53319361e-05 8.10955753e-05 3.23018867e-05 9.29296766e-06
1.25487904e-05 7.40313089e-06 2.78445892e-04 2.99154199e-03
3.78724711e-04 1.21161138e-04 1.08959524e-04 4.52505737e-05
1.70245068e-04 1.63981225e-04 1.11761401e-04 9.13805503e-04
2.76584015e-05 4.85704193e-04 8.96239453e-06 8.32033838e-05
7.59714749e-05 5.65359805e-05 1.73191249e-04 6.54873802e-05
2.07793521e-04 6.98733900e-04 1.35064911e-04 5.09257312e-04
1.95461107e-04 2.54977634e-03 1.84799355e-04 1.79246435e-05
2.23795490e-04 1.17462705e-05 8.37473490e-05 4.66217345e-04
9.20921302e-05 6.95714916e-05 5.27464772e-06 8.62578582e-03
1.39323893e-04 4.93507432e-05 1.76188332e-04 2.94449273e-02
1.55619404e-04 1.01388898e-03 1.95309069e-04 6.41724091e-06
1.14455390e-04 2.71182416e-05 6.47203298e-04 1.00095363e-04
6.33788586e-05 1.11424779e-04 7.55546498e-05 5.89844713e-04
6.43710955e-05 2.88552710e-05 9.80928168e-03 6.12233780e-05
1.06373584e-04 5.20272406e-05 1.53895508e-05 3.58710568e-05
3.59345358e-05 4.73626016e-04 1.26029961e-04 2.86329567e-04
2.38124921e-04 6.03371416e-04 3.55180528e-05 2.16321168e-05
2.72363777e-05 8.89033035e-05 4.06742422e-03 1.29921682e-04
2.86551396e-04 2.84782218e-05 1.06897111e-04 2.70382123e-04
1.81409428e-04 8.40800058e-05 4.40880867e-05 9.01982770e-04
1.81780735e-04 8.00840371e-06 1.01473459e-04 2.90756556e-03
7.21756514e-05 1.36662420e-05 6.18706963e-06 2.27368524e-04
7.00342061e-05 4.50232001e-05 3.86600768e-05 7.42896300e-05
1.27039864e-04 1.73732784e-04 5.75884114e-05 4.97821784e-05
3.84462801e-05 1.79283034e-05 3.22828419e-04 3.65375286e-06
1.82539625e-05 4.14854847e-04 2.76429375e-04 2.79783591e-04
1.49271495e-04 5.04715135e-04 7.00918376e-04 1.32299065e-05
2.96655082e-04 1.73062974e-04 3.41755454e-04 4.89280465e-05
1.67422098e-04 4.59228140e-05 4.93768748e-05 6.83523656e-04
3.65148531e-04 1.26141720e-04 2.51824094e-04 3.87691660e-04
2.50834078e-02 6.52788149e-05 7.25845341e-04 1.34653674e-04
1.08346785e-03 2.12903451e-05 7.40867181e-05 2.24470091e-03
9.96265298e-05 1.42169208e-03 9.08934817e-06 2.26477478e-05
1.45459187e-03 8.01711634e-04 1.98526774e-04 7.26202925e-05
3.08887684e-05 1.84962082e-05 8.66356830e-04 1.23538586e-04
2.15153977e-05 2.23801049e-04 7.42496850e-05 6.04413508e-05
2.95608315e-05 1.18541066e-04 5.44077542e-04 4.79755603e-04
1.19070751e-06 2.86531867e-04 1.08347449e-04 1.23221835e-04
1.60122472e-05 3.52789066e-05 7.62058926e-06 1.49062726e-06
1.97045141e-04 3.07448827e-05 2.99867289e-03 1.13841954e-04
4.60953343e-05 1.01471902e-03 9.77512900e-05 2.65716197e-04
2.96324670e-05 1.18826174e-04 1.49232650e-03 3.68589041e-04
3.39436228e-04 1.83823809e-04 3.75682794e-05 6.95802446e-05
7.74253858e-04 2.01023329e-04 1.54460280e-03 2.88035953e-04
2.01830422e-04 5.24513925e-05 5.96768878e-05 3.64893567e-05
8.01541464e-05 2.58367072e-04 1.18391141e-02 2.76369654e-04
7.56522408e-04 1.20333594e-03 1.34738511e-04 3.40762839e-04
1.82323365e-05 7.10968015e-05 1.12378566e-05 5.10631944e-04
1.91196290e-04 1.09206350e-03 3.15800498e-05 2.18468718e-04
1.12467869e-05 4.41428600e-03 2.27257988e-04 6.61191953e-05
4.01627585e-05 1.24009266e-05 2.44076873e-05 4.03825616e-05
4.58932802e-04 1.87291380e-03 3.13152181e-04 3.02730768e-05
4.86531015e-03 1.40191099e-04 3.30081070e-03 1.58697047e-04
3.27433518e-04 4.06419967e-05 2.01998424e-04 1.32928126e-05
4.85229393e-05 1.75380628e-05 2.26789562e-04 4.04121878e-04
2.61310168e-04 4.74795932e-04 1.89746468e-04 3.73692531e-03
3.27527232e-04 9.13831318e-05 3.24898690e-04 4.24559839e-05
1.04334940e-04 2.81458633e-04 3.25866567e-04 3.70700407e-04
9.77324744e-05 4.96280300e-05 4.70058789e-04 7.70517101e-04
1.39880329e-04 3.93471419e-04 4.72605170e-05 1.75979440e-05
2.38323264e-05 3.17902995e-05 2.39583111e-04 4.03178186e-04
4.79454611e-05 7.74909800e-04 1.22122496e-04 7.42310367e-04
4.16312163e-04 4.45918304e-05 2.92662182e-04 1.42756937e-04
3.69849222e-05 1.75505877e-04 1.69684878e-04 2.01773928e-05
5.17364060e-05 8.94230106e-05 6.82292040e-04 5.67962707e-04
9.67533968e-04 1.49089133e-03 1.81955820e-05 2.01489129e-05
6.49139620e-05 2.10260070e-04 1.31405532e-05 2.13075604e-04
4.66043202e-05 1.07829161e-04 1.37850569e-04 2.17440233e-04
9.95940281e-05 1.59908854e-03 4.40408097e-04 9.27102883e-05
1.46864229e-04 1.02436473e-03 1.35910641e-05 5.05308453e-05
9.87842668e-06 3.81041842e-04 1.07388140e-03 5.56326668e-05
8.97882041e-04 8.16292595e-05 3.35576275e-04 5.29770681e-04
2.01572650e-04 3.38251411e-05 1.80425523e-05 5.76800609e-04
9.82797137e-05 1.02833973e-03 5.09234742e-05 2.18567718e-03
6.12997974e-05 3.24978668e-04 1.80621195e-04 3.68183901e-05
1.51478671e-04 4.82440781e-04 6.04991401e-05 7.77881432e-05
1.26551939e-04 1.84923367e-04 3.15761950e-04 5.93294317e-05
6.60373989e-05 6.65641564e-05 1.59843130e-05 1.58041785e-05
5.12444996e-04 4.13116722e-05 1.11431109e-05 1.06844418e-04
2.59616121e-04 2.77282033e-05 1.40357995e-04 1.17170248e-05

1.01555357e-04 7.50477266e-05 4.15015711e-05 6.11421565e-05
1.11586787e-03 1.75463210e-04 2.01662551e-04 2.15432628e-05
1.05642561e-04 8.10383644e-04 1.73605731e-05 7.83597934e-04
1.32822432e-03 1.24145346e-03 2.37979722e-04 1.58674651e-04
3.66873719e-05 2.97207967e-04 6.27743066e-05 5.37725755e-05
5.02291659e-05 4.35538124e-04 5.03882147e-05 5.22157527e-04
2.6369772e-05 3.11685086e-04 7.64837372e-04 1.75272813e-03
6.76614523e-04 5.19993082e-05 4.07306979e-05 1.40892621e-03
6.39285936e-05 3.50955548e-03 1.14453929e-02 2.54656974e-04
1.06379361e-04 2.97680050e-01 2.45035899e-05 3.06748225e-05
1.32625370e-04 4.29442771e-05 6.94501068e-05 1.06415697e-04
1.08547676e-04 6.69606961e-04 5.19007190e-05 1.91055569e-05
2.42599664e-04 5.64372109e-04 4.39090800e-05 1.56143127e-04
4.16793482e-05 8.86967537e-05 2.07324396e-04 3.20165709e-05
1.20084711e-04 3.51153547e-04 1.97403730e-04 2.12631872e-04
1.54889669e-04 6.05181776e-05 1.16133422e-03 5.36538319e-05
5.46015275e-04 9.81851845e-05 7.44502031e-05 4.70522762e-04
1.39306416e-04 4.22805641e-03 9.15047858e-05 2.49385266e-05
5.90733907e-05 4.27730993e-04 2.60059896e-05 5.41474583e-05
2.61146983e-04 8.86572598e-05 3.34721953e-01 3.97562602e-04
2.51476711e-04 1.31613610e-03 4.18234449e-05 2.01638992e-04
2.66266416e-05 8.53372330e-05 6.14622666e-04 2.26133983e-04
9.31883405e-05 8.49861390e-05 9.05829074e-04 3.95126153e-05
2.19148878e-05 8.25024472e-05 2.39794645e-05 1.10723381e-03
7.13291302e-05 2.36116393e-04 2.40695794e-04 2.60236062e-04
1.50290534e-05 3.14099132e-04 2.34945110e-04 4.29183965e-05
1.69071529e-04 2.40478839e-04 4.90417042e-05 5.88735391e-04
3.61158673e-05 1.56817423e-05 5.22980590e-05 5.17263907e-05
1.26827363e-04 7.73858483e-05 6.48025365e-04 1.14715646e-03
3.61663464e-04 4.45229030e-04 2.72112986e-04 2.87087892e-06
3.61288403e-04 6.21178304e-04 2.04670141e-05 7.67436577e-05
8.71896627e-05 6.25145214e-04 3.63889936e-04 1.31533103e-04
3.38180194e-04 1.70185650e-03 2.65352952e-04 1.38429314e-04
9.57641714e-06 5.85757953e-04 2.93099583e-04 2.71516124e-04
1.10013694e-04 2.12021041e-05 5.32478502e-04 1.58851806e-04
7.28346786e-05 7.45117213e-05 5.06315439e-04 7.12172943e-04
2.33235605e-05 4.29070642e-04 2.58522086e-05 1.45645684e-03
1.60161668e-04 2.95481477e-05 1.37279043e-04 6.89423759e-05
2.97003971e-05 1.15219445e-04 3.95013085e-05 7.65176228e-05
8.95155099e-05 1.20341807e-04 1.30225468e-04 9.33943084e-05
6.54157624e-03 2.40754762e-05 4.81955431e-05 4.00307385e-04
6.05372363e-04 3.09340947e-04 2.67484138e-04 1.92242232e-03
1.71189918e-03 1.99379836e-04 3.41214036e-05 8.55870836e-04
1.34391375e-04 9.12115065e-05 2.63787092e-06 2.00020131e-05
2.3024142e-04 4.57847527e-05 6.59130455e-04 1.43710384e-03
1.65375485e-03 9.48748726e-04 7.93987885e-04 7.88530568e-04
1.26306771e-03 1.57873219e-05 6.23157975e-05 5.07640434e-05
1.00583668e-04 2.63427868e-02 2.79214786e-04 4.11254296e-04
2.00241338e-03 2.15464479e-05 2.20117727e-04 5.20894673e-06
1.06383231e-04 3.89995228e-04 1.18828044e-04 1.47819592e-05
1.56530696e-05 1.03744633e-05 8.23242299e-04 1.07548170e-04
8.51472083e-04 1.68794504e-05 6.11567113e-04 1.31508459e-05
1.45642407e-05 1.70212661e-05 4.47813227e-06 2.96817211e-06
7.97575922e-06 9.19564045e-04 5.69658412e-04 1.06845036e-04
1.38349315e-05 6.94129221e-06 3.45503300e-04 4.58483337e-05
5.70481236e-04 3.14873498e-04 2.07158388e-04 7.35614376e-05
3.47069254e-05 2.23567567e-05 3.26645531e-04 2.17293986e-04
6.19800921e-05 6.93862268e-04 2.12625891e-05 3.41955456e-05
7.15996284e-05 4.02755650e-05 1.31912733e-04 1.87830556e-05
4.05132487e-05 8.79637955e-05 1.04369083e-03 1.91554471e-04
3.08821982e-05 3.44705804e-05 9.25602726e-05 2.59073931e-05
1.07711840e-05 6.71665708e-04 6.75932461e-05 6.22531879e-05
4.64484265e-06 7.00603268e-05 2.42308190e-04 1.96481833e-05
3.98481963e-04 2.74188315e-05 1.39456693e-04 5.24139730e-04
5.99602470e-04 9.94585389e-06 1.29866559e-04 8.90876327e-05
2.63735164e-05 2.77296203e-04 3.93479859e-05 9.76353040e-05
6.76064592e-05 3.16987825e-05 2.46331724e-03 8.19425768e-05
9.37245277e-05 1.14760885e-04 3.52426905e-05 9.93357971e-04
5.93172399e-06 1.44978758e-05 2.35269217e-05 2.61552759e-05
2.09589707e-05 6.33978652e-06 2.15005775e-06 5.56230952e-06
1.75527548e-05 3.31104275e-06 1.74388711e-04 8.14965926e-04]]

```
-----  
AttributeError                                Traceback (most recent call last)  
<ipython-input-17-83421ec204f5> in <module>  
      7  
      8 print(model.predict(test_image))  
----> 9 print(model.predict_classes(test_image))  
     10 print(y_test[0:1])  
  
AttributeError: 'Functional' object has no attribute 'predict_classes'
```

Confusion matrix

```
In [20]: Y_pred = custom_resnet_model.predict(X_test)
print(Y_pred)
y_pred = np.argmax(Y_pred, axis=1)
print(y_pred)
#y_pred = model.predict_classes(X_test)
#print(y_pred)
target_names = ['class 0(Normal)', 'class 1(Pneumonia)']
print(classification_report(np.argmax(y_test,axis=1), y_pred,target_names=target_names))
print(confusion_matrix(np.argmax(y_test,axis=1), y_pred))
```

```
[[6.5542844e-07 9.9999940e-01]
 [9.9147195e-01 8.5280919e-03]
 [1.9132092e-05 9.9998093e-01]
 ...
 [2.7467880e-07 9.9999976e-01]
 [8.9294559e-08 9.9999988e-01]
 [9.9940789e-01 5.9217890e-04]]
[1 0 1 ... 1 1 0]
```

	precision	recall	f1-score	support
class 0(Normal)	0.95	0.94	0.94	312
class 1(Pneumonia)	0.98	0.98	0.98	860
accuracy			0.97	1172
macro avg	0.96	0.96	0.96	1172
weighted avg	0.97	0.97	0.97	1172

```
[[293 19]
 [ 17 843]]
```

Inception V3 model

```
In [17]: score = custom_resnet_model.evaluate(X_test, y_test, verbose=0)
print('Test Loss:', score[0])
print('Test accuracy:', score[1])

test_image = X_test[0:1]
print (test_image.shape)

print(model.predict(test_image))
print(model.predict_classes(test_image))
print(y_test[0:1])
```


$(1, 224, 224, 3)$

[illegible]

```
AttributeError                                Traceback (most recent call last)
<ipython-input-17-83421ec204f5> in <module>
      7
      8 print(model.predict(test_image))
---->  9 print(model.predict_classes(test_image))
     10 print(y_test[0:1])

AttributeError: 'Functional' object has no attribute 'predict_classes'
```

Testing a new image

```
In [18]: test_image_path = 'D:/Harold/MyDNN/DataSet/Chest_xray_seperate/PNEUMONIA/person11_bacteria_45.jpeg'
test_image = image.load_img(test_image_path, target_size=(224, 224))
x = image.img_to_array(test_image)
x = np.expand_dims(x, axis=0)
x = preprocess_input(x)
print (x.shape)
```

```
# if num_channel==1:
#     if (K.image_data_format() == 'channels_first'):
#         test_image= np.expand_dims(test_image, axis=0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=3)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
# else:
#     if (K.image_data_format() == 'channels_first'):
#         test_image=np.rollaxis(test_image,2,0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)

# Predicting the test image
yhat = custom_resnet_model.predict(x)
print(yhat)
# print(custom_resnet_model.predict_classes(x))
label = decode_predictions(yhat)
# retrieve the most likely result, e.g. highest probability
label = label[0][0]
```

```
(1, 224, 224, 3)
[[7.9543006e-20 1.0000000e+00]]
```

```
-----
ValueError                                Traceback (most recent call last)
<ipython-input-18-505048f79341> in <module>
    30 print(yhat)
    31 # print(custom_resnet_model.predict_classes(x))
--> 32 label = decode_predictions(yhat)
    33 # retrieve the most likely result, e.g. highest probability
    34 label = label[0][0]
```

```
D:\Anaconda3\lib\site-packages\tensorflow\python\keras\applications\inception_v3.py in decode_predictions(preds, top)
    412 @keras_export('keras.applications.inception_v3.decode_predictions')
    413 def decode_predictions(preds, top=5):
--> 414     return imagenet_utils.decode_predictions(preds, top=top)
    415
    416
```

```
D:\Anaconda3\lib\site-packages\tensorflow\python\keras\applications\imagenet_utils.py in decode_predictions(preds, top)
    149         'a batch of predictions '
    150         '(i.e. a 2D array of shape (samples, 1000)). '
--> 151         'Found array with shape: ' + str(preds.shape))
    152     if CLASS_INDEX is None:
    153         fpath = data_utils.get_file(
```

```
ValueError: `decode_predictions` expects a batch of predictions (i.e. a 2D array of shape (samples, 1000)). Found array with shape:
(1, 2)
```

```
In [20]: Y_pred = custom_resnet_model.predict(X_test)
print(Y_pred)
y_pred = np.argmax(Y_pred, axis=1)
print(y_pred)
#y_pred = model.predict_classes(X_test)
#print(y_pred)
target_names = ['class 0(Normal)', 'class 1(Pneumonia)']
print(classification_report(np.argmax(y_test,axis=1), y_pred,target_names=target_names))
print(confusion_matrix(np.argmax(y_test,axis=1), y_pred))

[[2.1685484e-10 1.0000000e+00]
 [9.9941623e-01 5.8379851e-04]
 [2.1206019e-14 1.0000000e+00]
 ...
 [3.4830153e-22 1.0000000e+00]
 [1.7426691e-19 1.0000000e+00]
 [8.4951813e-07 9.9999917e-01]]
[1 0 1 ... 1 1 1]

              precision    recall  f1-score   support

   class 0(Normal)         0.93         0.70         0.79         312
   class 1(Pneumonia)       0.90         0.98         0.94         860

    accuracy              0.90              1172
   macro avg              0.91              1172
   weighted avg           0.91              1172

[[217  95]
 [ 17 843]]
```

Compute confusion matrix

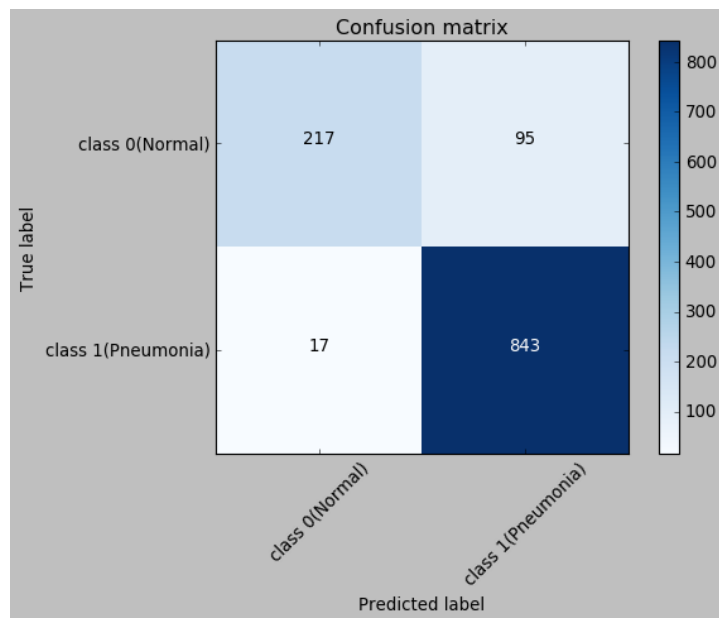
```
In [21]: # Compute confusion matrix
cnf_matrix = (confusion_matrix(np.argmax(y_test,axis=1), y_pred))

np.set_printoptions(precision=2)

# Plot non-normalized confusion matrix
plot_confusion_matrix(cnf_matrix, classes=target_names,
                      title='Confusion matrix')
```

Confusion matrix, without normalization

```
[[217  95]
 [ 17 843]]
```



Mobile NetV2 Model

```
In [17]: score = custom_resnet_model.evaluate(X_test, y_test, verbose=0)
print('Test Loss:', score[0])
print('Test accuracy:', score[1])

test_image = X_test[0:1]
print (test_image.shape)

print(model.predict(test_image))
print(model.predict_classes(test_image))
print(y_test[0:1])
```

Test Loss: 0.24162615835666656

Test accuracy: 0.914675772190094

(1, 224, 224, 3)

[[8.33914601e-05 9.43122213e-05 6.17418555e-05 6.54168252e-05
1.13233073e-04 4.74682347e-05 5.82300017e-05 1.08953674e-04
2.47885200e-05 1.91886254e-04 7.08437146e-05 3.57683566e-05
2.95793929e-04 5.55347469e-05 2.16536369e-04 6.21841609e-05
3.03070352e-04 4.57870956e-05 1.19120559e-04 8.90225492e-05
1.66406928e-04 9.70872716e-05 1.62744705e-04 1.49278203e-04
7.53177155e-05 1.09933528e-04 2.17667766e-04 1.03260376e-04
1.68310271e-05 4.71250714e-05 2.25656520e-04 1.11117632e-04
1.19079450e-04 3.50336049e-05 7.37054288e-05 1.12858543e-04
1.31092936e-04 5.13734412e-05 1.09661698e-04 1.33358277e-04
1.32257570e-04 3.87834370e-05 6.33041956e-04 8.69823198e-05
1.75937254e-04 4.36656082e-05 3.17085214e-04 4.07961088e-05
3.52774805e-05 9.89007895e-05 3.13197001e-04 3.97188778e-05
3.01094784e-04 2.42384500e-04 3.31761781e-04 3.03487395e-05
3.10513788e-05 1.01327387e-04 8.12207290e-04 2.79306027e-04
7.95677552e-05 3.58391902e-04 2.18163841e-04 1.20014038e-05
3.81811369e-05 1.32926434e-04 2.31196143e-04 2.24524265e-05
9.85946317e-05 1.02861639e-04 1.61079763e-04 2.17821958e-04
3.50604132e-05 2.44382943e-04 6.57923752e-04 1.51681917e-04
2.78034480e-04 1.11821643e-03 1.29941225e-04 7.72509258e-04
3.47995025e-04 1.38023504e-04 6.40542203e-05 4.19817152e-05
4.27196617e-04 4.55504705e-05 4.25938524e-05 6.52940753e-06
2.40640133e-04 1.50604916e-04 5.46822557e-04 1.12956704e-03
1.23571826e-03 6.57338824e-05 9.26602224e-05 4.90679340e-05
2.16124783e-04 5.13464547e-05 7.67093807e-05 4.40209726e-04
1.56303606e-04 3.04156449e-04 3.77853285e-05 1.33213616e-05
7.79672555e-05 1.12029833e-04 4.41315169e-05 5.07919642e-04
2.39844224e-03 1.99436821e-04 1.03430102e-04 1.63807563e-05
4.74627122e-05 1.43858822e-04 3.36380654e-05 9.86379018e-05
3.65483051e-04 2.97627383e-04 3.29633949e-05 1.55277128e-04
1.97401114e-05 5.45085131e-05 7.30986221e-06 7.17431467e-05
1.04610435e-05 1.34972914e-04 8.48182535e-05 5.58899483e-04
8.61329172e-05 1.34347181e-03 4.01100580e-04 9.56244621e-05
5.89205556e-05 7.04675767e-05 1.13356138e-04 1.90605904e-04
5.86017450e-05 2.83766531e-05 2.41395646e-05 2.35308864e-04
1.98521317e-04 9.12860487e-05 2.52444355e-04 1.62257536e-04
6.40077284e-04 1.32496934e-04 2.10071681e-04 6.35759934e-05
1.41753102e-04 1.35031165e-04 5.41926653e-04 3.66952663e-05
3.07578885e-05 6.28180278e-04 2.15198015e-05 8.01470087e-05
8.16880274e-05 8.74915640e-05 4.15920149e-05 7.44503195e-05
8.24786548e-05 5.42770031e-05 5.48054668e-05 4.74896988e-05
9.28548325e-05 4.52796885e-05 1.77980764e-05 1.84390505e-04
1.98084726e-05 1.04686733e-04 1.61515454e-05 1.20808099e-05
1.41941000e-05 6.96301358e-05 2.30870028e-05 1.98415233e-04
1.17952135e-04 5.10479294e-05 1.49995074e-04 8.08492041e-05
3.51375347e-05 2.29028607e-04 8.90064184e-05 3.12090066e-04
1.59218849e-04 4.08091466e-04 2.47155094e-05 6.27728441e-05
5.85914713e-05 1.24783706e-04 8.47743824e-04 2.00780225e-04
1.14579025e-05 4.24381542e-05 2.20518123e-04 2.41503993e-04
3.80996120e-04 1.66334023e-04 4.57426009e-04 2.28657053e-04
8.80086518e-05 2.42261885e-04 1.38663643e-04 1.01071002e-03
5.62849418e-05 1.17982512e-04 8.66682560e-04 2.85082497e-04
5.68611940e-05 3.75254785e-05 3.42668318e-05 9.50891044e-05
5.70256343e-05 1.05972678e-04 3.37440358e-03 1.68348161e-05
1.13453483e-03 9.97516909e-05 2.97156566e-05 9.10738425e-04
8.94142868e-05 3.80023732e-04 1.13284055e-04 1.53732763e-05
6.09609342e-05 4.30636464e-05 7.16900686e-05 1.03029714e-04
1.17967102e-04 2.91551783e-04 2.07727862e-05 2.68855474e-05
3.98764096e-04 1.12831323e-04 5.65429684e-04 1.08823864e-04
2.27547629e-04 8.15887543e-05 1.31144631e-03 1.72377404e-04
3.01062464e-05 1.25550898e-04 7.60590920e-05 1.35162845e-04
3.01215259e-05 7.59201357e-04 1.18775715e-04 1.13286223e-04
9.98448813e-05 5.13336236e-05 5.72519348e-05 2.53097154e-04
1.25184335e-04 2.11916649e-05 1.37296462e-04 3.80941747e-05
6.89057060e-06 1.19045166e-04 2.11454069e-04 4.46508348e-05
4.96068569e-05 4.19014759e-05 2.39833011e-04 1.74566085e-05
3.02299904e-05 6.74980110e-05 3.35405115e-04 9.43383129e-05
5.71627970e-05 1.47618804e-04 2.56361207e-04 8.28940611e-05
1.76530288e-04 5.17332774e-05 1.28926558e-05 2.16640274e-05
5.26299800e-06 6.59979298e-04 1.14130096e-04 2.94375030e-04
4.49465224e-05 3.86519037e-04 4.72550419e-05 5.00234251e-04
1.12830836e-03 8.48204363e-05 5.19808455e-05 9.09078008e-05
1.57584349e-04 2.78345804e-04 2.19877635e-04 4.03005979e-05
2.55742081e-04 1.43388272e-04 4.44702513e-04 2.79434986e-04
5.58324682e-04 5.39616995e-05 3.54843542e-05 1.03632403e-04
4.92443869e-05 2.78861407e-04 1.90620158e-05 8.88124196e-05
9.66425578e-05 6.71750022e-05 2.91049073e-04 7.77899695e-05
1.15416595e-03 1.02880273e-04 6.21788786e-05 2.89910240e-04
1.45386657e-04 1.36766161e-04 8.10648140e-04 6.69935369e-04
5.99710562e-04 3.34999604e-05 7.04395570e-05 2.60458299e-04
2.57278443e-04 1.14066189e-03 4.60094998e-05 4.41401935e-04
2.84365797e-05 3.70950802e-05 1.20685632e-04 1.47119304e-03
2.59156623e-05 2.63112506e-05 1.31254434e-04 2.98050581e-05
1.61606848e-04 5.05411881e-05 6.80081712e-06 9.02227985e-05
4.81993848e-05 1.60939198e-05 1.06712716e-04 2.13666572e-04
5.58225962e-04 2.65355629e-04 3.55328120e-05 5.77366372e-05
1.57414470e-05 7.45292418e-05 2.90369026e-05 3.63006620e-05
3.77016753e-04 8.69203868e-05 5.23683120e-05 3.87982968e-04

8.73839963e-05 3.36109719e-04 1.59285133e-04 3.57647252e-04
7.11770990e-05 4.68818267e-04 1.19898650e-04 2.17655106e-04
1.07248547e-04 2.41937068e-05 6.84147426e-06 1.47503679e-05
2.38636349e-05 2.54967075e-04 4.44853285e-05 1.28478190e-04
2.98674451e-04 2.83423171e-04 2.91537610e-04 4.44025900e-05
1.02678205e-05 2.48706638e-05 2.28041517e-05 6.26483583e-04
1.77620866e-04 1.27518970e-05 8.18574772e-05 3.33283795e-04
1.51482218e-05 3.01448337e-04 1.46569782e-05 2.15305390e-05
1.31581663e-04 2.44356837e-04 8.89293879e-05 9.52232804e-04
1.69589208e-03 1.24380473e-04 1.11265399e-05 6.32024094e-05
1.05321618e-04 1.18508018e-04 1.12465736e-04 2.44203184e-05
2.22543022e-04 5.47238378e-05 4.02125734e-04 2.07621395e-03
1.29073765e-03 4.26254934e-04 3.13847762e-04 7.91640923e-05
3.74486481e-05 6.06671856e-05 1.01255473e-05 3.70270791e-05
1.40693897e-04 1.74860365e-03 1.51431232e-05 9.23906977e-04
1.58916961e-03 2.58721877e-04 1.62998028e-02 4.60242045e-05
1.30477099e-04 8.42971145e-04 6.61320693e-04 1.53327564e-04
1.76025023e-05 2.58769403e-04 1.16294796e-04 2.02700309e-03
1.35269817e-02 1.61069701e-03 6.28815324e-05 4.83175245e-05
8.54622282e-04 3.00164014e-04 1.86769132e-04 8.30927311e-05
1.89164875e-05 1.16337396e-04 3.49664151e-05 5.81785971e-05
1.28549105e-03 7.51490414e-04 1.59569143e-04 1.30329296e-04
6.99314696e-05 4.23135440e-04 1.16385119e-04 1.07296789e-02
8.77031707e-05 8.07532706e-05 6.63905928e-04 8.49344142e-05
2.24430609e-04 1.57332077e-04 8.75685691e-06 7.69934268e-04
2.06400455e-05 3.19353530e-05 5.70448901e-05 1.94806253e-05
1.05689175e-03 3.56638891e-04 8.67792005e-06 2.08657671e-04
8.68590578e-05 3.12234624e-03 7.49109677e-05 2.20658985e-04
5.02509007e-04 3.99989784e-02 8.39983113e-05 7.98574547e-05
5.47855336e-04 1.25976512e-05 5.13877967e-05 9.06003697e-05
5.23220806e-05 9.12444102e-06 2.34634455e-04 6.79804885e-04
3.31446150e-04 1.17720541e-04 9.75174538e-04 9.81520046e-04
9.12562013e-04 3.47146924e-05 1.41495068e-04 3.66432505e-04
6.45259570e-05 1.28914064e-04 7.21394317e-05 2.28986819e-03
3.76409618e-03 2.80516082e-03 1.48055784e-04 1.07101614e-05
4.61423806e-05 1.39898417e-04 1.82776726e-04 1.25178485e-04
3.83468068e-05 2.35304586e-03 3.29897623e-04 1.33856607e-03
1.06934596e-04 7.09476590e-04 8.69219512e-05 1.59723614e-03
1.43893967e-05 4.45995526e-03 6.88059226e-05 2.52713544e-05
2.10724911e-03 3.38549835e-05 2.71558602e-05 4.11760557e-04
1.15444149e-04 2.09193953e-04 1.98860871e-04 3.77124990e-04
1.20693345e-04 9.27479705e-04 3.17542319e-04 1.70437590e-04
2.65740935e-04 9.83073642e-06 1.02659171e-04 2.55792460e-04
1.47136417e-03 3.02899862e-05 1.36041519e-04 9.07273439e-04
5.02650626e-04 8.89950534e-05 1.07500923e-03 2.25143507e-04
2.58525215e-05 2.61601672e-04 1.77471444e-03 4.47884595e-05
1.29528707e-04 1.72686905e-05 1.25372258e-04 1.40629854e-04
1.54930531e-05 2.03966832e-04 1.26344059e-03 1.41774456e-03
3.88985645e-05 7.27562001e-05 2.10891685e-05 2.49873228e-05
1.81706215e-04 2.27203942e-03 8.91905569e-04 1.27628358e-04
1.71735854e-04 3.64318621e-05 7.76324887e-05 2.27155426e-04
3.80234778e-01 1.11255853e-03 4.91088031e-05 7.29268868e-05
6.97635114e-04 2.08517420e-04 4.85319458e-03 8.05228367e-04
1.44731428e-04 9.00824016e-05 5.56111154e-05 3.25820292e-05
6.10917123e-05 4.72707288e-05 3.22346226e-04 5.27381853e-05
3.36610625e-04 1.03090955e-04 2.43045681e-04 4.20626864e-04
1.52519468e-04 6.08003756e-04 5.78366977e-04 6.52293442e-04
6.34193653e-04 7.08757085e-04 9.81467194e-04 1.93246495e-04
2.93523885e-06 5.46953233e-04 7.79101756e-05 1.26446816e-04
3.07050510e-03 7.41158001e-05 2.09731492e-03 1.64307945e-03
1.22230573e-04 5.14956249e-04 2.00441311e-04 9.52015325e-05
4.60930576e-04 1.96795794e-04 5.05214091e-04 1.70887785e-03
5.94898360e-04 2.80495060e-05 6.12656368e-05 4.76151326e-04
6.06533897e-04 6.90728833e-04 2.45146477e-03 3.85669337e-05
1.50876329e-03 1.49309664e-04 8.26684013e-03 1.86099369e-05
3.37498757e-04 8.78973442e-05 2.75093160e-04 1.37260722e-04
2.65541067e-03 3.17754515e-04 1.07748630e-04 9.04394081e-04
1.07567979e-03 3.74500407e-04 5.36410778e-04 2.80095974e-05
2.61744281e-05 1.83778873e-04 1.61253134e-04 5.47529125e-05
7.77427267e-05 3.06203519e-03 1.72918866e-04 9.53771960e-05
2.25956505e-03 3.95534909e-04 6.81745078e-05 1.34670379e-04
1.79122146e-02 1.01388141e-03 6.40834635e-03 4.89571248e-04
4.02923361e-05 4.73684740e-05 1.40673510e-04 1.73157154e-04
1.48242374e-03 8.56971892e-05 1.73195396e-04 8.43464513e-05
2.08894999e-04 4.32949892e-05 6.85631094e-05 6.85166684e-04
5.59892542e-05 1.69968873e-04 8.19040215e-05 5.71506680e-04
3.53219875e-05 1.04701510e-04 2.14931388e-05 3.53869582e-05
3.78410798e-04 5.04389638e-04 3.27573434e-05 2.83361995e-04
2.72948737e-03 9.21294268e-05 1.48180025e-05 2.46270763e-04
4.13689813e-05 4.71792555e-05 1.51615334e-03 3.52851202e-05
9.80600948e-04 2.52885919e-04 1.63716366e-04 3.37931451e-05
1.07107393e-04 6.50193251e-04 2.43731687e-04 6.05064961e-05
1.35610229e-04 6.65127416e-04 3.26190260e-04 7.57016824e-05
3.46393699e-06 6.07279690e-05 3.57153876e-05 7.71200066e-05
1.00423140e-03 1.00033205e-04 5.33735838e-05 1.09746339e-04
2.88255571e-04 9.57804441e-05 3.41402010e-05 2.11888662e-04
1.22098456e-04 6.21896761e-05 9.76745723e-05 8.55488179e-05
9.62076883e-04 4.20940109e-04 5.54108410e-05 1.32340530e-04
4.31249244e-03 5.03420888e-05 8.13174192e-05 4.18817438e-03
7.12255451e-06 9.48144123e-04 2.42499755e-05 7.69914873e-03
5.26841131e-06 2.83917616e-04 4.28069943e-05 1.27932144e-05

2.66362971e-04 1.69233754e-04 4.73914290e-04 1.19051627e-04
1.91241088e-05 5.86830685e-03 7.07294850e-04 2.50239773e-05
3.75086616e-04 1.55779562e-04 1.32933401e-05 4.63665128e-05
4.63213632e-03 1.30101602e-04 6.92261747e-05 1.00446850e-04
1.05526822e-03 1.89026483e-04 9.04939006e-05 1.03394501e-04
9.69200992e-05 5.64555405e-04 2.52025693e-05 3.57972931e-05
4.88542646e-05 1.33715745e-04 1.04600520e-04 9.90291475e-04
1.22292840e-04 7.27627921e-05 2.14016090e-05 3.12159164e-03
2.09966645e-04 1.85880926e-04 1.90800958e-04 5.63547364e-04
2.16300876e-04 2.20534217e-04 6.04555644e-05 6.00815147e-05
2.14696829e-05 1.93196393e-04 6.29920323e-05 1.38893153e-03
1.97276357e-03 3.87275737e-04 4.70409956e-04 1.04977466e-04
8.66038899e-05 2.10130784e-05 2.21918861e-04 1.92755160e-05
9.36520300e-05 3.28548958e-05 1.16980786e-03 4.76077330e-05
5.36649197e-04 1.12778936e-04 3.09132611e-06 1.16967904e-05
3.23557877e-04 2.54508495e-06 4.74176253e-04 6.56044402e-04
8.27612123e-04 6.55535914e-05 1.99410575e-03 1.48264132e-03
8.31854195e-05 9.95933297e-05 4.19858785e-04 4.76322530e-05
4.94358654e-04 2.63915467e-03 7.25556747e-04 7.74318527e-04
7.02826772e-04 3.97677359e-05 9.80811380e-03 2.64382761e-05
4.16843759e-05 7.06135994e-04 2.61750283e-05 8.92328855e-04
1.27251481e-03 1.58596595e-04 2.71753408e-04 6.71056274e-04
1.84216391e-04 3.36448866e-04 4.11851470e-05 1.33671318e-04
1.52624969e-04 2.06730019e-05 3.06273869e-04 2.71964981e-03
4.70857427e-04 3.60155609e-05 7.55196816e-05 1.57036399e-03
1.54864065e-05 2.31870697e-04 1.19008953e-02 3.09153111e-04
1.25879757e-04 1.03011444e-04 1.46702502e-03 7.43132041e-05
2.84773414e-05 5.63574489e-04 1.61093209e-04 8.43522896e-04
9.58754797e-04 5.08159465e-05 2.06946715e-04 6.66140520e-04
5.87129070e-05 2.06479308e-06 2.25120777e-04 1.80725110e-05
1.10015128e-04 4.83739830e-04 1.34500238e-04 4.28811356e-04
5.18853834e-04 1.01364413e-02 3.64524050e-04 5.43028364e-05
2.39440371e-04 2.05363482e-04 1.51743268e-04 9.51730253e-05
1.73279313e-05 5.83487927e-05 1.47452345e-04 3.08834296e-03
1.24863713e-04 1.79667713e-05 3.07209697e-03 2.79738251e-05
7.99115587e-05 2.34446015e-05 9.82822618e-04 2.34153867e-03
1.13934024e-04 2.91827037e-05 1.47375016e-04 5.59431974e-05
1.17024174e-04 1.57576782e-04 3.46450892e-04 7.32991204e-04
1.69161067e-04 1.59455743e-03 1.73376029e-05 8.59025677e-05
1.01042865e-03 1.09213754e-04 1.79306717e-05 5.39152825e-04
1.89648981e-05 7.20300188e-04 1.79962174e-03 3.27884639e-03
7.25260616e-05 3.15109646e-04 7.32668326e-04 1.53096189e-05
4.52849163e-05 3.47554538e-04 2.79870495e-04 6.85701016e-05
1.07863576e-04 1.66341953e-04 1.45367114e-04 1.51955945e-04
1.18492193e-04 7.17391435e-04 2.06604693e-03 2.06299519e-04
2.54458486e-04 6.54346426e-04 2.97473860e-04 7.34544301e-04
6.25758403e-05 2.90876233e-05 1.74098328e-04 1.11928071e-04
1.14851497e-01 2.62955250e-03 2.40852195e-03 1.72162443e-04
8.75283731e-05 6.66758686e-04 1.76057420e-05 1.84296805e-04
1.70327738e-04 2.79838045e-04 4.37207578e-04 8.90741474e-04
2.95937969e-03 3.18998966e-04 7.71548730e-05 1.85021563e-04
1.17419753e-03 2.78085681e-05 9.33784249e-05 6.90118832e-05
4.32995730e-05 2.92342224e-06 4.15394534e-05 1.52136008e-05
1.29134523e-05 2.94119789e-04 9.48211455e-05 6.97282157e-05
6.17292535e-05 4.16714574e-05 5.00194401e-05 1.13236427e-04
5.65801565e-06 1.36517001e-05 4.56229645e-06 5.45705625e-05
1.98253110e-05 1.48473191e-04 6.39737045e-05 2.67284995e-05
4.29977372e-05 1.51909131e-04 1.80505929e-04 3.08210932e-04
2.26825607e-04 7.97288594e-05 9.52363989e-05 4.29513457e-05
5.10523641e-05 2.09242899e-05 6.18914710e-05 5.63330032e-05
1.90053997e-05 6.33332165e-05 5.60914486e-05 3.97748627e-05
1.83574355e-06 1.78109458e-05 5.12331644e-05 2.07487301e-05
3.00131378e-05 2.81950579e-05 4.41239914e-03 1.48228282e-04
8.43753296e-05 5.18359993e-05 9.11546085e-05 5.44237951e-03
1.05446445e-04 7.14217837e-04 6.95484108e-04 1.59877454e-04
1.04986575e-04 1.55470407e-04 5.15914580e-04 2.69758631e-04
5.53757884e-04 9.67251690e-05 1.41127763e-04 5.17273264e-04
1.65178091e-04 4.38820600e-04 7.76213055e-05 7.71105988e-04
8.71600205e-05 6.32327283e-04 2.78978841e-04 3.60339385e-04
1.25893086e-04 1.04214778e-04 5.29835888e-05 8.40803841e-05
3.67418434e-05 7.11586326e-05 4.39933137e-05 1.84522447e-04]]

```
-----  
AttributeError                                Traceback (most recent call last)  
<ipython-input-17-83421ec204f5> in <module>  
      7  
      8 print(model.predict(test_image))  
---->  9 print(model.predict_classes(test_image))  
     10 print(y_test[0:1])  
  
AttributeError: 'Functional' object has no attribute 'predict_classes'
```

Testing a new image

```
In [18]: test_image_path = 'D:/Harold/MyDNN/DataSet/Chest_xray_seperate/PNEUMONIA/person11_bacteria_45.jpeg'
test_image = image.load_img(test_image_path, target_size=(224, 224))
x = image.img_to_array(test_image)
x = np.expand_dims(x, axis=0)
x = preprocess_input(x)
print (x.shape)
```

```
# if num_channel==1:
#     if (K.image_data_format() == 'channels_first'):
#         test_image= np.expand_dims(test_image, axis=0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=3)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
# else:
#     if (K.image_data_format() == 'channels_first'):
#         test_image=np.rollaxis(test_image,2,0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)

# Predicting the test image
yhat = custom_resnet_model.predict(x)
print(yhat)
# print(custom_resnet_model.predict_classes(x))
label = decode_predictions(yhat)
# retrieve the most likely result, e.g. highest probability
label = label[0][0]
```

```
(1, 224, 224, 3)
[[0.04744451 0.95255554]]
```

```
-----
ValueError                                Traceback (most recent call last)
<ipython-input-18-505048f79341> in <module>
    30 print(yhat)
    31 # print(custom_resnet_model.predict_classes(x))
--> 32 label = decode_predictions(yhat)
    33 # retrieve the most likely result, e.g. highest probability
    34 label = label[0][0]
```

```
D:\Anaconda3\lib\site-packages\tensorflow\python\keras\applications\inception_v3.py in decode_predictions(preds, top)
    412 @keras_export('keras.applications.inception_v3.decode_predictions')
    413 def decode_predictions(preds, top=5):
--> 414     return imagenet_utils.decode_predictions(preds, top=top)
    415
    416
```

```
D:\Anaconda3\lib\site-packages\tensorflow\python\keras\applications\imagenet_utils.py in decode_predictions(preds, top)
    149         'a batch of predictions '
    150         '(i.e. a 2D array of shape (samples, 1000)). '
--> 151         'Found array with shape: ' + str(preds.shape))
    152     if CLASS_INDEX is None:
    153         fpath = data_utils.get_file(
```

```
ValueError: `decode_predictions` expects a batch of predictions (i.e. a 2D array of shape (samples, 1000)). Found array with shape:
(1, 2)
```

```
In [17]: score = custom_resnet_model.evaluate(X_test, y_test, verbose=0)
print('Test Loss:', score[0])
print('Test accuracy:', score[1])

test_image = X_test[0:1]
print (test_image.shape)

print(model.predict(test_image))
print(model.predict_classes(test_image))
print(y_test[0:1])
```

Test Loss: 0.13225294649600983
Test accuracy: 0.9744027256965637

(1, 224, 224, 3)

[[4.82067890e-07 2.68882036e-06 2.46438071e-06 1.46347942e-04
1.14383847e-05 4.81724349e-08 2.50176436e-06 1.93318300e-07
1.12078631e-07 1.33230387e-07 1.55274762e-07 4.57866662e-07
8.66559446e-07 2.24399500e-07 6.64423965e-08 1.10845995e-07
1.35442542e-06 5.03833007e-07 1.79790817e-07 6.47744287e-07
3.05023384e-08 9.56223190e-08 1.10445868e-07 1.10137853e-06
7.54676677e-08 4.63201928e-07 2.87497016e-07 7.07335062e-08
5.35643494e-07 1.51549584e-05 2.59477027e-08 2.91564879e-06
7.73133081e-07 4.95935637e-07 1.98538643e-07 2.52716568e-08
1.83687155e-06 7.88459204e-07 3.27636599e-07 1.85687998e-06
1.74747020e-05 5.42289399e-07 8.17253891e-08 6.31402060e-08
3.37520163e-07 1.02205229e-06 7.63697062e-06 5.59314408e-08
7.33921922e-07 4.65218932e-07 4.84302063e-06 6.55129497e-08
2.73576143e-06 3.79595463e-06 7.06228604e-07 5.70710426e-06
9.65091203e-07 2.63103260e-07 4.50051573e-07 6.29083502e-07
2.09377595e-06 1.72502826e-06 1.25136467e-06 6.38593519e-07
7.39525774e-07 3.34286518e-08 9.70000769e-07 1.03747618e-06
3.67367988e-07 1.14979400e-06 3.29745234e-07 2.08577912e-05
1.47276842e-05 4.14989336e-06 7.56582722e-06 6.36893992e-06
8.98751387e-06 9.69357734e-06 2.66852585e-05 2.10938379e-05
9.31147923e-08 4.12824157e-08 4.71741544e-08 1.20277676e-07
2.82732515e-07 7.39384305e-08 4.42064376e-08 4.16540644e-07
7.95168603e-07 5.63529099e-08 8.61598224e-08 3.73003417e-07
3.13063069e-08 1.43328606e-07 2.56055932e-06 1.59463571e-08
2.61807202e-07 2.08426371e-07 2.79966450e-09 2.37027564e-07
8.26529174e-07 9.52778294e-07 8.84871895e-07 6.70584797e-08
4.19899031e-08 3.42380346e-08 7.15500903e-08 1.04467529e-06
1.84370560e-06 1.10508445e-07 1.32549957e-07 1.20732386e-06
4.30385662e-05 2.32143202e-06 1.15385831e-06 2.89288717e-08
7.44268647e-08 1.84088658e-05 1.56171754e-05 3.71134149e-07
2.79030104e-07 1.25042470e-06 1.56302053e-06 1.15310036e-06
2.40664394e-06 1.65841428e-07 4.56227317e-05 2.70883021e-07
7.59855823e-09 1.88321238e-07 1.26450095e-06 6.97321809e-08
8.32176397e-07 2.10207972e-07 2.25861598e-07 9.37061273e-08
1.62370782e-07 8.13763492e-08 4.42455530e-07 9.24499304e-08
5.98086899e-06 3.80057621e-08 3.88246093e-08 8.12120277e-07
1.10410170e-07 3.98443092e-07 2.99336165e-07 5.37295023e-07
1.38613882e-06 2.95345467e-06 7.86266270e-08 2.17395083e-07
7.75959563e-08 1.36170161e-06 7.63363289e-07 3.60099807e-06
1.50860660e-06 4.62941415e-07 4.47918381e-07 1.67911749e-05
5.81790516e-07 6.88692182e-07 1.80056261e-07 5.41830104e-06
1.66806376e-06 1.10999708e-05 2.09217092e-06 5.40804081e-07
2.85339570e-06 7.10956510e-06 7.90489878e-07 5.44291697e-06
2.60349816e-06 4.81833240e-06 5.70534951e-07 5.57362114e-07
1.02933473e-06 1.04458866e-06 4.40884469e-05 1.90422298e-07
1.48606148e-07 4.46824458e-07 1.89701916e-06 3.72328145e-07
2.70331907e-06 1.42775207e-06 2.15032173e-06 1.50941526e-06
4.57386960e-07 2.96538616e-07 9.06139022e-08 1.39788681e-05
1.63965376e-06 2.08460364e-07 8.88446650e-08 1.14750846e-05
6.00025544e-07 1.29099749e-06 1.63062253e-07 1.93221626e-06
3.26226086e-07 2.26011565e-07 1.00890395e-06 3.64552739e-06
2.38593344e-07 3.57042865e-07 1.42592728e-06 1.51925451e-06
1.76033518e-06 1.65332074e-06 1.73654357e-07 6.30601699e-06
1.29027364e-06 2.55493205e-06 1.25809265e-07 6.50791947e-07
2.88133009e-07 3.74395086e-07 1.28145970e-07 7.58302292e-07
2.54769958e-07 5.29478257e-07 9.42013060e-07 1.10448559e-06
1.35511527e-06 2.74070345e-07 4.58742932e-07 1.18524120e-07
1.96650092e-07 8.79864217e-07 6.76232673e-07 1.05864603e-06
7.84085387e-07 1.61685605e-07 1.75777370e-06 3.27096785e-07
2.50208632e-06 2.27774103e-06 1.32969490e-06 2.00462722e-07
1.95692806e-07 3.73978310e-07 1.26169480e-05 9.73640454e-07
7.51481721e-09 3.16245831e-07 1.91538516e-06 1.11579197e-07
5.43942451e-06 3.55741150e-07 2.83933048e-07 5.07502136e-06
2.17246588e-06 2.19098570e-06 3.57639419e-06 2.53928022e-07
2.06906194e-07 3.27570973e-07 1.17160323e-06 3.96143605e-07
7.07220309e-08 2.98140570e-08 1.48195642e-08 5.41426914e-07
3.64813886e-08 8.76403021e-07 1.02535796e-06 6.06517460e-07
3.88640473e-07 7.15487602e-07 6.84218890e-07 4.20073462e-07
2.81511035e-07 4.72189640e-07 4.67730388e-08 1.60121218e-07
9.27563732e-08 2.03835686e-08 8.10001666e-09 6.71593412e-08
3.04826777e-08 2.13095677e-06 5.48989783e-06 3.82242615e-06
4.02625938e-06 2.03273908e-06 5.60944713e-08 3.95658226e-08
1.71023977e-08 2.75019083e-08 3.71704267e-08 3.75872098e-08
3.24326976e-07 1.62003513e-08 2.47091805e-08 5.32830882e-08
5.27075144e-07 6.81948364e-08 2.12066894e-07 4.73816954e-06
3.06480871e-07 4.37967174e-06 3.83125638e-07 2.59487564e-07
2.16806015e-07 1.23415191e-07 7.58255283e-07 5.34728883e-07
7.38245376e-07 1.07934525e-07 8.58397584e-07 6.98828399e-06
4.37092168e-06 1.10831215e-05 1.25559709e-05 2.39583405e-05
1.33940068e-06 1.43173759e-06 4.31827516e-07 5.54513917e-07
1.33714991e-07 2.53679673e-07 5.93662328e-08 6.51950813e-08
7.06708306e-07 8.72265176e-08 1.03443178e-07 4.26473889e-05
2.43622111e-07 4.32115826e-07 7.37000335e-07 5.41148424e-08
2.39720293e-07 1.38767909e-05 2.39044539e-07 1.00757500e-08
5.52106911e-08 2.85622832e-07 1.39397207e-06 8.98978954e-07
2.47298317e-06 2.48008291e-06 1.03912527e-07 9.20034751e-08
1.58720539e-07 2.18869877e-06 1.01007416e-07 9.33262001e-08
4.16232069e-07 2.41899780e-08 7.14705735e-08 8.87272677e-07

1.79500682e-06 2.23839402e-06 3.55981769e-07 2.05079019e-07
7.03610851e-07 5.05889552e-08 1.70242330e-07 3.62089213e-06
1.10505658e-08 1.03553134e-07 6.95341598e-08 7.76962622e-07
1.41140958e-07 2.01013586e-06 1.21628759e-07 1.97350872e-07
8.37480698e-08 1.60278351e-08 2.88566788e-08 1.44770496e-07
3.86343402e-07 2.70098553e-07 5.88721143e-08 1.20218971e-07
1.64466272e-07 1.54137524e-07 2.27946529e-07 6.77903458e-08
2.53453805e-07 5.70535192e-08 7.12599970e-08 7.95363135e-08
8.29713365e-09 8.91881768e-07 1.03283935e-06 2.96836102e-08
6.41702442e-08 8.35964215e-07 4.04590082e-06 5.43997976e-06
3.47285507e-08 7.03444130e-06 1.76835019e-05 4.78230675e-07
2.89848629e-07 6.81267096e-08 1.10332494e-05 7.61832460e-04
9.27375368e-06 2.28788940e-05 9.84669805e-06 4.32519073e-07
9.30536407e-06 8.10772235e-07 1.49337247e-05 1.51340806e-04
4.47854944e-08 5.67501527e-04 1.10032076e-07 2.25565200e-05
6.77242133e-05 6.33004038e-06 4.39342193e-06 8.99438674e-05
7.28008381e-05 2.19390568e-05 1.23546979e-05 1.25297393e-05
4.20370634e-06 3.13102093e-04 3.20764656e-07 5.62300329e-06
1.37011142e-04 1.94485983e-06 4.81351935e-06 9.91487832e-05
1.63843524e-05 8.57108080e-06 1.35260478e-07 1.85790210e-04
4.36487335e-06 2.35706648e-05 3.92094895e-04 1.27935903e-02
2.70703390e-06 1.01705751e-04 4.49157596e-05 6.13191901e-07
1.15572448e-05 1.44345540e-04 2.29499601e-05 1.85781857e-06
2.22390395e-06 9.81719568e-05 9.97880561e-05 7.9112107e-05
3.96518726e-06 3.96266415e-07 8.61680601e-04 6.50838228e-06
2.06842174e-06 7.68140399e-06 1.07899541e-05 2.91467586e-06
2.84931957e-06 1.47860090e-04 4.30169575e-06 1.07488831e-05
4.59787589e-05 5.46703232e-05 7.03009573e-06 1.99430560e-05
3.42860039e-06 4.22399671e-06 6.29881397e-05 5.03593292e-05
1.70047624e-05 2.34267577e-06 6.42493023e-06 1.19209335e-06
2.15346809e-04 3.78162440e-06 8.40231860e-06 5.85121888e-05
1.26830651e-06 3.94399819e-07 9.75983203e-05 3.36703924e-05
1.23899110e-04 3.43893862e-06 2.21809387e-06 2.35215539e-05
1.90323753e-05 1.14208233e-05 3.64518337e-06 2.74725782e-04
1.66802511e-05 4.90731800e-05 2.14859301e-05 8.27478743e-07
6.38046004e-06 6.69588553e-06 4.86439931e-05 2.94567570e-07
1.45346610e-06 2.36542764e-05 5.61834313e-05 5.70035423e-04
7.14134103e-06 5.04184165e-04 9.28850553e-04 2.66718762e-05
1.97937461e-05 2.34957828e-04 1.12313012e-06 1.54577606e-06
1.91156039e-04 2.92513650e-05 3.64735342e-06 8.05021045e-06
5.52102756e-06 2.36397736e-05 1.86461883e-04 8.69344076e-05
7.38855801e-04 1.13097872e-06 3.45201647e-06 2.67267733e-05
5.17669905e-05 3.94152949e-06 9.94110039e-08 1.02107367e-03
3.70649213e-05 7.88004545e-05 2.20635525e-06 1.11422360e-05
1.52004883e-04 2.07467747e-05 2.19970061e-05 1.89681759e-05
2.91477704e-07 2.03410423e-06 1.43415236e-04 4.07606819e-07
3.71343731e-07 1.21202063e-07 1.54469137e-06 5.18416382e-05
2.46199988e-07 1.38334117e-05 2.72481393e-05 1.73800756e-04
1.29404398e-07 1.32949333e-06 5.21159427e-06 4.13591010e-08
7.22797097e-07 5.77609389e-05 2.60100387e-06 2.77295186e-07
8.79857544e-05 2.48557399e-06 3.33061253e-05 1.84840428e-06
2.06588447e-06 9.42903935e-05 1.27391613e-05 3.86221131e-04
2.89812959e-07 9.07293781e-07 9.14044576e-05 1.18784788e-04
1.44671003e-05 3.06865513e-05 2.65360472e-06 2.47145545e-06
7.08662847e-05 4.67554706e-08 7.63243952e-05 4.61026211e-05
4.03199054e-04 2.75975566e-07 6.68384700e-06 3.74048363e-06
2.48065794e-06 7.88056695e-06 3.06886015e-03 1.04375024e-04
4.88235783e-05 3.2222841e-05 2.76110386e-05 3.63568074e-06
1.40114649e-08 3.91990943e-05 3.56440665e-07 2.30869646e-05
7.97552639e-05 1.15201226e-03 1.25383340e-05 3.24318215e-04
2.47357548e-06 6.83440303e-05 6.86029762e-06 4.83607437e-06
1.21831170e-06 4.36718392e-06 5.05537037e-06 3.59021215e-05
1.23418329e-04 4.54013381e-04 1.95090820e-06 1.79916094e-06
3.72640061e-04 2.53085134e-04 8.23253049e-06 1.11542977e-06
1.29810622e-04 6.05422611e-06 5.47393465e-05 3.38295195e-07
1.88352317e-07 1.01384667e-05 7.33205843e-06 4.75033376e-05
9.63159891e-06 5.74934878e-04 1.54930181e-04 4.78529284e-04
5.10371116e-04 1.49897926e-06 6.17007436e-06 1.60871841e-05
4.19408798e-06 2.20982770e-06 2.88490191e-05 2.74454069e-05
8.71936663e-06 3.60013473e-05 2.32321071e-03 2.51333404e-04
2.51822348e-04 4.09818222e-06 1.71445140e-06 9.45266763e-08
5.68463793e-06 1.36511799e-05 2.69595825e-04 1.61458447e-03
2.42772586e-07 1.55881673e-04 6.31328949e-05 4.16917392e-05
5.45173316e-06 7.44753208e-07 7.79232232e-06 5.39383000e-05
4.50248399e-06 3.55712473e-06 1.38671661e-04 2.44844637e-06
4.92295039e-05 1.22241481e-05 3.03424658e-05 1.46714054e-04
1.15310540e-05 3.78642610e-04 1.30925957e-06 3.95897177e-06
1.03032858e-06 2.12843702e-07 1.84312103e-05 1.35504897e-05
3.08177623e-05 9.31254078e-07 2.19377671e-06 1.40750149e-06
2.77103413e-06 1.50421634e-03 5.47954687e-06 7.84884105e-06
1.34442689e-05 1.27914784e-04 2.03610898e-06 2.55877762e-06
7.29639680e-07 4.32577008e-06 8.44035894e-05 1.19183301e-06
1.02465716e-03 6.94555201e-05 1.54967870e-06 5.26904332e-05
2.13380144e-06 3.60782650e-07 6.98907070e-06 5.14113344e-04
1.20219784e-05 3.05959074e-05 1.08575435e-07 1.02883430e-04
8.84334731e-04 3.34499382e-05 3.18635784e-06 7.57592898e-06
6.32017545e-05 6.54682008e-05 3.43472857e-05 1.74541310e-05
1.87886071e-05 8.99571205e-06 8.88989052e-06 3.25882465e-05
4.18749369e-06 3.24225698e-06 3.95771849e-06 5.82839266e-06
2.83199537e-04 1.09739551e-04 1.02437571e-06 2.33481678e-05
6.11256382e-07 3.62084575e-05 2.69073917e-06 6.38804295e-07

1.68975330e-05 7.29936858e-07 4.69654879e-05 9.58434839e-07
1.16623396e-05 3.18129332e-06 2.63417624e-05 1.61167279e-07
2.44845182e-07 1.47840212e-04 3.11488293e-06 1.70200997e-06
1.69117455e-04 1.04137844e-05 9.06369337e-07 7.02995094e-05
6.30474779e-06 3.74916417e-05 8.15141789e-07 1.95482149e-04
1.63665663e-05 9.46354266e-05 1.25333008e-05 1.66010122e-05
1.99244059e-05 2.11239931e-05 1.94751701e-05 2.05911729e-05
3.29209695e-04 3.14238532e-05 3.62695204e-07 3.52477255e-05
9.37491313e-06 1.92738080e-05 4.31706831e-05 2.57115266e-06
1.95169850e-05 8.58446956e-03 1.87854255e-06 1.59508318e-06
2.46171567e-05 2.86161753e-06 1.01712494e-05 3.48290519e-06
2.47714834e-05 2.18087720e-04 2.13829549e-06 5.23960916e-05
1.30321478e-05 4.40097683e-05 3.76050639e-06 3.32131895e-05
7.28104169e-07 9.54941061e-06 1.03316641e-04 2.34550794e-06
2.92102450e-05 4.44840378e-04 2.78215832e-03 3.26271955e-04
8.13547449e-05 9.37511345e-07 1.28339278e-04 3.95520556e-06
5.72408362e-06 4.45332120e-07 6.10011957e-05 9.87950716e-06
6.84851830e-06 6.87362626e-04 2.87053790e-05 1.62784681e-06
1.16443750e-03 1.19960787e-04 6.68877601e-06 7.05155526e-06
7.51127482e-06 4.41425545e-05 8.68385494e-01 4.11766086e-06
7.50159688e-06 4.51385858e-04 1.72617217e-06 3.76659416e-04
1.03966563e-07 6.19818229e-06 3.53539377e-07 2.29919351e-05
1.92507523e-05 9.72411362e-06 3.90483037e-04 3.46978004e-06
3.11856165e-05 2.66272559e-06 1.59578831e-05 2.38444496e-04
1.12737598e-05 9.12053565e-06 5.20690662e-07 1.06905558e-04
8.12665277e-08 8.96848462e-07 4.85853207e-06 1.47953815e-05
1.02085451e-07 1.24309815e-06 7.21049048e-07 1.65302597e-04
1.28473184e-05 1.82003646e-07 1.76036301e-05 3.73523676e-06
1.19854949e-05 6.30159866e-06 6.03208225e-03 6.73122471e-04
3.23600170e-06 6.13015072e-06 6.28529466e-04 4.05782004e-07
3.67343338e-04 2.02310417e-04 1.44595761e-04 2.47344688e-05
1.92117986e-05 2.67578725e-04 5.09061269e-04 1.11675604e-06
3.60307458e-05 5.94534067e-05 6.04638262e-05 9.43270777e-07
1.87736043e-06 1.16022029e-05 9.44192434e-05 2.90690459e-05
1.21500841e-06 1.10533176e-06 5.06489945e-04 3.64165658e-06
4.67211663e-07 5.10106838e-06 5.83519519e-04 3.80706479e-05
1.76308986e-05 5.20052490e-05 4.91349647e-06 8.12508097e-06
3.82704748e-08 2.07618473e-06 1.39368840e-05 5.41324471e-06
8.11370192e-05 8.19496287e-04 8.66509424e-08 5.77576611e-06
2.67183350e-05 3.54495683e-06 1.76372367e-07 7.78879257e-06
6.08678255e-03 1.53235869e-05 4.25836661e-05 1.81716296e-05
3.10815585e-06 1.14924820e-04 1.62769153e-04 5.68528594e-05
7.11927249e-04 4.77184221e-05 1.43913121e-06 3.70840717e-04
5.84533154e-06 1.34695501e-05 1.47781648e-07 4.95524375e-07
3.60853737e-04 3.74378851e-06 2.02726806e-03 8.60695764e-06
9.59523313e-04 1.05018029e-03 2.76033854e-04 1.06034509e-03
1.02264516e-04 5.07008451e-07 1.24075723e-05 4.30560576e-05
4.01811376e-05 3.60652879e-02 9.52280534e-05 1.05566389e-04
6.14607241e-04 4.45982351e-07 9.85839288e-05 1.09745661e-05
4.13857578e-07 1.45892869e-07 9.47745502e-06 1.27931187e-07
1.88063609e-06 7.31243233e-07 1.09413508e-04 5.59842147e-06
1.08451268e-05 2.47595250e-04 3.22890701e-05 5.87084855e-07
5.33968034e-07 1.60047946e-07 4.48863204e-08 6.79547218e-09
4.62726888e-07 4.34527137e-05 4.75226880e-05 2.52664768e-06
9.74151772e-06 3.40019170e-07 5.78103791e-05 4.44442804e-07
6.88957289e-06 8.11696373e-06 1.18416517e-06 7.13754634e-05
9.49173284e-07 2.48400141e-07 2.54554798e-05 9.35261924e-05
1.13045726e-05 2.96297640e-06 2.50614960e-07 6.59583566e-06
2.65514132e-06 5.55613678e-06 1.38014566e-05 4.61311402e-06
1.00772695e-06 3.14598365e-05 2.84581678e-03 1.14023233e-05
1.62482070e-06 4.54612916e-07 3.88075478e-06 6.19703883e-07
1.11640183e-07 1.78760547e-05 9.90545141e-06 7.91555806e-07
1.24757719e-07 3.79951121e-06 9.05809065e-06 4.07820153e-06
1.76368158e-05 6.19397781e-07 4.93730397e-07 3.93128685e-05
5.72621911e-06 2.75433791e-08 1.81638745e-06 3.58180927e-07
2.10690857e-07 2.07777812e-05 1.16643912e-06 3.86570804e-07
1.94599423e-07 4.16157627e-06 4.51370783e-04 7.39174948e-06
1.35012567e-06 7.68352857e-06 1.72485688e-08 1.03070211e-04
8.34732958e-08 9.23699091e-08 1.82024227e-08 2.52095500e-08
1.99341756e-07 4.80747442e-09 1.07438849e-08 4.42679458e-08
1.15368221e-06 2.79566930e-08 3.99251403e-05 2.31175116e-04]]

```
-----  
AttributeError                                Traceback (most recent call last)  
<ipython-input-17-83421ec204f5> in <module>  
      7  
      8 print(model.predict(test_image))  
----> 9 print(model.predict_classes(test_image))  
     10 print(y_test[0:1])
```

AttributeError: 'Functional' object has no attribute 'predict_classes'

```
In [21]: Y_pred = custom_resnet_model.predict(X_test)
print(Y_pred)
y_pred = np.argmax(Y_pred, axis=1)
print(y_pred)
#y_pred = model.predict_classes(X_test)
#print(y_pred)
target_names = ['class 0(Normal)', 'class 1(Pneumonia)']
print(classification_report(np.argmax(y_test,axis=1), y_pred,target_names=target_names))
print(confusion_matrix(np.argmax(y_test,axis=1), y_pred))

[[9.9328565e-07 9.9999905e-01]
 [1.0000000e+00 6.4503549e-16]
 [1.3406505e-04 9.9986589e-01]
 ...
 [1.9769919e-10 1.0000000e+00]
 [7.7234494e-09 1.0000000e+00]
 [9.9999988e-01 9.0127173e-08]]
[1 0 1 ... 1 1 0]

              precision    recall  f1-score   support

   class 0(Normal)         0.94         0.96         0.95         312
   class 1(Pneumonia)       0.99         0.98         0.98         860

    accuracy              0.97              1172
   macro avg              0.96              1172
   weighted avg           0.97              1172

[[301  11]
 [ 19 841]]
```

Compute confusion matrix

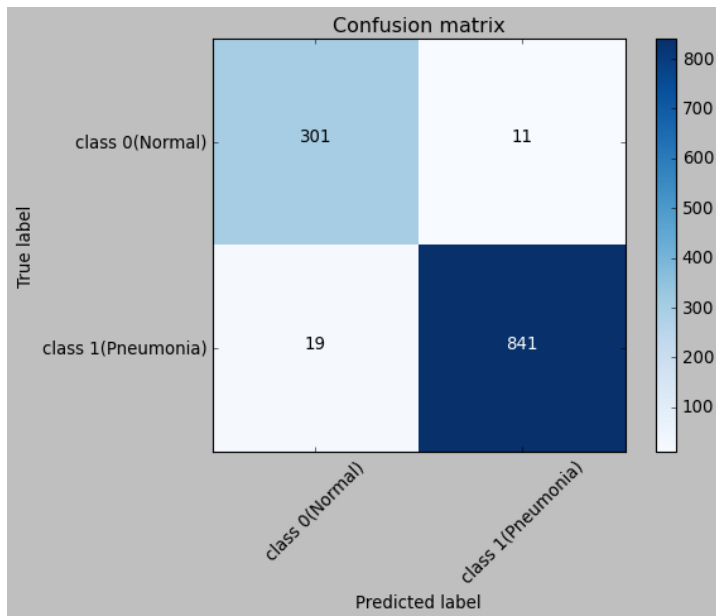
```
In [22]: # Compute confusion matrix
cnf_matrix = (confusion_matrix(np.argmax(y_test,axis=1), y_pred))

np.set_printoptions(precision=2)

# Plot non-normalized confusion matrix
plot_confusion_matrix(cnf_matrix, classes=target_names,
                      title='Confusion matrix')
```

Confusion matrix, without normalization

```
[[301  11]
 [ 19 841]]
```



regulization

```

In [19]: # def get_featuremaps(model, Layer_idx, X_batch):
#         print(model.layers[0].input)
#         print(model.layers[layer_idx].output)
#         print(test_image.shape)
#         get_activations = K.function([model.layers[0].input, K.learning_phase()], [model.layers[layer_idx].output,])
#         activations = get_activations([X_batch, 0])
#         activations = 0
#         return activations

# layer_num=3
# filter_num=0
# # print(X_train[0])

# activations = get_featuremaps(model, int(layer_num), test_image)

# # print (np.shape(activations))
# # feature_maps = activations[0][0]
# # print (np.shape(feature_maps))

# feature_maps = model.predict(test_image_nd)
# print(feature_maps.shape)
# square = 4
# index = 1
# for _ in range(square):
#     for _ in range(square):
#         ax = plt.subplot(square, square, index)
#         ax.set_xticks([])
#         ax.set_yticks([])
#         plt.imshow(feature_maps[0, index-1], cmap='viridis')
#         index += 1
# plt.show()

# model.summary()

# successive_feature_maps = model.predict(test_image_nd)

# layer_names = [layer.name for layer in model.layers]
# for layer_name, feature_map in zip(layer_names, successive_feature_maps):
#     print(feature_map.shape)
#     if len(feature_map.shape) == 4:

#         # Plot Feature maps for the conv / maxpool layers, not the fully-connected layers

#         n_features = feature_map.shape[-1] # number of features in the feature map
#         size = feature_map.shape[1] # feature map shape (1, size, size, n_features)

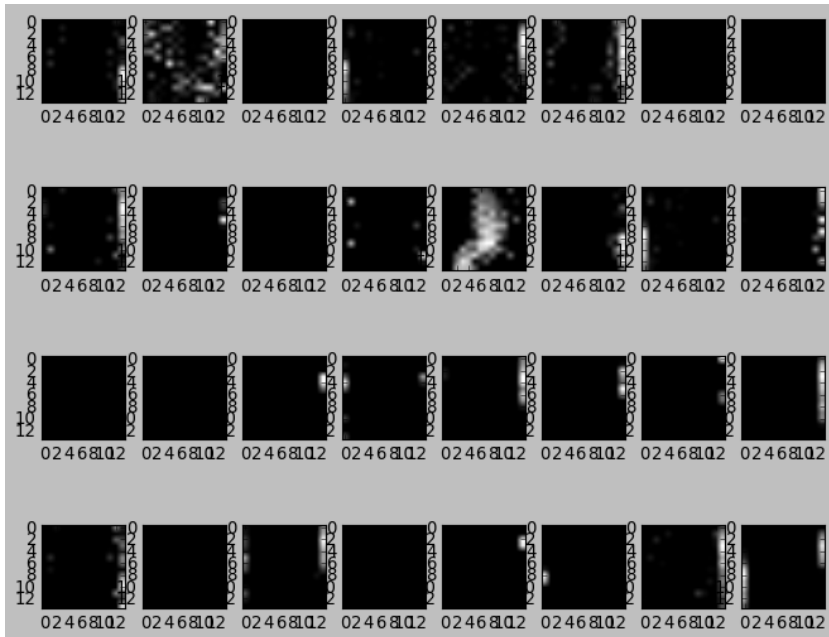
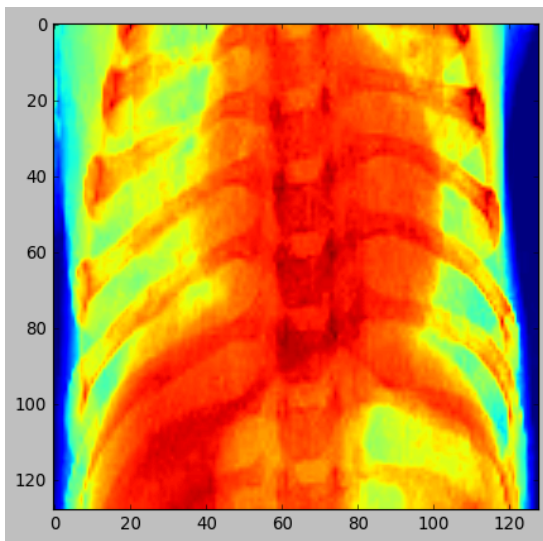
#         # We will tile our images in this matrix
#         display_grid = np.zeros((size, size * n_features))

#         # Postprocess the feature to be visually palatable
#         for i in range(n_features):
#             x = feature_map[0, :, :, i]
#             x -= x.mean()
#             x /= x.std()
#             x *= 64
#             x += 128
#             x = np.clip(x, 0, 255).astype('uint8')
#             # Tile each filter into a horizontal grid
#             display_grid[:, i * size : (i + 1) * size] = x # Display the grid
#         scale = 20. / n_features
#         plt.figure( figsize=(scale * n_features, scale) )
#         plt.title ( layer_name )
#         plt.grid ( False )
#         plt.imshow( display_grid, aspect='auto', cmap='viridis' )

from keras.models import Model
layer_outputs = [layer.output for layer in model.layers]
activation_model = Model(inputs=model.input, outputs=layer_outputs)
activations = activation_model.predict(X_train[10].reshape(1,128,128,3))

def display_activation(activations, col_size, row_size, act_index):
    activation = activations[act_index]
    activation_index=0
    fig, ax = plt.subplots(row_size, col_size, figsize=(row_size*2.5,col_size*1))
    for row in range(0,row_size):
        for col in range(0,col_size):
            ax[row][col].imshow(activation[0, :, :, activation_index], cmap='gray')
            activation_index += 1
plt.imshow(test_image)
plt.imshow(X_train[10][:,:,0]);
display_activation(activations, 8, 4, 3)

```

efficient Net80 model

```
In [18]: test_image_path = 'D:/Harold/MyDNN/DataSet/Chest_xray_seperate/PNEUMONIA/person11_bacteria_45.jpeg'
test_image = image.load_img(test_image_path, target_size=(224, 224))
x = image.img_to_array(test_image)
x = np.expand_dims(x, axis=0)
x = preprocess_input(x)
print (x.shape)
```

```
# if num_channel==1:
#     if (K.image_data_format() == 'channels_first'):
#         test_image= np.expand_dims(test_image, axis=0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=3)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
# else:
#     if (K.image_data_format() == 'channels_first'):
#         test_image=np.rollaxis(test_image,2,0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)

# Predicting the test image
yhat = custom_resnet_model.predict(x)
print(yhat)
# print(custom_resnet_model.predict_classes(x))
label = decode_predictions(yhat)
# retrieve the most likely result, e.g. highest probability
label = label[0][0]
```

```
(1, 224, 224, 3)
[[0.00159899 0.998401  ]]
```

```
-----
ValueError                                Traceback (most recent call last)
```

```
<ipython-input-18-505048f79341> in <module>
    30 print(yhat)
    31 # print(custom_resnet_model.predict_classes(x))
--> 32 label = decode_predictions(yhat)
    33 # retrieve the most likely result, e.g. highest probability
    34 label = label[0][0]
```

```
D:\Anaconda3\lib\site-packages\tensorflow\python\keras\applications\inception_v3.py in decode_predictions(preds, top)
    412 @keras_export('keras.applications.inception_v3.decode_predictions')
    413 def decode_predictions(preds, top=5):
--> 414     return imagenet_utils.decode_predictions(preds, top=top)
    415
    416
```

```
D:\Anaconda3\lib\site-packages\tensorflow\python\keras\applications\imagenet_utils.py in decode_predictions(preds, top)
    149         'a batch of predictions '
    150         '(i.e. a 2D array of shape (samples, 1000)). '
--> 151         'Found array with shape: ' + str(preds.shape))
    152     if CLASS_INDEX is None:
    153         fpath = data_utils.get_file(
```

```
ValueError: `decode_predictions` expects a batch of predictions (i.e. a 2D array of shape (samples, 1000)). Found array with shape:
(1, 2)
```

Visualizing the intermediate layer

```

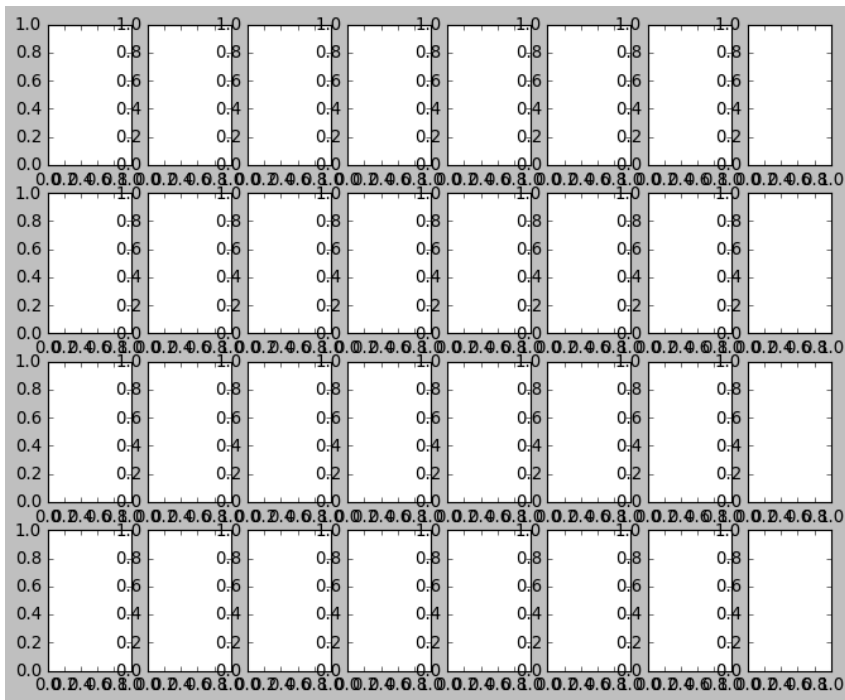
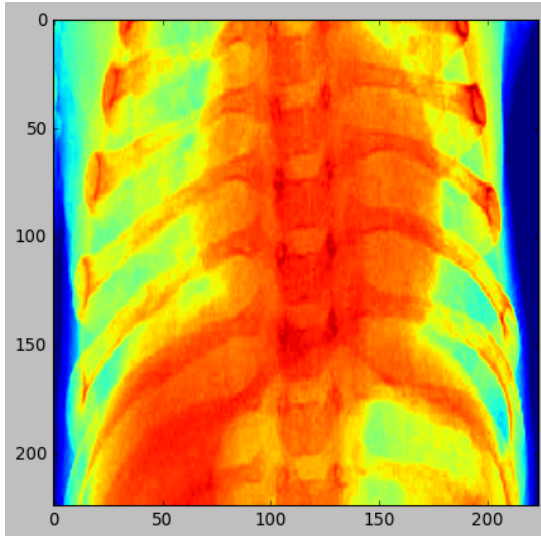
In [19]: from keras.models import Model
layer_outputs = [layer.output for layer in model.layers]
activation_model = Model(inputs=custom_resnet_model.input, outputs=layer_outputs)
activations = custom_resnet_model.predict(X_train[10].reshape(1,224,224,3))
print(activations.shape)
def display_activation(activations, col_size, row_size, act_index):
    activation = activations[0, act_index]
    activation_index=1
    fig, ax = plt.subplots(row_size, col_size, figsize=(row_size*2.5,col_size*1))
    for row in range(0,row_size):
        for col in range(0,col_size):
            ax[row][col].imshow(activation[0, :, :, activation_index], cmap='gray')
            activation_index += 1
plt.imshow(test_image)
plt.imshow(X_train[10][:,:,0]);
display_activation(activations, 8, 4, 1)

```

(1, 2)

```
-----  
IndexError                                Traceback (most recent call last)  
<ipython-input-19-32e820fb41b> in <module>  
    14 plt.imshow(test_image)  
    15 plt.imshow(X_train[10][:,:,0]);  
--> 16 display_activation(activations, 8, 4, 1)  
  
<ipython-input-19-32e820fb41b> in display_activation(activations, col_size, row_size, act_index)  
    10     for row in range(0,row_size):  
    11         for col in range(0,col_size):  
--> 12             ax[row][col].imshow(activation[0, :, :, activation_index], cmap='gray')  
    13             activation_index += 1  
    14 plt.imshow(test_image)
```

IndexError: invalid index to scalar variable.



Inception V3 visualization

```

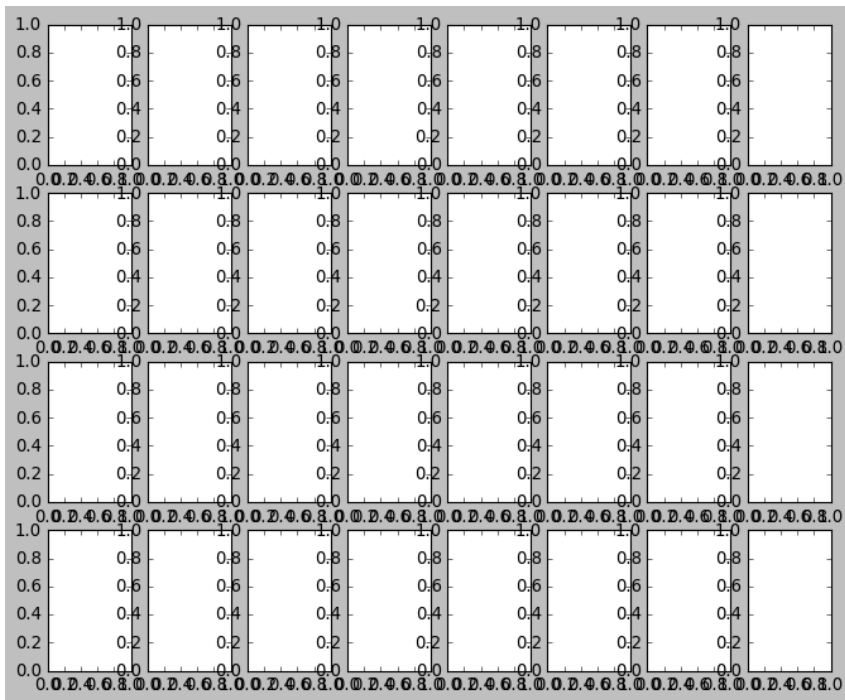
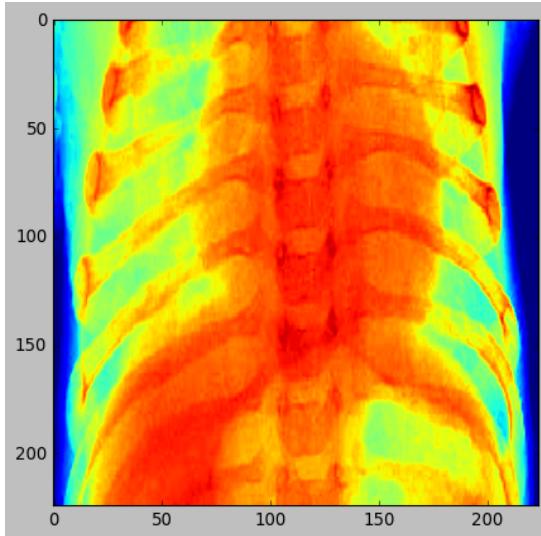
In [19]: from keras.models import Model
layer_outputs = [layer.output for layer in model.layers]
activation_model = Model(inputs=custom_resnet_model.input, outputs=layer_outputs)
activations = custom_resnet_model.predict(X_train[10].reshape(1,224,224,3))
print(activations.shape)
def display_activation(activations, col_size, row_size, act_index):
    activation = activations[0, act_index]
    activation_index=1
    fig, ax = plt.subplots(row_size, col_size, figsize=(row_size*2.5,col_size*1))
    for row in range(0,row_size):
        for col in range(0,col_size):
            ax[row][col].imshow(activation[0, :, :, activation_index], cmap='gray')
            activation_index += 1
plt.imshow(test_image)
plt.imshow(X_train[10][:,:,0]);
display_activation(activations, 8, 4, 1)

```

(1, 2)

```
-----  
IndexError                                Traceback (most recent call last)  
<ipython-input-19-32e820fb41b> in <module>  
    14 plt.imshow(test_image)  
    15 plt.imshow(X_train[10][:,:,0]);  
--> 16 display_activation(activations, 8, 4, 1)  
  
<ipython-input-19-32e820fb41b> in display_activation(activations, col_size, row_size, act_index)  
    10     for row in range(0,row_size):  
    11         for col in range(0,col_size):  
--> 12             ax[row][col].imshow(activation[0, :, :, activation_index], cmap='gray')  
    13             activation_index += 1  
    14 plt.imshow(test_image)
```

IndexError: invalid index to scalar variable.



```
In [17]: score = custom_resnet_model.evaluate(X_test, y_test, verbose=0)
print('Test Loss:', score[0])
print('Test accuracy:', score[1])

test_image = X_test[0:1]
print (test_image.shape)

print(model.predict(test_image))
print(model.predict_classes(test_image))
print(y_test[0:1])
```

Test Loss: 0.24162615835666656

Test accuracy: 0.914675772190094

(1, 224, 224, 3)

[[8.33914601e-05 9.43122213e-05 6.17418555e-05 6.54168252e-05
1.13233073e-04 4.74682347e-05 5.82300017e-05 1.08953674e-04
2.47885200e-05 1.91886254e-04 7.08437146e-05 3.57683566e-05
2.95793929e-04 5.55347469e-05 2.16536369e-04 6.21841609e-05
3.03070352e-04 4.57870956e-05 1.19120559e-04 8.90225492e-05
1.66406928e-04 9.70872716e-05 1.62744705e-04 1.49278203e-04
7.53177155e-05 1.09933528e-04 2.17667766e-04 1.03260376e-04
1.68310271e-05 4.71250714e-05 2.25656520e-04 1.11117632e-04
1.19079450e-04 3.50336049e-05 7.37054288e-05 1.12858543e-04
1.31092936e-04 5.13734412e-05 1.09661698e-04 1.33358277e-04
1.32257570e-04 3.87834370e-05 6.33041956e-04 8.69823198e-05
1.75937254e-04 4.36656082e-05 3.17085214e-04 4.07961088e-05
3.52774805e-05 9.89007895e-05 3.13197001e-04 3.97188778e-05
3.01094784e-04 2.42384500e-04 3.31761781e-04 3.03487395e-05
3.10513788e-05 1.01327387e-04 8.12207290e-04 2.79306027e-04
7.95677552e-05 3.58391902e-04 2.18163841e-04 1.20014038e-05
3.81811369e-05 1.32926434e-04 2.31196143e-04 2.24524265e-05
9.85946317e-05 1.02861639e-04 1.61079763e-04 2.17821958e-04
3.50604132e-05 2.44382943e-04 6.57923752e-04 1.51681917e-04
2.78034480e-04 1.11821643e-03 1.29941225e-04 7.72509258e-04
3.47995025e-04 1.38023504e-04 6.40542203e-05 4.19817152e-05
4.27196617e-04 4.55504705e-05 4.25938524e-05 6.52940753e-06
2.40640133e-04 1.50604916e-04 5.46822557e-04 1.12956704e-03
1.23571826e-03 6.57338824e-05 9.26602224e-05 4.90679340e-05
2.16124783e-04 5.13464547e-05 7.67093807e-05 4.40209726e-04
1.56303606e-04 3.04156449e-04 3.77853285e-05 1.33213616e-05
7.79672555e-05 1.12029833e-04 4.41315169e-05 5.07919642e-04
2.39844224e-03 1.99436821e-04 1.03430102e-04 1.63807563e-05
4.74627122e-05 1.43858822e-04 3.36380654e-05 9.86379018e-05
3.65483051e-04 2.97627383e-04 3.29633949e-05 1.55277128e-04
1.97401114e-05 5.45085131e-05 7.30986221e-06 7.17431467e-05
1.04610435e-05 1.34972914e-04 8.48182535e-05 5.58899483e-04
8.61329172e-05 1.34347181e-03 4.01100580e-04 9.56244621e-05
5.89205556e-05 7.04675767e-05 1.13356138e-04 1.90605904e-04
5.86017450e-05 2.83766531e-05 2.41395646e-05 2.35308864e-04
1.98521317e-04 9.12860487e-05 2.52444355e-04 1.62257536e-04
6.40077284e-04 1.32496934e-04 2.10071681e-04 6.35759934e-05
1.41753102e-04 1.35031165e-04 5.41926653e-04 3.66952663e-05
3.07578885e-05 6.28180278e-04 2.15198015e-05 8.01470087e-05
8.16880274e-05 8.74915640e-05 4.15920149e-05 7.44503195e-05
8.24786548e-05 5.42770031e-05 5.48054668e-05 4.74896988e-05
9.28548325e-05 4.52796885e-05 1.77980764e-05 1.84390505e-04
1.98084726e-05 1.04686733e-04 1.61515454e-05 1.20808099e-05
1.41941000e-05 6.96301358e-05 2.30870028e-05 1.98415233e-04
1.17952135e-04 5.10479294e-05 1.49995074e-04 8.08492041e-05
3.51375347e-05 2.29028607e-04 8.90064184e-05 3.12090066e-04
1.59218849e-04 4.08091466e-04 2.47155094e-05 6.27728441e-05
5.85914713e-05 1.24783706e-04 8.47743824e-04 2.00780225e-04
1.14579025e-05 4.24381542e-05 2.20518123e-04 2.41503993e-04
3.80996120e-04 1.66334023e-04 4.57426009e-04 2.28657053e-04
8.80086518e-05 2.42261885e-04 1.38663643e-04 1.01071002e-03
5.62849418e-05 1.17982512e-04 8.66682560e-04 2.85082497e-04
5.68611940e-05 3.75254785e-05 3.42668318e-05 9.50891044e-05
5.70256343e-05 1.05972678e-04 3.37440358e-03 1.68348161e-05
1.13453483e-03 9.97516909e-05 2.97156566e-05 9.10738425e-04
8.94142868e-05 3.80023732e-04 1.13284055e-04 1.53732763e-05
6.09609342e-05 4.30636464e-05 7.16900686e-05 1.03029714e-04
1.17967102e-04 2.91551783e-04 2.07727862e-05 2.68855474e-05
3.98764096e-04 1.12831323e-04 5.65429684e-04 1.08823864e-04
2.27547629e-04 8.15887543e-05 1.31144631e-03 1.72377404e-04
3.01062464e-05 1.25550898e-04 7.60590920e-05 1.35162845e-04
3.01215259e-05 7.59201357e-04 1.18775715e-04 1.13286223e-04
9.98448813e-05 5.13336236e-05 5.72519348e-05 2.53097154e-04
1.25184335e-04 2.11916649e-05 1.37296462e-04 3.80941747e-05
6.89057060e-06 1.19045166e-04 2.11454069e-04 4.46508348e-05
4.96068569e-05 4.19014759e-05 2.39833011e-04 1.74566085e-05
3.02299904e-05 6.74980110e-05 3.35405115e-04 9.43383129e-05
5.71627970e-05 1.47618804e-04 2.56361207e-04 8.28940611e-05
1.76530288e-04 5.17332774e-05 1.28926558e-05 2.16640274e-05
5.26299800e-06 6.59979298e-04 1.14130096e-04 2.94375030e-04
4.49465224e-05 3.86519037e-04 4.72550419e-05 5.00234251e-04
1.12830836e-03 8.48204363e-05 5.19808455e-05 9.09078008e-05
1.57584349e-04 2.78345804e-04 2.19877635e-04 4.03005979e-05
2.55742081e-04 1.43388272e-04 4.44702513e-04 2.79434986e-04
5.58324682e-04 5.39616995e-05 3.54843542e-05 1.03632403e-04
4.92443869e-05 2.78861407e-04 1.90620158e-05 8.88124196e-05
9.66425578e-05 6.71750022e-05 2.91049073e-04 7.77899695e-05
1.15416595e-03 1.02880273e-04 6.21788786e-05 2.89910240e-04
1.45386657e-04 1.36766161e-04 8.10648140e-04 6.69935369e-04
5.99710562e-04 3.34999604e-05 7.04395570e-05 2.60458299e-04
2.57278443e-04 1.14066189e-03 4.60094998e-05 4.41401935e-04
2.84365797e-05 3.70950802e-05 1.20685632e-04 1.47119304e-03
2.59156623e-05 2.63112506e-05 1.31254434e-04 2.98050581e-05
1.61606848e-04 5.05411881e-05 6.80081712e-06 9.02227985e-05
4.81993848e-05 1.60939198e-05 1.06712716e-04 2.13666572e-04
5.58225962e-04 2.65355629e-04 3.55328120e-05 5.77366372e-05
1.57414470e-05 7.45292418e-05 2.90369026e-05 3.63006620e-05
3.77016753e-04 8.69203868e-05 5.23683120e-05 3.87982968e-04

8.73839963e-05 3.36109719e-04 1.59285133e-04 3.57647252e-04
7.11770990e-05 4.68818267e-04 1.19898650e-04 2.17655106e-04
1.07248547e-04 2.41937068e-05 6.84147426e-06 1.47503679e-05
2.38636349e-05 2.54967075e-04 4.44853285e-05 1.28478190e-04
2.98674451e-04 2.83423171e-04 2.91537610e-04 4.44025900e-05
1.02678205e-05 2.48706638e-05 2.28041517e-05 6.26483583e-04
1.77620866e-04 1.27518970e-05 8.18574772e-05 3.33283795e-04
1.51482218e-05 3.01448337e-04 1.46569782e-05 2.15305390e-05
1.31581663e-04 2.44356837e-04 8.89293879e-05 9.52232804e-04
1.69589208e-03 1.24380473e-04 1.11265399e-05 6.32024094e-05
1.05321618e-04 1.18508018e-04 1.12465736e-04 2.44203184e-05
2.22543022e-04 5.47238378e-05 4.02125734e-04 2.07621395e-03
1.29073765e-03 4.26254934e-04 3.13847762e-04 7.91640923e-05
3.74486481e-05 6.06671856e-05 1.01255473e-05 3.70270791e-05
1.40693897e-04 1.74860365e-03 1.51431232e-05 9.23906977e-04
1.58916961e-03 2.58721877e-04 1.62998028e-02 4.60242045e-05
1.30477099e-04 8.42971145e-04 6.61320693e-04 1.53327564e-04
1.76025023e-05 2.58769403e-04 1.16294796e-04 2.02700309e-03
1.35269817e-02 1.61069701e-03 6.28815324e-05 4.83175245e-05
8.54622282e-04 3.00164014e-04 1.86769132e-04 8.30927311e-05
1.89164875e-05 1.16337396e-04 3.49664151e-05 5.81785971e-05
1.28549105e-03 7.51490414e-04 1.59569143e-04 1.30329296e-04
6.99314696e-05 4.23135440e-04 1.16385119e-04 1.07296789e-02
8.77031707e-05 8.07532706e-05 6.63905928e-04 8.49344142e-05
2.24430609e-04 1.57332077e-04 8.75685691e-06 7.69934268e-04
2.06400455e-05 3.19353530e-05 5.70448901e-05 1.94806253e-05
1.05689175e-03 3.56638891e-04 8.67792005e-06 2.08657671e-04
8.68590578e-05 3.12234624e-03 7.49109677e-05 2.20658985e-04
5.02509007e-04 3.99989784e-02 8.39983113e-05 7.98574547e-05
5.47855336e-04 1.25976512e-05 5.13877967e-05 9.06003697e-05
5.23220806e-05 9.12444102e-06 2.34634455e-04 6.79804885e-04
3.31446150e-04 1.17720541e-04 9.75174538e-04 9.81520046e-04
9.12562013e-04 3.47146924e-05 1.41495068e-04 3.66432505e-04
6.45259570e-05 1.28914064e-04 7.21394317e-05 2.28986819e-03
3.76409618e-03 2.80516082e-03 1.48055784e-04 1.07101614e-05
4.61423806e-05 1.39898417e-04 1.82776726e-04 1.25178485e-04
3.83468068e-05 2.35304586e-03 3.29897623e-04 1.33856607e-03
1.06934596e-04 7.09476590e-04 8.69219512e-05 1.59723614e-03
1.43893967e-05 4.45995526e-03 6.88059226e-05 2.52713544e-05
2.10724911e-03 3.38549835e-05 2.71558602e-05 4.11760557e-04
1.15444149e-04 2.09193953e-04 1.98860871e-04 3.77124990e-04
1.20693345e-04 9.27479705e-04 3.17542319e-04 1.70437590e-04
2.65740935e-04 9.83073642e-06 1.02659171e-04 2.55792460e-04
1.47136417e-03 3.02899862e-05 1.36041519e-04 9.07273439e-04
5.02650626e-04 8.89950534e-05 1.07500923e-03 2.25143507e-04
2.58525215e-05 2.61601672e-04 1.77471444e-03 4.47884595e-05
1.29528707e-04 1.72686905e-05 1.25372258e-04 1.40629854e-04
1.54930531e-05 2.03966832e-04 1.26344059e-03 1.41774456e-03
3.88985645e-05 7.27562001e-05 2.10891685e-05 2.49873228e-05
1.81706215e-04 2.27203942e-03 8.91905569e-04 1.27628358e-04
1.71735854e-04 3.64318621e-05 7.76324887e-05 2.27155426e-04
3.80234778e-01 1.11255853e-03 4.91088031e-05 7.29268868e-05
6.97635114e-04 2.08517420e-04 4.85319458e-03 8.05228367e-04
1.44731428e-04 9.00824016e-05 5.56111154e-05 3.25820292e-05
6.10917123e-05 4.72707288e-05 3.22346226e-04 5.27381853e-05
3.36610625e-04 1.03090955e-04 2.43045681e-04 4.20626864e-04
1.52519468e-04 6.08003756e-04 5.78366977e-04 6.52293442e-04
6.34193653e-04 7.08757085e-04 9.81467194e-04 1.93246495e-04
2.93523885e-06 5.46953233e-04 7.79101756e-05 1.26446816e-04
3.07050510e-03 7.41158001e-05 2.09731492e-03 1.64307945e-03
1.22230573e-04 5.14956249e-04 2.00441311e-04 9.52015325e-05
4.60930576e-04 1.96795794e-04 5.05214091e-04 1.70887785e-03
5.94898360e-04 2.80495060e-05 6.12656368e-05 4.76151326e-04
6.06533897e-04 6.90728833e-04 2.45146477e-03 3.85669337e-05
1.50876329e-03 1.49309664e-04 8.26684013e-03 1.86099369e-05
3.37498757e-04 8.78973442e-05 2.75093160e-04 1.37260722e-04
2.65541067e-03 3.17754515e-04 1.07748630e-04 9.04394081e-04
1.07567979e-03 3.74500407e-04 5.36410778e-04 2.80095974e-05
2.61744281e-05 1.83778873e-04 1.61253134e-04 5.47529125e-05
7.77427267e-05 3.06203519e-03 1.72918866e-04 9.53771960e-05
2.25956505e-03 3.95534909e-04 6.81745078e-05 1.34670379e-04
1.79122146e-02 1.01388141e-03 6.40834635e-03 4.89571248e-04
4.02923361e-05 4.73684740e-05 1.40673510e-04 1.73157154e-04
1.48242374e-03 8.56971892e-05 1.73195396e-04 8.43464513e-05
2.08894999e-04 4.32949892e-05 6.85631094e-05 6.85166684e-04
5.59892542e-05 1.69968873e-04 8.19040215e-05 5.71506680e-04
3.53219875e-05 1.04701510e-04 2.14931388e-05 3.53869582e-05
3.78410798e-04 5.04389638e-04 3.27573434e-05 2.83361995e-04
2.72948737e-03 9.21294268e-05 1.48180025e-05 2.46270763e-04
4.13689813e-05 4.71792555e-05 1.51615334e-03 3.52851202e-05
9.80600948e-04 2.52885919e-04 1.63716366e-04 3.37931451e-05
1.07107393e-04 6.50193251e-04 2.43731687e-04 6.05064961e-05
1.35610229e-04 6.65127416e-04 3.26190260e-04 7.57016824e-05
3.46393699e-06 6.07279690e-05 3.57153876e-05 7.71200066e-05
1.00423140e-03 1.00033205e-04 5.33735838e-05 1.09746339e-04
2.88255571e-04 9.57804441e-05 3.41402010e-05 2.11888662e-04
1.22098456e-04 6.21896761e-05 9.76745723e-05 8.55488179e-05
9.62076883e-04 4.20940109e-04 5.54108410e-05 1.32340530e-04
4.31249244e-03 5.03420888e-05 8.13174192e-05 4.18817438e-03
7.12255451e-06 9.48144123e-04 2.42499755e-05 7.69914873e-03
5.26841131e-06 2.83917616e-04 4.28069943e-05 1.27932144e-05

2.66362971e-04 1.69233754e-04 4.73914290e-04 1.19051627e-04
1.91241088e-05 5.86830685e-03 7.07294850e-04 2.50239773e-05
3.75086616e-04 1.55779562e-04 1.32933401e-05 4.63665128e-05
4.63213632e-03 1.30101602e-04 6.92261747e-05 1.00446850e-04
1.05526822e-03 1.89026483e-04 9.04939006e-05 1.03394501e-04
9.69200992e-05 5.64555405e-04 2.52025693e-05 3.57972931e-05
4.88542646e-05 1.33715745e-04 1.04600520e-04 9.90291475e-04
1.22292840e-04 7.27627921e-05 2.14016090e-05 3.12159164e-03
2.09966645e-04 1.85880926e-04 1.90800958e-04 5.63547364e-04
2.16300876e-04 2.20534217e-04 6.04555644e-05 6.00815147e-05
2.14696829e-05 1.93196393e-04 6.29920323e-05 1.38893153e-03
1.97276357e-03 3.87275737e-04 4.70409956e-04 1.04977466e-04
8.66038899e-05 2.10130784e-05 2.21918861e-04 1.92755160e-05
9.36520300e-05 3.28548958e-05 1.16980786e-03 4.76077330e-05
5.36649197e-04 1.12778936e-04 3.09132611e-06 1.16967904e-05
3.23557877e-04 2.54508495e-06 4.74176253e-04 6.56044402e-04
8.27612123e-04 6.55535914e-05 1.99410575e-03 1.48264132e-03
8.31854195e-05 9.95933297e-05 4.19858785e-04 4.76322530e-05
4.94358654e-04 2.63915467e-03 7.25556747e-04 7.74318527e-04
7.02826772e-04 3.97677359e-05 9.80811380e-03 2.64382761e-05
4.16843759e-05 7.06135994e-04 2.61750283e-05 8.92328855e-04
1.27251481e-03 1.58596595e-04 2.71753408e-04 6.71056274e-04
1.84216391e-04 3.36448866e-04 4.11851470e-05 1.33671318e-04
1.52624969e-04 2.06730019e-05 3.06273869e-04 2.71964981e-03
4.70857427e-04 3.60155609e-05 7.55196816e-05 1.57036399e-03
1.54864065e-05 2.31870697e-04 1.19008953e-02 3.09153111e-04
1.25879757e-04 1.03011444e-04 1.46702502e-03 7.43132041e-05
2.84773414e-05 5.63574489e-04 1.61093209e-04 8.43522896e-04
9.58754797e-04 5.08159465e-05 2.06946715e-04 6.66140520e-04
5.87129070e-05 2.06479308e-06 2.25120777e-04 1.80725110e-05
1.10015128e-04 4.83739830e-04 1.34500238e-04 4.28811356e-04
5.18853834e-04 1.01364413e-02 3.64524050e-04 5.43028364e-05
2.39440371e-04 2.05363482e-04 1.51743268e-04 9.51730253e-05
1.73279313e-05 5.83487927e-05 1.47452345e-04 3.08834296e-03
1.24863713e-04 1.79667713e-05 3.07209697e-03 2.79738251e-05
7.99115587e-05 2.34446015e-05 9.82822618e-04 2.34153867e-03
1.13934024e-04 2.91827037e-05 1.47375016e-04 5.59431974e-05
1.17024174e-04 1.57576782e-04 3.46450892e-04 7.32991204e-04
1.69161067e-04 1.59455743e-03 1.73376029e-05 8.59025677e-05
1.01042865e-03 1.09213754e-04 1.79306717e-05 5.39152825e-04
1.89648981e-05 7.20300188e-04 1.79962174e-03 3.27884639e-03
7.25260616e-05 3.15109646e-04 7.32668326e-04 1.53096189e-05
4.52849163e-05 3.47554538e-04 2.79870495e-04 6.85701016e-05
1.07863576e-04 1.66341953e-04 1.45367114e-04 1.51955945e-04
1.18492193e-04 7.17391435e-04 2.06604693e-03 2.06299519e-04
2.54458486e-04 6.54346426e-04 2.97473860e-04 7.34544301e-04
6.25758403e-05 2.90876233e-05 1.74098328e-04 1.11928071e-04
1.14851497e-01 2.62955250e-03 2.40852195e-03 1.72162443e-04
8.75283731e-05 6.66758686e-04 1.76057420e-05 1.84296805e-04
1.70327738e-04 2.79838045e-04 4.37207578e-04 8.90741474e-04
2.95937969e-03 3.18998966e-04 7.71548730e-05 1.85021563e-04
1.17419753e-03 2.78085681e-05 9.33784249e-05 6.90118832e-05
4.32995730e-05 2.92342224e-06 4.15394534e-05 1.52136008e-05
1.29134523e-05 2.94119789e-04 9.48211455e-05 6.97282157e-05
6.17292535e-05 4.16714574e-05 5.00194401e-05 1.13236427e-04
5.65801565e-06 1.36517001e-05 4.56229645e-06 5.45705625e-05
1.98253110e-05 1.48473191e-04 6.39737045e-05 2.67284995e-05
4.29977372e-05 1.51909131e-04 1.80505929e-04 3.08210932e-04
2.26825607e-04 7.97288594e-05 9.52363989e-05 4.29513457e-05
5.10523641e-05 2.09242899e-05 6.18914710e-05 5.63330032e-05
1.90053997e-05 6.33332165e-05 5.60914486e-05 3.97748627e-05
1.83574355e-06 1.78109458e-05 5.12331644e-05 2.07487301e-05
3.00131378e-05 2.81950579e-05 4.41239914e-03 1.48228282e-04
8.43753296e-05 5.18359993e-05 9.11546085e-05 5.44237951e-03
1.05446445e-04 7.14217837e-04 6.95484108e-04 1.59877454e-04
1.04986575e-04 1.55470407e-04 5.15914580e-04 2.69758631e-04
5.53757884e-04 9.67251690e-05 1.41127763e-04 5.17273264e-04
1.65178091e-04 4.38820600e-04 7.76213055e-05 7.71105988e-04
8.71600205e-05 6.32327283e-04 2.78978841e-04 3.60339385e-04
1.25893086e-04 1.04214778e-04 5.29835888e-05 8.40803841e-05
3.67418434e-05 7.11586326e-05 4.39933137e-05 1.84522447e-04]]

```
-----  
AttributeError                                Traceback (most recent call last)  
<ipython-input-17-83421ec204f5> in <module>  
      7  
      8 print(model.predict(test_image))  
---->  9 print(model.predict_classes(test_image))  
     10 print(y_test[0:1])  
  
AttributeError: 'Functional' object has no attribute 'predict_classes'
```

Testing a new image

```
In [18]: test_image_path = 'D:/Harold/MyDNN/DataSet/Chest_xray_seperate/PNEUMONIA/person11_bacteria_45.jpeg'
test_image = image.load_img(test_image_path, target_size=(224, 224))
x = image.img_to_array(test_image)
x = np.expand_dims(x, axis=0)
x = preprocess_input(x)
print (x.shape)
```

```
# if num_channel==1:
#     if (K.image_data_format() == 'channels_first'):
#         test_image= np.expand_dims(test_image, axis=0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=3)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
# else:
#     if (K.image_data_format() == 'channels_first'):
#         test_image=np.rollaxis(test_image,2,0)
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)
#     else:
#         test_image= np.expand_dims(test_image, axis=0)
#         print (test_image.shape)

# Predicting the test image
yhat = custom_resnet_model.predict(x)
print(yhat)
# print(custom_resnet_model.predict_classes(x))
label = decode_predictions(yhat)
# retrieve the most likely result, e.g. highest probability
label = label[0][0]
```

```
(1, 224, 224, 3)
[[0.04744451 0.95255554]]
```

```
-----
ValueError                                Traceback (most recent call last)
<ipython-input-18-505048f79341> in <module>
    30 print(yhat)
    31 # print(custom_resnet_model.predict_classes(x))
--> 32 label = decode_predictions(yhat)
    33 # retrieve the most likely result, e.g. highest probability
    34 label = label[0][0]
```

```
D:\Anaconda3\lib\site-packages\tensorflow\python\keras\applications\inception_v3.py in decode_predictions(preds, top)
    412 @keras_export('keras.applications.inception_v3.decode_predictions')
    413 def decode_predictions(preds, top=5):
--> 414     return imagenet_utils.decode_predictions(preds, top=top)
    415
    416
```

```
D:\Anaconda3\lib\site-packages\tensorflow\python\keras\applications\imagenet_utils.py in decode_predictions(preds, top)
    149     'a batch of predictions '
    150     '(i.e. a 2D array of shape (samples, 1000)). '
--> 151     'Found array with shape: ' + str(preds.shape))
    152     if CLASS_INDEX is None:
    153         fpath = data_utils.get_file(
```

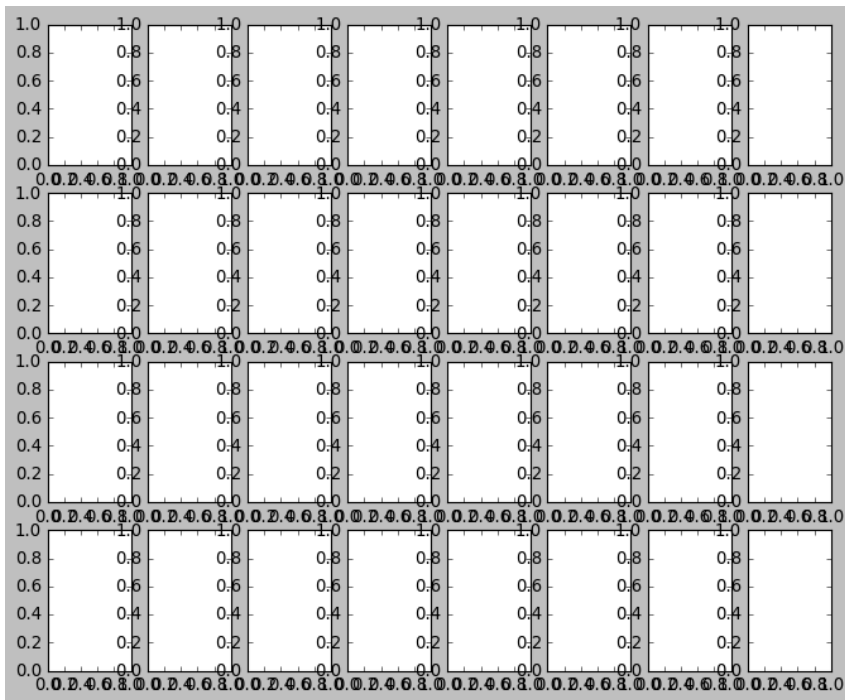
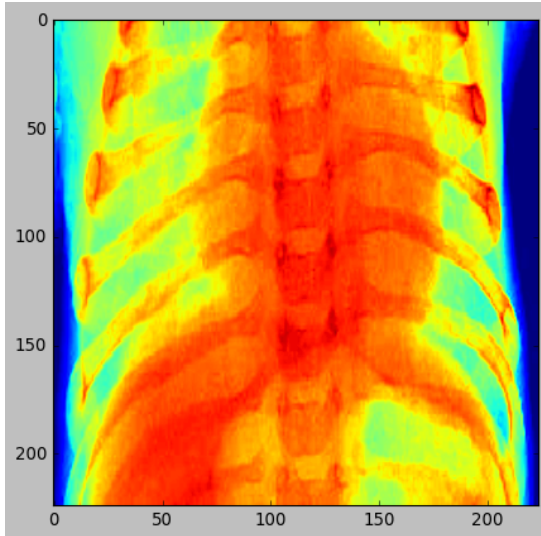
```
ValueError: `decode_predictions` expects a batch of predictions (i.e. a 2D array of shape (samples, 1000)). Found array with shape:
(1, 2)
```

```
In [19]: from keras.models import Model
layer_outputs = [layer.output for layer in model.layers]
activation_model = Model(inputs=custom_resnet_model.input, outputs=layer_outputs)
activations = custom_resnet_model.predict(X_train[10].reshape(1,224,224,3))
print(activations.shape)
def display_activation(activations, col_size, row_size, act_index):
    activation = activations[0, act_index]
    activation_index=1
    fig, ax = plt.subplots(row_size, col_size, figsize=(row_size*2.5,col_size*1))
    for row in range(0,row_size):
        for col in range(0,col_size):
            ax[row][col].imshow(activation[0, :, :, activation_index], cmap='gray')
            activation_index += 1
plt.imshow(test_image)
plt.imshow(X_train[10][:,:,0]);
display_activation(activations, 8, 4, 1)
```

(1, 2)

```
-----  
IndexError                                Traceback (most recent call last)  
<ipython-input-19-32e820fb41b> in <module>  
    14 plt.imshow(test_image)  
    15 plt.imshow(X_train[10][:,:,0]);  
--> 16 display_activation(activations, 8, 4, 1)  
  
<ipython-input-19-32e820fb41b> in display_activation(activations, col_size, row_size, act_index)  
    10     for row in range(0,row_size):  
    11         for col in range(0,col_size):  
--> 12             ax[row][col].imshow(activation[0, :, :, activation_index], cmap='gray')  
    13             activation_index += 1  
    14 plt.imshow(test_image)
```

IndexError: invalid index to scalar variable.



Overall breadth and depth of commentary/ explanations ¶

In this report, we have conducted 5 models with are Resnet50, mobileNetV2, EfficientNet80, VGG19 and InceptionV3. We have also 2 classs 0 and 1 which is normal and Pneumonia. In the MobileNetV2 we see that in class 0, the precision is 0.88, recall is 0.73, f1-score is 0.83 and support is 312. In class 1, the precision is 0.93, recall is 0.96, f1-score is 0.94 and support is 860. In the VGG19 we see that in class 0, the precision is 0.89, recall is 0.85, f1-score is 0.87 and support is 312. In class 1, the precision is 0.95, recall is 0.96, f1-score is 0.95 and support is 860. In the Inception V3 we see that in class 0, the precision is 0.93, recall is 0.70, f1-score is 0.79 and support is 312. In class 1, the precision is 0.90, recall is 0.98, f1-score is 0.94 and support is 860. In the ResNet50 we see that in class 0, the precision is 0.94, recall is 0.96, f1-score is 0.95 and support is 312. In class 1, the precision is 0.99, recall is 0.98, f1-score is 0.98 and support is 860. In the EfficientNetB0 we see that in class 0, the precision is 0.95, recall is 0.94, f1-score is 0.94 and support is 312. In class 1, the precision is 0.98, recall is 0.98, f1-score is 0.98 and support is 860. Based on the f1-score, the best models used in the x-ray dataset is EfficientNetB0.

In []: