```python
# import keras
# from keras.datasets import cifar10
# from keras.models import Model, Sequential
# from keras.layers import Dense, Dropout, Flatten, Input, AveragePooling2D, merge, Activation
# from keras.layers import Conv2D, MaxPooling2D, BatchNormalization
# from keras.layers import Concatenate
# from keras.optimizers import Adam
from tensorflow.keras import models, layers
from tensorflow.keras.models import Model
from tensorflow.keras.layers import BatchNormalization, Activation, Flatten
from tensorflow.keras.optimizers import Adam
import numpy as np


# this part will prevent tensorflow to allocate all the avaliable GPU Memory
# backend
import tensorflow as tf


# Hyperparameters
batch_size = 128
num_classes = 10
epochs = 50
l = 40
num_filter = 12
compression = 0.5
compression_1 = 2
compression_2 = 4
dropout_rate = 0.2


# Load CIFAR10 Data
(X_train, y_train), (X_test, y_test) = tf.keras.datasets.cifar10.load_data()
img_height, img_width, channel = X_train.shape[1],X_train.shape[2],X_train.shape[3]

# convert to one hot encoing
y_train = tf.keras.utils.to_categorical(y_train, num_classes)
y_test = tf.keras.utils.to_categorical(y_test, num_classes)
```

```
Downloading data from https://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz
170500096/170498071 [==============================] - 11s 0us/step
```

```python
X_train=X_train.astype("float32")
X_test=X_test.astype("float32")
mean=np.mean(X_train)
std=np.std(X_train)
X_test=(X_test-mean)/std
X_train=(X_train-mean)/std
```

```python
X_train.shape
```

```
(50000, 32, 32, 3)
```

```python
X_test.shape
```

```
(10000, 32, 32, 3)
```

Image Augmentation

```python
from keras.preprocessing.image import ImageDataGenerator
datagen = ImageDataGenerator(width_shift_range=0.3, height_shift_range=0.3, horizontal_flip=True,
datagen.fit(X_train)
img_train = datagen.flow(X_train, y_train, batch_size=64)
steps = int(X_train.shape[0] / 64)
```

```
Using TensorFlow backend.
```

Kernel of size 3x3 running for 50 epochs

```python
# Dense Block
def denseblock(input, num_filter = 12, dropout_rate = 0.2):
    global compression
```

```python
        global compression_1
        global compression_2
        temp = input
        for _ in range(l):
            BatchNorm = layers.BatchNormalization()(temp)
            relu = layers.Activation('relu')(BatchNorm)
            Conv2D_3_3 = layers.Conv2D(int(num_filter*compression), (3,3), use_bias=False ,padding='s
            #if dropout_rate>0:
                #Conv2D_3_3 = layers.Dropout(dropout_rate)(Conv2D_3_3)
            concat = layers.Concatenate(axis=-1)([temp,Conv2D_3_3])

            temp = concat

        return temp


    ## transition Block
    def transition(input, num_filter = 12, dropout_rate = 0.2):
        global compression
        BatchNorm = layers.BatchNormalization()(input)
        relu = layers.Activation('relu')(BatchNorm)
        Conv2D_BottleNeck = layers.Conv2D(int(num_filter*compression), (1,1), use_bias=False ,padding
        #if dropout_rate>0:
            #Conv2D_BottleNeck = layers.Dropout(dropout_rate)(Conv2D_BottleNeck)
        avg = layers.AveragePooling2D(pool_size=(2,2))(Conv2D_BottleNeck)
        return avg


    #output layer
    def output_layer(input):
        global compression
        BatchNorm = layers.BatchNormalization()(input)
        relu = layers.Activation('relu')(BatchNorm)
        AvgPooling = layers.AveragePooling2D(pool_size=(2,2))(relu)
        #flat = layers.Flatten()(AvgPooling)
        First = layers.Conv2D(int(10), (2,2), use_bias=False ,padding='same')(AvgPooling)
        Maxpool = layers.GlobalMaxPooling2D()(First)
        output = layers.Activation("softmax")(Maxpool)
        return output
```

```
num_filter = 12
dropout_rate = 0.2
l = 12
input = layers.Input(shape=(img_height, img_width, channel,))
First_Conv2D = layers.Conv2D(num_filter, (3,3), use_bias=False ,padding='same')(input)

First_Block = denseblock(First_Conv2D, num_filter, dropout_rate)
First_Transition = transition(First_Block, num_filter, dropout_rate)

Second_Block = denseblock(First_Transition, num_filter, dropout_rate)
Second_Transition = transition(Second_Block, num_filter, dropout_rate)

Third_Block = denseblock(Second_Transition, num_filter, dropout_rate)
Third_Transition = transition(Third_Block, num_filter, dropout_rate)

Last_Block = denseblock(Third_Transition,  num_filter, dropout_rate)
output = output_layer(Last_Block)


#https://arxiv.org/pdf/1608.06993.pdf
from IPython.display import IFrame, YouTubeVideo
YouTubeVideo(id='-W6y8xnd--U', width=600)
```

```
model = Model(inputs=[input], outputs=[output])
model.summary()
```

⌲

```
Model: "model"
```

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 32, 32, 3)]	0	
conv2d (Conv2D)	(None, 32, 32, 12)	324	input_1[0][0]
batch_normalization (BatchNorma	(None, 32, 32, 12)	48	conv2d[0][0]
activation (Activation)	(None, 32, 32, 12)	0	batch_normalization[0][0]
conv2d_1 (Conv2D)	(None, 32, 32, 6)	648	activation[0][0]
concatenate (Concatenate)	(None, 32, 32, 18)	0	conv2d[0][0] conv2d_1[0][0]
batch_normalization_1 (BatchNor	(None, 32, 32, 18)	72	concatenate[0][0]
activation_1 (Activation)	(None, 32, 32, 18)	0	batch_normalization_1[0][0]
conv2d_2 (Conv2D)	(None, 32, 32, 6)	972	activation_1[0][0]
concatenate_1 (Concatenate)	(None, 32, 32, 24)	0	concatenate[0][0] conv2d_2[0][0]
batch_normalization_2 (BatchNor	(None, 32, 32, 24)	96	concatenate_1[0][0]
activation_2 (Activation)	(None, 32, 32, 24)	0	batch_normalization_2[0][0]
conv2d_3 (Conv2D)	(None, 32, 32, 6)	1296	activation_2[0][0]
concatenate_2 (Concatenate)	(None, 32, 32, 30)	0	concatenate_1[0][0] conv2d_3[0][0]
batch_normalization_3 (BatchNor	(None, 32, 32, 30)	120	concatenate_2[0][0]
activation_3 (Activation)	(None, 32, 32, 30)	0	batch_normalization_3[0][0]
conv2d_4 (Conv2D)	(None, 32, 32, 6)	1620	activation_3[0][0]
concatenate_3 (Concatenate)	(None, 32, 32, 36)	0	concatenate_2[0][0] conv2d_4[0][0]

conv2d_4[0][0]

batch_normalization_4 (BatchNor	(None, 32, 32, 36)	144	concatenate_3[0][0]
activation_4 (Activation)	(None, 32, 32, 36)	0	batch_normalization_4[0][0]
conv2d_5 (Conv2D)	(None, 32, 32, 6)	1944	activation_4[0][0]
concatenate_4 (Concatenate)	(None, 32, 32, 42)	0	concatenate_3[0][0] conv2d_5[0][0]
batch_normalization_5 (BatchNor	(None, 32, 32, 42)	168	concatenate_4[0][0]
activation_5 (Activation)	(None, 32, 32, 42)	0	batch_normalization_5[0][0]
conv2d_6 (Conv2D)	(None, 32, 32, 6)	2268	activation_5[0][0]
concatenate_5 (Concatenate)	(None, 32, 32, 48)	0	concatenate_4[0][0] conv2d_6[0][0]
batch_normalization_6 (BatchNor	(None, 32, 32, 48)	192	concatenate_5[0][0]
activation_6 (Activation)	(None, 32, 32, 48)	0	batch_normalization_6[0][0]
conv2d_7 (Conv2D)	(None, 32, 32, 6)	2592	activation_6[0][0]
concatenate_6 (Concatenate)	(None, 32, 32, 54)	0	concatenate_5[0][0] conv2d_7[0][0]
batch_normalization_7 (BatchNor	(None, 32, 32, 54)	216	concatenate_6[0][0]
activation_7 (Activation)	(None, 32, 32, 54)	0	batch_normalization_7[0][0]
conv2d_8 (Conv2D)	(None, 32, 32, 6)	2916	activation_7[0][0]
concatenate_7 (Concatenate)	(None, 32, 32, 60)	0	concatenate_6[0][0] conv2d_8[0][0]
batch_normalization_8 (BatchNor	(None, 32, 32, 60)	240	concatenate_7[0][0]
activation_8 (Activation)	(None, 32, 32, 60)	0	batch_normalization_8[0][0]
conv2d_9 (Conv2D)	(None, 32, 32, 6)	3240	activation_8[0][0]

concatenate_8 (Concatenate)	(None, 32, 32, 66)	0	concatenate_7[0][0] conv2d_9[0][0]
batch_normalization_9 (BatchNor	(None, 32, 32, 66)	264	concatenate_8[0][0]
activation_9 (Activation)	(None, 32, 32, 66)	0	batch_normalization_9[0][0]
conv2d_10 (Conv2D)	(None, 32, 32, 6)	3564	activation_9[0][0]
concatenate_9 (Concatenate)	(None, 32, 32, 72)	0	concatenate_8[0][0] conv2d_10[0][0]
batch_normalization_10 (BatchNo	(None, 32, 32, 72)	288	concatenate_9[0][0]
activation_10 (Activation)	(None, 32, 32, 72)	0	batch_normalization_10[0][0]
conv2d_11 (Conv2D)	(None, 32, 32, 6)	3888	activation_10[0][0]
concatenate_10 (Concatenate)	(None, 32, 32, 78)	0	concatenate_9[0][0] conv2d_11[0][0]
batch_normalization_11 (BatchNo	(None, 32, 32, 78)	312	concatenate_10[0][0]
activation_11 (Activation)	(None, 32, 32, 78)	0	batch_normalization_11[0][0]
conv2d_12 (Conv2D)	(None, 32, 32, 6)	4212	activation_11[0][0]
concatenate_11 (Concatenate)	(None, 32, 32, 84)	0	concatenate_10[0][0] conv2d_12[0][0]
batch_normalization_12 (BatchNo	(None, 32, 32, 84)	336	concatenate_11[0][0]
activation_12 (Activation)	(None, 32, 32, 84)	0	batch_normalization_12[0][0]
conv2d_13 (Conv2D)	(None, 32, 32, 6)	504	activation_12[0][0]
average_pooling2d (AveragePooli	(None, 16, 16, 6)	0	conv2d_13[0][0]
batch_normalization_13 (BatchNo	(None, 16, 16, 6)	24	average_pooling2d[0][0]
activation_13 (Activation)	(None, 16, 16, 6)	0	batch_normalization_13[0][0]
conv2d_14 (Conv2D)	(None, 16, 16, 6)	324	activation_13[0][0]

concatenate_12 (Concatenate)	(None, 16, 16, 12)	0	average_pooling2d[0][0] conv2d_14[0][0]
batch_normalization_14 (BatchNo	(None, 16, 16, 12)	48	concatenate_12[0][0]
activation_14 (Activation)	(None, 16, 16, 12)	0	batch_normalization_14[0][0]
conv2d_15 (Conv2D)	(None, 16, 16, 6)	648	activation_14[0][0]
concatenate_13 (Concatenate)	(None, 16, 16, 18)	0	concatenate_12[0][0] conv2d_15[0][0]
batch_normalization_15 (BatchNo	(None, 16, 16, 18)	72	concatenate_13[0][0]
activation_15 (Activation)	(None, 16, 16, 18)	0	batch_normalization_15[0][0]
conv2d_16 (Conv2D)	(None, 16, 16, 6)	972	activation_15[0][0]
concatenate_14 (Concatenate)	(None, 16, 16, 24)	0	concatenate_13[0][0] conv2d_16[0][0]
batch_normalization_16 (BatchNo	(None, 16, 16, 24)	96	concatenate_14[0][0]
activation_16 (Activation)	(None, 16, 16, 24)	0	batch_normalization_16[0][0]
conv2d_17 (Conv2D)	(None, 16, 16, 6)	1296	activation_16[0][0]
concatenate_15 (Concatenate)	(None, 16, 16, 30)	0	concatenate_14[0][0] conv2d_17[0][0]
batch_normalization_17 (BatchNo	(None, 16, 16, 30)	120	concatenate_15[0][0]
activation_17 (Activation)	(None, 16, 16, 30)	0	batch_normalization_17[0][0]
conv2d_18 (Conv2D)	(None, 16, 16, 6)	1620	activation_17[0][0]
concatenate_16 (Concatenate)	(None, 16, 16, 36)	0	concatenate_15[0][0] conv2d_18[0][0]
batch_normalization_18 (BatchNo	(None, 16, 16, 36)	144	concatenate_16[0][0]
activation 18 (Activation)	(None, 16, 16, 36)	0	batch normalization 18[0][0]

conv2d_19 (Conv2D)	(None, 16, 16, 6)	1944	activation_18[0][0]
concatenate_17 (Concatenate)	(None, 16, 16, 42)	0	concatenate_16[0][0] conv2d_19[0][0]
batch_normalization_19 (BatchNo	(None, 16, 16, 42)	168	concatenate_17[0][0]
activation_19 (Activation)	(None, 16, 16, 42)	0	batch_normalization_19[0][0]
conv2d_20 (Conv2D)	(None, 16, 16, 6)	2268	activation_19[0][0]
concatenate_18 (Concatenate)	(None, 16, 16, 48)	0	concatenate_17[0][0] conv2d_20[0][0]
batch_normalization_20 (BatchNo	(None, 16, 16, 48)	192	concatenate_18[0][0]
activation_20 (Activation)	(None, 16, 16, 48)	0	batch_normalization_20[0][0]
conv2d_21 (Conv2D)	(None, 16, 16, 6)	2592	activation_20[0][0]
concatenate_19 (Concatenate)	(None, 16, 16, 54)	0	concatenate_18[0][0] conv2d_21[0][0]
batch_normalization_21 (BatchNo	(None, 16, 16, 54)	216	concatenate_19[0][0]
activation_21 (Activation)	(None, 16, 16, 54)	0	batch_normalization_21[0][0]
conv2d_22 (Conv2D)	(None, 16, 16, 6)	2916	activation_21[0][0]
concatenate_20 (Concatenate)	(None, 16, 16, 60)	0	concatenate_19[0][0] conv2d_22[0][0]
batch_normalization_22 (BatchNo	(None, 16, 16, 60)	240	concatenate_20[0][0]
activation_22 (Activation)	(None, 16, 16, 60)	0	batch_normalization_22[0][0]
conv2d_23 (Conv2D)	(None, 16, 16, 6)	3240	activation_22[0][0]
concatenate_21 (Concatenate)	(None, 16, 16, 66)	0	concatenate_20[0][0] conv2d_23[0][0]
batch_normalization_23 (BatchNo	(None, 16, 16, 66)	264	concatenate_21[0][0]

Layer	Output Shape	Param #	Connected to
activation_23 (Activation)	(None, 16, 16, 66)	0	batch_normalization_23[0][0]
conv2d_24 (Conv2D)	(None, 16, 16, 6)	3564	activation_23[0][0]
concatenate_22 (Concatenate)	(None, 16, 16, 72)	0	concatenate_21[0][0] conv2d_24[0][0]
batch_normalization_24 (BatchNo	(None, 16, 16, 72)	288	concatenate_22[0][0]
activation_24 (Activation)	(None, 16, 16, 72)	0	batch_normalization_24[0][0]
conv2d_25 (Conv2D)	(None, 16, 16, 6)	3888	activation_24[0][0]
concatenate_23 (Concatenate)	(None, 16, 16, 78)	0	concatenate_22[0][0] conv2d_25[0][0]
batch_normalization_25 (BatchNo	(None, 16, 16, 78)	312	concatenate_23[0][0]
activation_25 (Activation)	(None, 16, 16, 78)	0	batch_normalization_25[0][0]
conv2d_26 (Conv2D)	(None, 16, 16, 6)	468	activation_25[0][0]
average_pooling2d_1 (AveragePoo	(None, 8, 8, 6)	0	conv2d_26[0][0]
batch_normalization_26 (BatchNo	(None, 8, 8, 6)	24	average_pooling2d_1[0][0]
activation_26 (Activation)	(None, 8, 8, 6)	0	batch_normalization_26[0][0]
conv2d_27 (Conv2D)	(None, 8, 8, 6)	324	activation_26[0][0]
concatenate_24 (Concatenate)	(None, 8, 8, 12)	0	average_pooling2d_1[0][0] conv2d_27[0][0]
batch_normalization_27 (BatchNo	(None, 8, 8, 12)	48	concatenate_24[0][0]
activation_27 (Activation)	(None, 8, 8, 12)	0	batch_normalization_27[0][0]
conv2d_28 (Conv2D)	(None, 8, 8, 6)	648	activation_27[0][0]
concatenate_25 (Concatenate)	(None, 8, 8, 18)	0	concatenate_24[0][0] conv2d_28[0][0]

batch_normalization_28 (BatchNo	(None, 8, 8, 18)	72	concatenate_25[0][0]
activation_28 (Activation)	(None, 8, 8, 18)	0	batch_normalization_28[0][0]
conv2d_29 (Conv2D)	(None, 8, 8, 6)	972	activation_28[0][0]
concatenate_26 (Concatenate)	(None, 8, 8, 24)	0	concatenate_25[0][0] conv2d_29[0][0]
batch_normalization_29 (BatchNo	(None, 8, 8, 24)	96	concatenate_26[0][0]
activation_29 (Activation)	(None, 8, 8, 24)	0	batch_normalization_29[0][0]
conv2d_30 (Conv2D)	(None, 8, 8, 6)	1296	activation_29[0][0]
concatenate_27 (Concatenate)	(None, 8, 8, 30)	0	concatenate_26[0][0] conv2d_30[0][0]
batch_normalization_30 (BatchNo	(None, 8, 8, 30)	120	concatenate_27[0][0]
activation_30 (Activation)	(None, 8, 8, 30)	0	batch_normalization_30[0][0]
conv2d_31 (Conv2D)	(None, 8, 8, 6)	1620	activation_30[0][0]
concatenate_28 (Concatenate)	(None, 8, 8, 36)	0	concatenate_27[0][0] conv2d_31[0][0]
batch_normalization_31 (BatchNo	(None, 8, 8, 36)	144	concatenate_28[0][0]
activation_31 (Activation)	(None, 8, 8, 36)	0	batch_normalization_31[0][0]
conv2d_32 (Conv2D)	(None, 8, 8, 6)	1944	activation_31[0][0]
concatenate_29 (Concatenate)	(None, 8, 8, 42)	0	concatenate_28[0][0] conv2d_32[0][0]
batch_normalization_32 (BatchNo	(None, 8, 8, 42)	168	concatenate_29[0][0]
activation_32 (Activation)	(None, 8, 8, 42)	0	batch_normalization_32[0][0]
conv2d_33 (Conv2D)	(None, 8, 8, 6)	2268	activation_32[0][0]
concatenate_30 (Concatenate)	(None, 8, 8, 48)	0	concatenate_29[0][0]

			conv2d_33[0][0]
batch_normalization_33 (BatchNo	(None, 8, 8, 48)	192	concatenate_30[0][0]
activation_33 (Activation)	(None, 8, 8, 48)	0	batch_normalization_33[0][0]
conv2d_34 (Conv2D)	(None, 8, 8, 6)	2592	activation_33[0][0]
concatenate_31 (Concatenate)	(None, 8, 8, 54)	0	concatenate_30[0][0] conv2d_34[0][0]
batch_normalization_34 (BatchNo	(None, 8, 8, 54)	216	concatenate_31[0][0]
activation_34 (Activation)	(None, 8, 8, 54)	0	batch_normalization_34[0][0]
conv2d_35 (Conv2D)	(None, 8, 8, 6)	2916	activation_34[0][0]
concatenate_32 (Concatenate)	(None, 8, 8, 60)	0	concatenate_31[0][0] conv2d_35[0][0]
batch_normalization_35 (BatchNo	(None, 8, 8, 60)	240	concatenate_32[0][0]
activation_35 (Activation)	(None, 8, 8, 60)	0	batch_normalization_35[0][0]
conv2d_36 (Conv2D)	(None, 8, 8, 6)	3240	activation_35[0][0]
concatenate_33 (Concatenate)	(None, 8, 8, 66)	0	concatenate_32[0][0] conv2d_36[0][0]
batch_normalization_36 (BatchNo	(None, 8, 8, 66)	264	concatenate_33[0][0]
activation_36 (Activation)	(None, 8, 8, 66)	0	batch_normalization_36[0][0]
conv2d_37 (Conv2D)	(None, 8, 8, 6)	3564	activation_36[0][0]
concatenate_34 (Concatenate)	(None, 8, 8, 72)	0	concatenate_33[0][0] conv2d_37[0][0]
batch_normalization_37 (BatchNo	(None, 8, 8, 72)	288	concatenate_34[0][0]
activation_37 (Activation)	(None, 8, 8, 72)	0	batch_normalization_37[0][0]
conv2d_38 (Conv2D)	(None, 8, 8, 6)	3888	activation_37[0][0]

Layer	Output Shape	Param #	Connected to
concatenate_35 (Concatenate)	(None, 8, 8, 78)	0	concatenate_34[0][0] conv2d_38[0][0]
batch_normalization_38 (BatchNo	(None, 8, 8, 78)	312	concatenate_35[0][0]
activation_38 (Activation)	(None, 8, 8, 78)	0	batch_normalization_38[0][0]
conv2d_39 (Conv2D)	(None, 8, 8, 6)	468	activation_38[0][0]
average_pooling2d_2 (AveragePoo	(None, 4, 4, 6)	0	conv2d_39[0][0]
batch_normalization_39 (BatchNo	(None, 4, 4, 6)	24	average_pooling2d_2[0][0]
activation_39 (Activation)	(None, 4, 4, 6)	0	batch_normalization_39[0][0]
conv2d_40 (Conv2D)	(None, 4, 4, 6)	324	activation_39[0][0]
concatenate_36 (Concatenate)	(None, 4, 4, 12)	0	average_pooling2d_2[0][0] conv2d_40[0][0]
batch_normalization_40 (BatchNo	(None, 4, 4, 12)	48	concatenate_36[0][0]
activation_40 (Activation)	(None, 4, 4, 12)	0	batch_normalization_40[0][0]
conv2d_41 (Conv2D)	(None, 4, 4, 6)	648	activation_40[0][0]
concatenate_37 (Concatenate)	(None, 4, 4, 18)	0	concatenate_36[0][0] conv2d_41[0][0]
batch_normalization_41 (BatchNo	(None, 4, 4, 18)	72	concatenate_37[0][0]
activation_41 (Activation)	(None, 4, 4, 18)	0	batch_normalization_41[0][0]
conv2d_42 (Conv2D)	(None, 4, 4, 6)	972	activation_41[0][0]
concatenate_38 (Concatenate)	(None, 4, 4, 24)	0	concatenate_37[0][0] conv2d_42[0][0]
batch_normalization_42 (BatchNo	(None, 4, 4, 24)	96	concatenate_38[0][0]
activation_42 (Activation)	(None, 4, 4, 24)	0	batch_normalization_42[0][0]

conv2d_43 (Conv2D)	(None, 4, 4, 6)	1296	activation_42[0][0]
concatenate_39 (Concatenate)	(None, 4, 4, 30)	0	concatenate_38[0][0] conv2d_43[0][0]
batch_normalization_43 (BatchNo	(None, 4, 4, 30)	120	concatenate_39[0][0]
activation_43 (Activation)	(None, 4, 4, 30)	0	batch_normalization_43[0][0]
conv2d_44 (Conv2D)	(None, 4, 4, 6)	1620	activation_43[0][0]
concatenate_40 (Concatenate)	(None, 4, 4, 36)	0	concatenate_39[0][0] conv2d_44[0][0]
batch_normalization_44 (BatchNo	(None, 4, 4, 36)	144	concatenate_40[0][0]
activation_44 (Activation)	(None, 4, 4, 36)	0	batch_normalization_44[0][0]
conv2d_45 (Conv2D)	(None, 4, 4, 6)	1944	activation_44[0][0]
concatenate_41 (Concatenate)	(None, 4, 4, 42)	0	concatenate_40[0][0] conv2d_45[0][0]
batch_normalization_45 (BatchNo	(None, 4, 4, 42)	168	concatenate_41[0][0]
activation_45 (Activation)	(None, 4, 4, 42)	0	batch_normalization_45[0][0]
conv2d_46 (Conv2D)	(None, 4, 4, 6)	2268	activation_45[0][0]
concatenate_42 (Concatenate)	(None, 4, 4, 48)	0	concatenate_41[0][0] conv2d_46[0][0]
batch_normalization_46 (BatchNo	(None, 4, 4, 48)	192	concatenate_42[0][0]
activation_46 (Activation)	(None, 4, 4, 48)	0	batch_normalization_46[0][0]
conv2d_47 (Conv2D)	(None, 4, 4, 6)	2592	activation_46[0][0]
concatenate_43 (Concatenate)	(None, 4, 4, 54)	0	concatenate_42[0][0] conv2d_47[0][0]
batch_normalization_47 (BatchNo	(None, 4, 4, 54)	216	concatenate_43[0][0]

activation_47 (Activation)	(None, 4, 4, 54)	0	batch_normalization_47[0][0]
conv2d_48 (Conv2D)	(None, 4, 4, 6)	2916	activation_47[0][0]
concatenate_44 (Concatenate)	(None, 4, 4, 60)	0	concatenate_43[0][0] conv2d_48[0][0]
batch_normalization_48 (BatchNo	(None, 4, 4, 60)	240	concatenate_44[0][0]
activation_48 (Activation)	(None, 4, 4, 60)	0	batch_normalization_48[0][0]
conv2d_49 (Conv2D)	(None, 4, 4, 6)	3240	activation_48[0][0]
concatenate_45 (Concatenate)	(None, 4, 4, 66)	0	concatenate_44[0][0] conv2d_49[0][0]
batch_normalization_49 (BatchNo	(None, 4, 4, 66)	264	concatenate_45[0][0]
activation_49 (Activation)	(None, 4, 4, 66)	0	batch_normalization_49[0][0]
conv2d_50 (Conv2D)	(None, 4, 4, 6)	3564	activation_49[0][0]
concatenate_46 (Concatenate)	(None, 4, 4, 72)	0	concatenate_45[0][0] conv2d_50[0][0]
batch_normalization_50 (BatchNo	(None, 4, 4, 72)	288	concatenate_46[0][0]
activation_50 (Activation)	(None, 4, 4, 72)	0	batch_normalization_50[0][0]
conv2d_51 (Conv2D)	(None, 4, 4, 6)	3888	activation_50[0][0]
concatenate_47 (Concatenate)	(None, 4, 4, 78)	0	concatenate_46[0][0] conv2d_51[0][0]
batch_normalization_51 (BatchNo	(None, 4, 4, 78)	312	concatenate_47[0][0]
activation_51 (Activation)	(None, 4, 4, 78)	0	batch_normalization_51[0][0]
average_pooling2d_3 (AveragePoo	(None, 2, 2, 78)	0	activation_51[0][0]
conv2d_52 (Conv2D)	(None, 2, 2, 10)	3120	average_pooling2d_3[0][0]
global_max_pooling2d (GlobalMax	(None, 10)	0	conv2d_52[0][0]

```
                        _____
      activation_52 (Activation)        (None, 10)          0          global_max_pooling2d[0][0]
      ================================================================================
      Total params: 118,908
      Trainable params: 114,384
```

```python
#https://www.tensorflow.org/tensorboard/scalars_and_keras
filepath="weights.best.hdf5"
from keras.callbacks import ModelCheckpoint, EarlyStopping, TensorBoard
callback_1 = tf.keras.callbacks.ModelCheckpoint(filepath=filepath ,
                              save_weights_only=True,
                              monitor="val_accuracy",
                              mode="max",
                              save_best_only=True,
                              verbose=1)
tensorboard_1 = TensorBoard(log_dir='graph_one', batch_size=16,update_freq='epoch')
```

```
 ⤷   /usr/local/lib/python3.6/dist-packages/keras/callbacks/tensorboard_v2.py:92: UserWarning: The TensorBoard callback `batch_size` ar
        warnings.warn('The TensorBoard callback `batch_size` argument '
```

```python
# determine Loss function and Optimizer
model.compile(loss='categorical_crossentropy',
            optimizer=Adam(),
            metrics=['accuracy'])
```

```python
history = model.fit_generator(img_train,
                    steps_per_epoch=steps,
                    epochs=epochs,
                    verbose=1,
                    validation_data=(X_test, y_test),
                    callbacks=callback_1)
```

 ⤷

```
WARNING:tensorflow:From <ipython-input-15-16fbaf398460>:6: Model.fit_generator (from tensorflow.python.keras.engine.training) is d
Instructions for updating:
Please use Model.fit, which supports generators.
Epoch 1/50
781/781 [==============================] - ETA: 0s - loss: 1.7687 - accuracy: 0.3304
Epoch 00001: val_accuracy improved from -inf to 0.39300, saving model to weights.best.hdf5
781/781 [==============================] - 125s 159ms/step - loss: 1.7687 - accuracy: 0.3304 - val_loss: 1.7819 - val_accuracy: 0.
Epoch 2/50
781/781 [==============================] - ETA: 0s - loss: 1.4507 - accuracy: 0.4648
Epoch 00002: val_accuracy improved from 0.39300 to 0.46300, saving model to weights.best.hdf5
781/781 [==============================] - 120s 154ms/step - loss: 1.4507 - accuracy: 0.4648 - val_loss: 1.5547 - val_accuracy: 0.
Epoch 3/50
781/781 [==============================] - ETA: 0s - loss: 1.3097 - accuracy: 0.5259
Epoch 00003: val_accuracy improved from 0.46300 to 0.50360, saving model to weights.best.hdf5
781/781 [==============================] - 120s 154ms/step - loss: 1.3097 - accuracy: 0.5259 - val_loss: 1.4587 - val_accuracy: 0.
Epoch 4/50
781/781 [==============================] - ETA: 0s - loss: 1.2128 - accuracy: 0.5623
Epoch 00004: val_accuracy improved from 0.50360 to 0.57770, saving model to weights.best.hdf5
781/781 [==============================] - 120s 154ms/step - loss: 1.2128 - accuracy: 0.5623 - val_loss: 1.2735 - val_accuracy: 0.
Epoch 5/50
781/781 [==============================] - ETA: 0s - loss: 1.1434 - accuracy: 0.5911
Epoch 00005: val_accuracy improved from 0.57770 to 0.58450, saving model to weights.best.hdf5
781/781 [==============================] - 120s 154ms/step - loss: 1.1434 - accuracy: 0.5911 - val_loss: 1.2057 - val_accuracy: 0.
Epoch 6/50
781/781 [==============================] - ETA: 0s - loss: 1.0712 - accuracy: 0.6198
Epoch 00006: val_accuracy improved from 0.58450 to 0.60990, saving model to weights.best.hdf5
781/781 [==============================] - 120s 154ms/step - loss: 1.0712 - accuracy: 0.6198 - val_loss: 1.1694 - val_accuracy: 0.
Epoch 7/50
781/781 [==============================] - ETA: 0s - loss: 1.0281 - accuracy: 0.6360
Epoch 00007: val_accuracy did not improve from 0.60990
781/781 [==============================] - 120s 154ms/step - loss: 1.0281 - accuracy: 0.6360 - val_loss: 1.3655 - val_accuracy: 0.
Epoch 8/50
781/781 [==============================] - ETA: 0s - loss: 0.9834 - accuracy: 0.6492
Epoch 00008: val_accuracy did not improve from 0.60990
781/781 [==============================] - 120s 154ms/step - loss: 0.9834 - accuracy: 0.6492 - val_loss: 1.3505 - val_accuracy: 0.
Epoch 9/50
781/781 [==============================] - ETA: 0s - loss: 0.9516 - accuracy: 0.6623
Epoch 00009: val_accuracy improved from 0.60990 to 0.67270, saving model to weights.best.hdf5
781/781 [==============================] - 120s 154ms/step - loss: 0.9516 - accuracy: 0.6623 - val_loss: 0.9450 - val_accuracy: 0.
Epoch 10/50
781/781 [==============================] - ETA: 0s - loss: 0.9174 - accuracy: 0.6747
Epoch 00010: val_accuracy did not improve from 0.67270
781/781 [==============================] - 120s 153ms/step - loss: 0.9174 - accuracy: 0.6747 - val_loss: 1.0983 - val_accuracy: 0.
```

```
781/781 [                            ] - 120s 153ms/step - loss: 0.9174 - accuracy: 0.6747 - val_loss: 1.0909 - val_accuracy: 0.
Epoch 11/50
781/781 [==============================] - ETA: 0s - loss: 0.8937 - accuracy: 0.6844
Epoch 00011: val_accuracy improved from 0.67270 to 0.68550, saving model to weights.best.hdf5
781/781 [==============================] - 120s 154ms/step - loss: 0.8937 - accuracy: 0.6844 - val_loss: 0.8986 - val_accuracy: 0.
Epoch 12/50
781/781 [==============================] - ETA: 0s - loss: 0.8688 - accuracy: 0.6935
Epoch 00012: val_accuracy did not improve from 0.68550
781/781 [==============================] - 120s 154ms/step - loss: 0.8688 - accuracy: 0.6935 - val_loss: 1.0878 - val_accuracy: 0.
Epoch 13/50
781/781 [==============================] - ETA: 0s - loss: 0.8494 - accuracy: 0.6990
Epoch 00013: val_accuracy did not improve from 0.68550
781/781 [==============================] - 120s 153ms/step - loss: 0.8494 - accuracy: 0.6990 - val_loss: 1.2085 - val_accuracy: 0.
Epoch 14/50
781/781 [==============================] - ETA: 0s - loss: 0.8325 - accuracy: 0.7074
Epoch 00014: val_accuracy improved from 0.68550 to 0.70740, saving model to weights.best.hdf5
781/781 [==============================] - 120s 154ms/step - loss: 0.8325 - accuracy: 0.7074 - val_loss: 0.8635 - val_accuracy: 0.
Epoch 15/50
781/781 [==============================] - ETA: 0s - loss: 0.8114 - accuracy: 0.7165
Epoch 00015: val_accuracy did not improve from 0.70740
781/781 [==============================] - 120s 153ms/step - loss: 0.8114 - accuracy: 0.7165 - val_loss: 1.0147 - val_accuracy: 0.
Epoch 16/50
781/781 [==============================] - ETA: 0s - loss: 0.7975 - accuracy: 0.7223
Epoch 00016: val_accuracy did not improve from 0.70740
781/781 [==============================] - 120s 153ms/step - loss: 0.7975 - accuracy: 0.7223 - val_loss: 1.0108 - val_accuracy: 0.
Epoch 17/50
781/781 [==============================] - ETA: 0s - loss: 0.7831 - accuracy: 0.7253
Epoch 00017: val_accuracy did not improve from 0.70740
781/781 [==============================] - 120s 153ms/step - loss: 0.7831 - accuracy: 0.7253 - val_loss: 0.9450 - val_accuracy: 0.
Epoch 18/50
781/781 [==============================] - ETA: 0s - loss: 0.7762 - accuracy: 0.7289
Epoch 00018: val_accuracy did not improve from 0.70740
781/781 [==============================] - 120s 153ms/step - loss: 0.7762 - accuracy: 0.7289 - val_loss: 0.9348 - val_accuracy: 0.
Epoch 19/50
781/781 [==============================] - ETA: 0s - loss: 0.7558 - accuracy: 0.7342
Epoch 00019: val_accuracy improved from 0.70740 to 0.71010, saving model to weights.best.hdf5
781/781 [==============================] - 120s 154ms/step - loss: 0.7558 - accuracy: 0.7342 - val_loss: 0.9070 - val_accuracy: 0.
Epoch 20/50
781/781 [==============================] - ETA: 0s - loss: 0.7450 - accuracy: 0.7411
Epoch 00020: val_accuracy improved from 0.71010 to 0.73860, saving model to weights.best.hdf5
781/781 [==============================] - 120s 154ms/step - loss: 0.7450 - accuracy: 0.7411 - val_loss: 0.7730 - val_accuracy: 0.
Epoch 21/50
781/781 [==============================] - ETA: 0s - loss: 0.7367 - accuracy: 0.7417
Epoch 00021: val_accuracy did not improve from 0.73860
```

```
                    Epoch 00021: val_accuracy did not improve from 0.73860
781/781 [==============================] - 120s 154ms/step - loss: 0.7367 - accuracy: 0.7417 - val_loss: 0.8017 - val_accuracy: 0.
Epoch 22/50
781/781 [==============================] - ETA: 0s - loss: 0.7238 - accuracy: 0.7476
Epoch 00022: val_accuracy did not improve from 0.73860
781/781 [==============================] - 120s 154ms/step - loss: 0.7238 - accuracy: 0.7476 - val_loss: 0.9311 - val_accuracy: 0.
Epoch 23/50
781/781 [==============================] - ETA: 0s - loss: 0.7121 - accuracy: 0.7505
Epoch 00023: val_accuracy did not improve from 0.73860
781/781 [==============================] - 120s 154ms/step - loss: 0.7121 - accuracy: 0.7505 - val_loss: 1.0110 - val_accuracy: 0.
Epoch 24/50
781/781 [==============================] - ETA: 0s - loss: 0.7026 - accuracy: 0.7547
Epoch 00024: val_accuracy improved from 0.73860 to 0.75750, saving model to weights.best.hdf5
781/781 [==============================] - 120s 154ms/step - loss: 0.7026 - accuracy: 0.7547 - val_loss: 0.7006 - val_accuracy: 0.
Epoch 25/50
781/781 [==============================] - ETA: 0s - loss: 0.6937 - accuracy: 0.7577
Epoch 00025: val_accuracy improved from 0.75750 to 0.76050, saving model to weights.best.hdf5
781/781 [==============================] - 121s 154ms/step - loss: 0.6937 - accuracy: 0.7577 - val_loss: 0.7010 - val_accuracy: 0.
Epoch 26/50
781/781 [==============================] - ETA: 0s - loss: 0.6853 - accuracy: 0.7614
Epoch 00026: val_accuracy did not improve from 0.76050
781/781 [==============================] - 121s 154ms/step - loss: 0.6853 - accuracy: 0.7614 - val_loss: 0.9216 - val_accuracy: 0.
Epoch 27/50
781/781 [==============================] - ETA: 0s - loss: 0.6789 - accuracy: 0.7607
Epoch 00027: val_accuracy did not improve from 0.76050
781/781 [==============================] - 120s 154ms/step - loss: 0.6789 - accuracy: 0.7607 - val_loss: 0.7221 - val_accuracy: 0.
Epoch 28/50
781/781 [==============================] - ETA: 0s - loss: 0.6712 - accuracy: 0.7656
Epoch 00028: val_accuracy did not improve from 0.76050
781/781 [==============================] - 120s 154ms/step - loss: 0.6712 - accuracy: 0.7656 - val_loss: 0.9455 - val_accuracy: 0.
Epoch 29/50
781/781 [==============================] - ETA: 0s - loss: 0.6706 - accuracy: 0.7653
Epoch 00029: val_accuracy did not improve from 0.76050
781/781 [==============================] - 120s 154ms/step - loss: 0.6706 - accuracy: 0.7653 - val_loss: 0.9505 - val_accuracy: 0.
Epoch 30/50
781/781 [==============================] - ETA: 0s - loss: 0.6571 - accuracy: 0.7695
Epoch 00030: val_accuracy did not improve from 0.76050
781/781 [==============================] - 120s 154ms/step - loss: 0.6571 - accuracy: 0.7695 - val_loss: 1.1000 - val_accuracy: 0.
Epoch 31/50
781/781 [==============================] - ETA: 0s - loss: 0.6565 - accuracy: 0.7732
Epoch 00031: val_accuracy did not improve from 0.76050
781/781 [==============================] - 120s 154ms/step - loss: 0.6565 - accuracy: 0.7732 - val_loss: 0.8448 - val_accuracy: 0.
Epoch 32/50
781/781 [==============================] - ETA: 0s - loss: 0.6531 - accuracy: 0.7718
```

```
Epoch 00032: val_accuracy did not improve from 0.76050
781/781 [==============================] - 120s 153ms/step - loss: 0.6531 - accuracy: 0.7718 - val_loss: 0.7974 - val_accuracy: 0.
Epoch 33/50
781/781 [==============================] - ETA: 0s - loss: 0.6390 - accuracy: 0.7760
Epoch 00033: val_accuracy did not improve from 0.76050
781/781 [==============================] - 118s 151ms/step - loss: 0.6390 - accuracy: 0.7760 - val_loss: 0.7383 - val_accuracy: 0.
Epoch 34/50
781/781 [==============================] - ETA: 0s - loss: 0.6350 - accuracy: 0.7786
Epoch 00034: val_accuracy improved from 0.76050 to 0.78300, saving model to weights.best.hdf5
781/781 [==============================] - 119s 152ms/step - loss: 0.6350 - accuracy: 0.7786 - val_loss: 0.6438 - val_accuracy: 0.
Epoch 35/50
781/781 [==============================] - ETA: 0s - loss: 0.6247 - accuracy: 0.7838
Epoch 00035: val_accuracy did not improve from 0.78300
781/781 [==============================] - 119s 152ms/step - loss: 0.6247 - accuracy: 0.7838 - val_loss: 0.6933 - val_accuracy: 0.
Epoch 36/50
781/781 [==============================] - ETA: 0s - loss: 0.6293 - accuracy: 0.7811
Epoch 00036: val_accuracy did not improve from 0.78300
781/781 [==============================] - 119s 152ms/step - loss: 0.6293 - accuracy: 0.7811 - val_loss: 0.6620 - val_accuracy: 0.
Epoch 37/50
781/781 [==============================] - ETA: 0s - loss: 0.6242 - accuracy: 0.7838
Epoch 00037: val_accuracy did not improve from 0.78300
781/781 [==============================] - 119s 152ms/step - loss: 0.6242 - accuracy: 0.7838 - val_loss: 0.6896 - val_accuracy: 0.
Epoch 38/50
781/781 [==============================] - ETA: 0s - loss: 0.6141 - accuracy: 0.7849
Epoch 00038: val_accuracy improved from 0.78300 to 0.78740, saving model to weights.best.hdf5
781/781 [==============================] - 119s 152ms/step - loss: 0.6141 - accuracy: 0.7849 - val_loss: 0.6309 - val_accuracy: 0.
Epoch 39/50
781/781 [==============================] - ETA: 0s - loss: 0.6140 - accuracy: 0.7850
Epoch 00039: val_accuracy improved from 0.78740 to 0.79750, saving model to weights.best.hdf5
781/781 [==============================] - 119s 152ms/step - loss: 0.6140 - accuracy: 0.7850 - val_loss: 0.6160 - val_accuracy: 0.
Epoch 40/50
781/781 [==============================] - ETA: 0s - loss: 0.6030 - accuracy: 0.7910
Epoch 00040: val_accuracy did not improve from 0.79750
781/781 [==============================] - 119s 152ms/step - loss: 0.6030 - accuracy: 0.7910 - val_loss: 0.7697 - val_accuracy: 0.
Epoch 41/50
781/781 [==============================] - ETA: 0s - loss: 0.6012 - accuracy: 0.7906
Epoch 00041: val_accuracy did not improve from 0.79750
781/781 [==============================] - 119s 152ms/step - loss: 0.6012 - accuracy: 0.7906 - val_loss: 0.6306 - val_accuracy: 0.
Epoch 42/50
781/781 [==============================] - ETA: 0s - loss: 0.5954 - accuracy: 0.7912
Epoch 00042: val_accuracy did not improve from 0.79750
781/781 [==============================] - 119s 152ms/step - loss: 0.5954 - accuracy: 0.7912 - val_loss: 0.7234 - val_accuracy: 0.
Epoch 43/50
```

```
781/781 [==============================] - ETA: 0s - loss: 0.5905 - accuracy: 0.7924
Epoch 00043: val_accuracy did not improve from 0.79750
781/781 [==============================] - 119s 152ms/step - loss: 0.5905 - accuracy: 0.7924 - val_loss: 0.6394 - val_accuracy: 0.
Epoch 44/50
781/781 [==============================] - ETA: 0s - loss: 0.5897 - accuracy: 0.7942
Epoch 00044: val_accuracy did not improve from 0.79750
781/781 [==============================] - 118s 152ms/step - loss: 0.5897 - accuracy: 0.7942 - val_loss: 0.6723 - val_accuracy: 0.
Epoch 45/50
781/781 [==============================] - ETA: 0s - loss: 0.5840 - accuracy: 0.7959
Epoch 00045: val_accuracy did not improve from 0.79750
781/781 [==============================] - 119s 152ms/step - loss: 0.5840 - accuracy: 0.7959 - val_loss: 0.6378 - val_accuracy: 0.
Epoch 46/50
781/781 [==============================] - ETA: 0s - loss: 0.5808 - accuracy: 0.7961
Epoch 00046: val_accuracy did not improve from 0.79750
781/781 [==============================] - 119s 152ms/step - loss: 0.5808 - accuracy: 0.7961 - val_loss: 0.5950 - val_accuracy: 0.
Epoch 47/50
781/781 [==============================] - ETA: 0s - loss: 0.5750 - accuracy: 0.7986
Epoch 00047: val_accuracy did not improve from 0.79750
781/781 [==============================] - 119s 152ms/step - loss: 0.5750 - accuracy: 0.7986 - val_loss: 0.6920 - val_accuracy: 0.
Epoch 48/50
781/781 [==============================] - ETA: 0s - loss: 0.5744 - accuracy: 0.7991
Epoch 00048: val_accuracy did not improve from 0.79750
781/781 [==============================] - 118s 152ms/step - loss: 0.5744 - accuracy: 0.7991 - val_loss: 0.7180 - val_accuracy: 0.
Epoch 49/50
781/781 [==============================] - ETA: 0s - loss: 0.5751 - accuracy: 0.7980
Epoch 00049: val_accuracy improved from 0.79750 to 0.79800, saving model to weights.best.hdf5
781/781 [==============================] - 119s 152ms/step - loss: 0.5751 - accuracy: 0.7980 - val_loss: 0.6035 - val_accuracy: 0.
Epoch 50/50
781/781 [==============================] - ETA: 0s - loss: 0.5691 - accuracy: 0.8019
Epoch 00050: val_accuracy did not improve from 0.79800
781/781 [==============================] - 118s 152ms/step - loss: 0.5691 - accuracy: 0.8019 - val_loss: 0.8045 - val_accuracy: 0.
```

```
import matplotlib.pyplot as plt
```

```python
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.grid()
plt.title('model_accuracy')
plt.xlabel('Epochs')
plt.ylabel('accuracy')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```
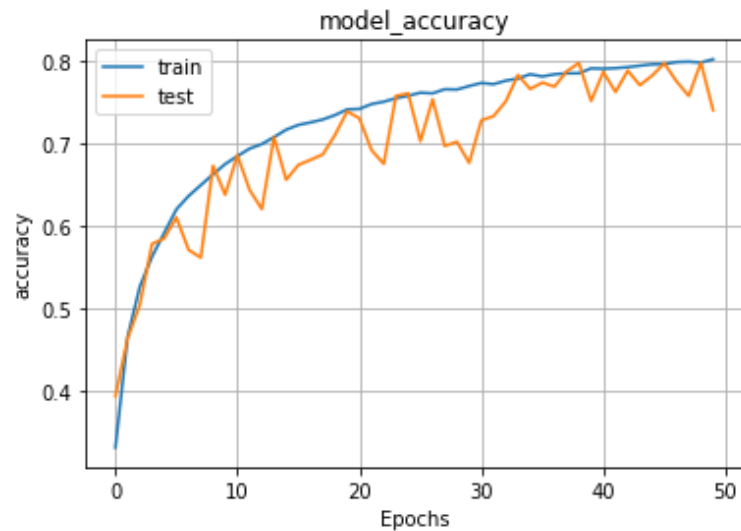


```python
# Save the trained weights in to .h5 format
model.save_weights("DNST_model.h5")
print("Saved model to disk")
```

Saved model to disk

```python
# Test the model
score = model.evaluate(X_test, y_test, verbose=1)
print('Test loss:', score[0])
print('Test accuracy:', score[1])
```

```
313/313 [==============================] - 8s 26ms/step - loss: 0.8045 - accuracy: 0.7401
Test loss: 0.8045022487640381
```

Model 2

```python
#https://www.pyimagesearch.com/2019/07/08/keras-imagedatagenerator-and-data-augmentation/
from keras.preprocessing.image import ImageDataGenerator

from keras.preprocessing.image import ImageDataGenerator
datagen = ImageDataGenerator(width_shift_range=0.3, height_shift_range=0.3, horizontal_flip=True,
datagen.fit(X_train)
it_train = datagen.flow(X_train, y_train, batch_size=64)
steps = int(X_train.shape[0] / 64)
```

```python
# Hyperparameters
batch_size = 128
num_classes = 10
epochs = 100
l = 40
num_filter = 24
compression = 0.5
compression_1 = 2
compression_2 = 4
dropout_rate = 0.2
```

```python
# Dense Block
def denseblock(input, num_filter = 12, dropout_rate = 0.2):
    global compression
    global compression_1
    global compression_2
    temp = input
    for _ in range(l):
        BatchNorm = layers.BatchNormalization()(temp)
        relu = layers.Activation('relu')(BatchNorm)
        Conv2D_3_3 = layers.Conv2D(int(num_filter*compression), (3,3), use_bias=False ,padding='s
        #if dropout_rate>0:
```

```python
        #Conv2D_3_3 = layers.Dropout(dropout_rate)(Conv2D_3_3)
        concat = layers.Concatenate(axis=-1)([temp,Conv2D_3_3])

        temp = concat


    return temp


## transition Block
def transition(input, num_filter = 12, dropout_rate = 0.2):
    global compression
    BatchNorm = layers.BatchNormalization()(input)
    relu = layers.Activation('relu')(BatchNorm)
    Conv2D_BottleNeck = layers.Conv2D(int(num_filter*compression), (1,1), use_bias=False ,padding
    #if dropout_rate>0:
        #Conv2D_BottleNeck = layers.Dropout(dropout_rate)(Conv2D_BottleNeck)
    avg = layers.AveragePooling2D(pool_size=(2,2))(Conv2D_BottleNeck)
    return avg


#output layer
def output_layer(input):
    global compression
    BatchNorm = layers.BatchNormalization()(input)
    relu = layers.Activation('relu')(BatchNorm)
    AvgPooling = layers.AveragePooling2D(pool_size=(2,2))(relu)
    #flat = layers.Flatten()(AvgPooling)
    First = layers.Conv2D(int(10), (2,2), use_bias=False ,padding='same')(AvgPooling)
    Maxpool = layers.GlobalMaxPooling2D()(First)
    output = layers.Activation("softmax")(Maxpool)
    return output



num_filter = 24
dropout_rate = 0.2
l = 12
input = layers.Input(shape=(img_height, img_width, channel,))
First_Conv2D = layers.Conv2D(num_filter, (3,3), use_bias=False ,padding='same')(input)

First_Block = denseblock(First_Conv2D, num_filter, dropout_rate)
First_Transition = transition(First_Block, num_filter, dropout_rate)
```

```
Second_Block = denseblock(First_Transition, num_filter, dropout_rate)
Second_Transition = transition(Second_Block, num_filter, dropout_rate)

Third_Block = denseblock(Second_Transition, num_filter, dropout_rate)
Third_Transition = transition(Third_Block, num_filter, dropout_rate)

Last_Block = denseblock(Third_Transition,  num_filter, dropout_rate)
output = output_layer(Last_Block)


model_2 = Model(inputs=[input], outputs=[output])
model_2.summary()
```

⟶

Model: "model"

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 32, 32, 3)]	0	
conv2d (Conv2D)	(None, 32, 32, 24)	648	input_1[0][0]
batch_normalization (BatchNorma	(None, 32, 32, 24)	96	conv2d[0][0]
activation (Activation)	(None, 32, 32, 24)	0	batch_normalization[0][0]
conv2d_1 (Conv2D)	(None, 32, 32, 12)	2592	activation[0][0]
concatenate (Concatenate)	(None, 32, 32, 36)	0	conv2d[0][0] conv2d_1[0][0]
batch_normalization_1 (BatchNor	(None, 32, 32, 36)	144	concatenate[0][0]
activation_1 (Activation)	(None, 32, 32, 36)	0	batch_normalization_1[0][0]
conv2d_2 (Conv2D)	(None, 32, 32, 12)	3888	activation_1[0][0]
concatenate_1 (Concatenate)	(None, 32, 32, 48)	0	concatenate[0][0] conv2d_2[0][0]
batch_normalization_2 (BatchNor	(None, 32, 32, 48)	192	concatenate_1[0][0]
activation_2 (Activation)	(None, 32, 32, 48)	0	batch_normalization_2[0][0]
conv2d_3 (Conv2D)	(None, 32, 32, 12)	5184	activation_2[0][0]
concatenate_2 (Concatenate)	(None, 32, 32, 60)	0	concatenate_1[0][0] conv2d_3[0][0]
batch_normalization_3 (BatchNor	(None, 32, 32, 60)	240	concatenate_2[0][0]
activation_3 (Activation)	(None, 32, 32, 60)	0	batch_normalization_3[0][0]
conv2d_4 (Conv2D)	(None, 32, 32, 12)	6480	activation_3[0][0]
concatenate_3 (Concatenate)	(None, 32, 32, 72)	0	concatenate_2[0][0] conv2d_4[0][0]

conv2d_4[0][0]

batch_normalization_4 (BatchNor	(None, 32, 32, 72)	288	concatenate_3[0][0]
activation_4 (Activation)	(None, 32, 32, 72)	0	batch_normalization_4[0][0]
conv2d_5 (Conv2D)	(None, 32, 32, 12)	7776	activation_4[0][0]
concatenate_4 (Concatenate)	(None, 32, 32, 84)	0	concatenate_3[0][0] conv2d_5[0][0]
batch_normalization_5 (BatchNor	(None, 32, 32, 84)	336	concatenate_4[0][0]
activation_5 (Activation)	(None, 32, 32, 84)	0	batch_normalization_5[0][0]
conv2d_6 (Conv2D)	(None, 32, 32, 12)	9072	activation_5[0][0]
concatenate_5 (Concatenate)	(None, 32, 32, 96)	0	concatenate_4[0][0] conv2d_6[0][0]
batch_normalization_6 (BatchNor	(None, 32, 32, 96)	384	concatenate_5[0][0]
activation_6 (Activation)	(None, 32, 32, 96)	0	batch_normalization_6[0][0]
conv2d_7 (Conv2D)	(None, 32, 32, 12)	10368	activation_6[0][0]
concatenate_6 (Concatenate)	(None, 32, 32, 108)	0	concatenate_5[0][0] conv2d_7[0][0]
batch_normalization_7 (BatchNor	(None, 32, 32, 108)	432	concatenate_6[0][0]
activation_7 (Activation)	(None, 32, 32, 108)	0	batch_normalization_7[0][0]
conv2d_8 (Conv2D)	(None, 32, 32, 12)	11664	activation_7[0][0]
concatenate_7 (Concatenate)	(None, 32, 32, 120)	0	concatenate_6[0][0] conv2d_8[0][0]
batch_normalization_8 (BatchNor	(None, 32, 32, 120)	480	concatenate_7[0][0]
activation_8 (Activation)	(None, 32, 32, 120)	0	batch_normalization_8[0][0]
conv2d_9 (Conv2D)	(None, 32, 32, 12)	12960	activation_8[0][0]

Layer (type)	Output Shape	Param #	Connected to
concatenate_8 (Concatenate)	(None, 32, 32, 132)	0	concatenate_7[0][0] conv2d_9[0][0]
batch_normalization_9 (BatchNor	(None, 32, 32, 132)	528	concatenate_8[0][0]
activation_9 (Activation)	(None, 32, 32, 132)	0	batch_normalization_9[0][0]
conv2d_10 (Conv2D)	(None, 32, 32, 12)	14256	activation_9[0][0]
concatenate_9 (Concatenate)	(None, 32, 32, 144)	0	concatenate_8[0][0] conv2d_10[0][0]
batch_normalization_10 (BatchNo	(None, 32, 32, 144)	576	concatenate_9[0][0]
activation_10 (Activation)	(None, 32, 32, 144)	0	batch_normalization_10[0][0]
conv2d_11 (Conv2D)	(None, 32, 32, 12)	15552	activation_10[0][0]
concatenate_10 (Concatenate)	(None, 32, 32, 156)	0	concatenate_9[0][0] conv2d_11[0][0]
batch_normalization_11 (BatchNo	(None, 32, 32, 156)	624	concatenate_10[0][0]
activation_11 (Activation)	(None, 32, 32, 156)	0	batch_normalization_11[0][0]
conv2d_12 (Conv2D)	(None, 32, 32, 12)	16848	activation_11[0][0]
concatenate_11 (Concatenate)	(None, 32, 32, 168)	0	concatenate_10[0][0] conv2d_12[0][0]
batch_normalization_12 (BatchNo	(None, 32, 32, 168)	672	concatenate_11[0][0]
activation_12 (Activation)	(None, 32, 32, 168)	0	batch_normalization_12[0][0]
conv2d_13 (Conv2D)	(None, 32, 32, 12)	2016	activation_12[0][0]
average_pooling2d (AveragePooli	(None, 16, 16, 12)	0	conv2d_13[0][0]
batch_normalization_13 (BatchNo	(None, 16, 16, 12)	48	average_pooling2d[0][0]
activation_13 (Activation)	(None, 16, 16, 12)	0	batch_normalization_13[0][0]
conv2d_14 (Conv2D)	(None, 16, 16, 12)	1296	activation_13[0][0]

concatenate_12 (Concatenate)	(None, 16, 16, 24)	0	average_pooling2d[0][0] conv2d_14[0][0]
batch_normalization_14 (BatchNo	(None, 16, 16, 24)	96	concatenate_12[0][0]
activation_14 (Activation)	(None, 16, 16, 24)	0	batch_normalization_14[0][0]
conv2d_15 (Conv2D)	(None, 16, 16, 12)	2592	activation_14[0][0]
concatenate_13 (Concatenate)	(None, 16, 16, 36)	0	concatenate_12[0][0] conv2d_15[0][0]
batch_normalization_15 (BatchNo	(None, 16, 16, 36)	144	concatenate_13[0][0]
activation_15 (Activation)	(None, 16, 16, 36)	0	batch_normalization_15[0][0]
conv2d_16 (Conv2D)	(None, 16, 16, 12)	3888	activation_15[0][0]
concatenate_14 (Concatenate)	(None, 16, 16, 48)	0	concatenate_13[0][0] conv2d_16[0][0]
batch_normalization_16 (BatchNo	(None, 16, 16, 48)	192	concatenate_14[0][0]
activation_16 (Activation)	(None, 16, 16, 48)	0	batch_normalization_16[0][0]
conv2d_17 (Conv2D)	(None, 16, 16, 12)	5184	activation_16[0][0]
concatenate_15 (Concatenate)	(None, 16, 16, 60)	0	concatenate_14[0][0] conv2d_17[0][0]
batch_normalization_17 (BatchNo	(None, 16, 16, 60)	240	concatenate_15[0][0]
activation_17 (Activation)	(None, 16, 16, 60)	0	batch_normalization_17[0][0]
conv2d_18 (Conv2D)	(None, 16, 16, 12)	6480	activation_17[0][0]
concatenate_16 (Concatenate)	(None, 16, 16, 72)	0	concatenate_15[0][0] conv2d_18[0][0]
batch_normalization_18 (BatchNo	(None, 16, 16, 72)	288	concatenate_16[0][0]
activation_18 (Activation)	(None, 16, 16, 72)	0	batch_normalization_18[0][0]

conv2d_19 (Conv2D)	(None, 16, 16, 12)	7776	activation_18[0][0]
concatenate_17 (Concatenate)	(None, 16, 16, 84)	0	concatenate_16[0][0]
			conv2d_19[0][0]
batch_normalization_19 (BatchNo	(None, 16, 16, 84)	336	concatenate_17[0][0]
activation_19 (Activation)	(None, 16, 16, 84)	0	batch_normalization_19[0][0]
conv2d_20 (Conv2D)	(None, 16, 16, 12)	9072	activation_19[0][0]
concatenate_18 (Concatenate)	(None, 16, 16, 96)	0	concatenate_17[0][0]
			conv2d_20[0][0]
batch_normalization_20 (BatchNo	(None, 16, 16, 96)	384	concatenate_18[0][0]
activation_20 (Activation)	(None, 16, 16, 96)	0	batch_normalization_20[0][0]
conv2d_21 (Conv2D)	(None, 16, 16, 12)	10368	activation_20[0][0]
concatenate_19 (Concatenate)	(None, 16, 16, 108)	0	concatenate_18[0][0]
			conv2d_21[0][0]
batch_normalization_21 (BatchNo	(None, 16, 16, 108)	432	concatenate_19[0][0]
activation_21 (Activation)	(None, 16, 16, 108)	0	batch_normalization_21[0][0]
conv2d_22 (Conv2D)	(None, 16, 16, 12)	11664	activation_21[0][0]
concatenate_20 (Concatenate)	(None, 16, 16, 120)	0	concatenate_19[0][0]
			conv2d_22[0][0]
batch_normalization_22 (BatchNo	(None, 16, 16, 120)	480	concatenate_20[0][0]
activation_22 (Activation)	(None, 16, 16, 120)	0	batch_normalization_22[0][0]
conv2d_23 (Conv2D)	(None, 16, 16, 12)	12960	activation_22[0][0]
concatenate_21 (Concatenate)	(None, 16, 16, 132)	0	concatenate_20[0][0]
			conv2d_23[0][0]
batch_normalization_23 (BatchNo	(None, 16, 16, 132)	528	concatenate_21[0][0]

activation_23 (Activation)	(None, 16, 16, 132)	0	batch_normalization_23[0][0]
conv2d_24 (Conv2D)	(None, 16, 16, 12)	14256	activation_23[0][0]
concatenate_22 (Concatenate)	(None, 16, 16, 144)	0	concatenate_21[0][0] conv2d_24[0][0]
batch_normalization_24 (BatchNo	(None, 16, 16, 144)	576	concatenate_22[0][0]
activation_24 (Activation)	(None, 16, 16, 144)	0	batch_normalization_24[0][0]
conv2d_25 (Conv2D)	(None, 16, 16, 12)	15552	activation_24[0][0]
concatenate_23 (Concatenate)	(None, 16, 16, 156)	0	concatenate_22[0][0] conv2d_25[0][0]
batch_normalization_25 (BatchNo	(None, 16, 16, 156)	624	concatenate_23[0][0]
activation_25 (Activation)	(None, 16, 16, 156)	0	batch_normalization_25[0][0]
conv2d_26 (Conv2D)	(None, 16, 16, 12)	1872	activation_25[0][0]
average_pooling2d_1 (AveragePoo	(None, 8, 8, 12)	0	conv2d_26[0][0]
batch_normalization_26 (BatchNo	(None, 8, 8, 12)	48	average_pooling2d_1[0][0]
activation_26 (Activation)	(None, 8, 8, 12)	0	batch_normalization_26[0][0]
conv2d_27 (Conv2D)	(None, 8, 8, 12)	1296	activation_26[0][0]
concatenate_24 (Concatenate)	(None, 8, 8, 24)	0	average_pooling2d_1[0][0] conv2d_27[0][0]
batch_normalization_27 (BatchNo	(None, 8, 8, 24)	96	concatenate_24[0][0]
activation_27 (Activation)	(None, 8, 8, 24)	0	batch_normalization_27[0][0]
conv2d_28 (Conv2D)	(None, 8, 8, 12)	2592	activation_27[0][0]
concatenate_25 (Concatenate)	(None, 8, 8, 36)	0	concatenate_24[0][0] conv2d_28[0][0]

batch_normalization_28 (BatchNo	(None, 8, 8, 36)	144	concatenate_25[0][0]
activation_28 (Activation)	(None, 8, 8, 36)	0	batch_normalization_28[0][0]
conv2d_29 (Conv2D)	(None, 8, 8, 12)	3888	activation_28[0][0]
concatenate_26 (Concatenate)	(None, 8, 8, 48)	0	concatenate_25[0][0] conv2d_29[0][0]
batch_normalization_29 (BatchNo	(None, 8, 8, 48)	192	concatenate_26[0][0]
activation_29 (Activation)	(None, 8, 8, 48)	0	batch_normalization_29[0][0]
conv2d_30 (Conv2D)	(None, 8, 8, 12)	5184	activation_29[0][0]
concatenate_27 (Concatenate)	(None, 8, 8, 60)	0	concatenate_26[0][0] conv2d_30[0][0]
batch_normalization_30 (BatchNo	(None, 8, 8, 60)	240	concatenate_27[0][0]
activation_30 (Activation)	(None, 8, 8, 60)	0	batch_normalization_30[0][0]
conv2d_31 (Conv2D)	(None, 8, 8, 12)	6480	activation_30[0][0]
concatenate_28 (Concatenate)	(None, 8, 8, 72)	0	concatenate_27[0][0] conv2d_31[0][0]
batch_normalization_31 (BatchNo	(None, 8, 8, 72)	288	concatenate_28[0][0]
activation_31 (Activation)	(None, 8, 8, 72)	0	batch_normalization_31[0][0]
conv2d_32 (Conv2D)	(None, 8, 8, 12)	7776	activation_31[0][0]
concatenate_29 (Concatenate)	(None, 8, 8, 84)	0	concatenate_28[0][0] conv2d_32[0][0]
batch_normalization_32 (BatchNo	(None, 8, 8, 84)	336	concatenate_29[0][0]
activation_32 (Activation)	(None, 8, 8, 84)	0	batch_normalization_32[0][0]
conv2d_33 (Conv2D)	(None, 8, 8, 12)	9072	activation_32[0][0]
concatenate_30 (Concatenate)	(None, 8, 8, 96)	0	concatenate_29[0][0]

			conv2d_33[0][0]
batch_normalization_33 (BatchNo	(None, 8, 8, 96)	384	concatenate_30[0][0]
activation_33 (Activation)	(None, 8, 8, 96)	0	batch_normalization_33[0][0]
conv2d_34 (Conv2D)	(None, 8, 8, 12)	10368	activation_33[0][0]
concatenate_31 (Concatenate)	(None, 8, 8, 108)	0	concatenate_30[0][0] conv2d_34[0][0]
batch_normalization_34 (BatchNo	(None, 8, 8, 108)	432	concatenate_31[0][0]
activation_34 (Activation)	(None, 8, 8, 108)	0	batch_normalization_34[0][0]
conv2d_35 (Conv2D)	(None, 8, 8, 12)	11664	activation_34[0][0]
concatenate_32 (Concatenate)	(None, 8, 8, 120)	0	concatenate_31[0][0] conv2d_35[0][0]
batch_normalization_35 (BatchNo	(None, 8, 8, 120)	480	concatenate_32[0][0]
activation_35 (Activation)	(None, 8, 8, 120)	0	batch_normalization_35[0][0]
conv2d_36 (Conv2D)	(None, 8, 8, 12)	12960	activation_35[0][0]
concatenate_33 (Concatenate)	(None, 8, 8, 132)	0	concatenate_32[0][0] conv2d_36[0][0]
batch_normalization_36 (BatchNo	(None, 8, 8, 132)	528	concatenate_33[0][0]
activation_36 (Activation)	(None, 8, 8, 132)	0	batch_normalization_36[0][0]
conv2d_37 (Conv2D)	(None, 8, 8, 12)	14256	activation_36[0][0]
concatenate_34 (Concatenate)	(None, 8, 8, 144)	0	concatenate_33[0][0] conv2d_37[0][0]
batch_normalization_37 (BatchNo	(None, 8, 8, 144)	576	concatenate_34[0][0]
activation_37 (Activation)	(None, 8, 8, 144)	0	batch_normalization_37[0][0]
conv2d_38 (Conv2D)	(None, 8, 8, 12)	15552	activation_37[0][0]

concatenate_35 (Concatenate)	(None, 8, 8, 156)	0	concatenate_34[0][0] conv2d_38[0][0]
batch_normalization_38 (BatchNo	(None, 8, 8, 156)	624	concatenate_35[0][0]
activation_38 (Activation)	(None, 8, 8, 156)	0	batch_normalization_38[0][0]
conv2d_39 (Conv2D)	(None, 8, 8, 12)	1872	activation_38[0][0]
average_pooling2d_2 (AveragePoo	(None, 4, 4, 12)	0	conv2d_39[0][0]
batch_normalization_39 (BatchNo	(None, 4, 4, 12)	48	average_pooling2d_2[0][0]
activation_39 (Activation)	(None, 4, 4, 12)	0	batch_normalization_39[0][0]
conv2d_40 (Conv2D)	(None, 4, 4, 12)	1296	activation_39[0][0]
concatenate_36 (Concatenate)	(None, 4, 4, 24)	0	average_pooling2d_2[0][0] conv2d_40[0][0]
batch_normalization_40 (BatchNo	(None, 4, 4, 24)	96	concatenate_36[0][0]
activation_40 (Activation)	(None, 4, 4, 24)	0	batch_normalization_40[0][0]
conv2d_41 (Conv2D)	(None, 4, 4, 12)	2592	activation_40[0][0]
concatenate_37 (Concatenate)	(None, 4, 4, 36)	0	concatenate_36[0][0] conv2d_41[0][0]
batch_normalization_41 (BatchNo	(None, 4, 4, 36)	144	concatenate_37[0][0]
activation_41 (Activation)	(None, 4, 4, 36)	0	batch_normalization_41[0][0]
conv2d_42 (Conv2D)	(None, 4, 4, 12)	3888	activation_41[0][0]
concatenate_38 (Concatenate)	(None, 4, 4, 48)	0	concatenate_37[0][0] conv2d_42[0][0]
batch_normalization_42 (BatchNo	(None, 4, 4, 48)	192	concatenate_38[0][0]
activation_42 (Activation)	(None, 4, 4, 48)	0	batch_normalization_42[0][0]

conv2d_43 (Conv2D)	(None, 4, 4, 12)	5184	activation_42[0][0]
concatenate_39 (Concatenate)	(None, 4, 4, 60)	0	concatenate_38[0][0] conv2d_43[0][0]
batch_normalization_43 (BatchNo	(None, 4, 4, 60)	240	concatenate_39[0][0]
activation_43 (Activation)	(None, 4, 4, 60)	0	batch_normalization_43[0][0]
conv2d_44 (Conv2D)	(None, 4, 4, 12)	6480	activation_43[0][0]
concatenate_40 (Concatenate)	(None, 4, 4, 72)	0	concatenate_39[0][0] conv2d_44[0][0]
batch_normalization_44 (BatchNo	(None, 4, 4, 72)	288	concatenate_40[0][0]
activation_44 (Activation)	(None, 4, 4, 72)	0	batch_normalization_44[0][0]
conv2d_45 (Conv2D)	(None, 4, 4, 12)	7776	activation_44[0][0]
concatenate_41 (Concatenate)	(None, 4, 4, 84)	0	concatenate_40[0][0] conv2d_45[0][0]
batch_normalization_45 (BatchNo	(None, 4, 4, 84)	336	concatenate_41[0][0]
activation_45 (Activation)	(None, 4, 4, 84)	0	batch_normalization_45[0][0]
conv2d_46 (Conv2D)	(None, 4, 4, 12)	9072	activation_45[0][0]
concatenate_42 (Concatenate)	(None, 4, 4, 96)	0	concatenate_41[0][0] conv2d_46[0][0]
batch_normalization_46 (BatchNo	(None, 4, 4, 96)	384	concatenate_42[0][0]
activation_46 (Activation)	(None, 4, 4, 96)	0	batch_normalization_46[0][0]
conv2d_47 (Conv2D)	(None, 4, 4, 12)	10368	activation_46[0][0]
concatenate_43 (Concatenate)	(None, 4, 4, 108)	0	concatenate_42[0][0] conv2d_47[0][0]
batch_normalization_47 (BatchNo	(None, 4, 4, 108)	432	concatenate_43[0][0]

activation_47 (Activation)	(None, 4, 4, 108)	0	batch_normalization_47[0][0]
conv2d_48 (Conv2D)	(None, 4, 4, 12)	11664	activation_47[0][0]
concatenate_44 (Concatenate)	(None, 4, 4, 120)	0	concatenate_43[0][0] conv2d_48[0][0]
batch_normalization_48 (BatchNo	(None, 4, 4, 120)	480	concatenate_44[0][0]
activation_48 (Activation)	(None, 4, 4, 120)	0	batch_normalization_48[0][0]
conv2d_49 (Conv2D)	(None, 4, 4, 12)	12960	activation_48[0][0]
concatenate_45 (Concatenate)	(None, 4, 4, 132)	0	concatenate_44[0][0] conv2d_49[0][0]
batch_normalization_49 (BatchNo	(None, 4, 4, 132)	528	concatenate_45[0][0]
activation_49 (Activation)	(None, 4, 4, 132)	0	batch_normalization_49[0][0]
conv2d_50 (Conv2D)	(None, 4, 4, 12)	14256	activation_49[0][0]
concatenate_46 (Concatenate)	(None, 4, 4, 144)	0	concatenate_45[0][0] conv2d_50[0][0]
batch_normalization_50 (BatchNo	(None, 4, 4, 144)	576	concatenate_46[0][0]
activation_50 (Activation)	(None, 4, 4, 144)	0	batch_normalization_50[0][0]
conv2d_51 (Conv2D)	(None, 4, 4, 12)	15552	activation_50[0][0]
concatenate_47 (Concatenate)	(None, 4, 4, 156)	0	concatenate_46[0][0] conv2d_51[0][0]
batch_normalization_51 (BatchNo	(None, 4, 4, 156)	624	concatenate_47[0][0]
activation_51 (Activation)	(None, 4, 4, 156)	0	batch_normalization_51[0][0]
average_pooling2d_3 (AveragePoo	(None, 2, 2, 156)	0	activation_51[0][0]
conv2d_52 (Conv2D)	(None, 2, 2, 10)	6240	average_pooling2d_3[0][0]
global_max_pooling2d (GlobalMax	(None, 10)	0	conv2d_52[0][0]

```
───────────────────────────────────────────────────────────────────────────────────
activation_52 (Activation)       (None, 10)          0          global_max_pooling2d[0][0]
===================================================================================
Total params: 450,648
Trainable params: 441,600
Non-trainable params: 9,048
───────────────────────────────────────────────────────────────────────────────────
```

```python
#https://www.tensorflow.org/tensorboard/scalars_and_keras
filepath="weights_3.best.hdf5"
from keras.callbacks import ModelCheckpoint, EarlyStopping, TensorBoard
callback_2 = tf.keras.callbacks.ModelCheckpoint(filepath=filepath ,
                          save_weights_only=True,
                          monitor="val_accuracy",
                          mode="max",
                          save_best_only=True,
                          verbose=1)
tensorboard_2 = TensorBoard(log_dir='graph_one', batch_size=64,update_freq='epoch')
```

```
/usr/local/lib/python3.6/dist-packages/keras/callbacks/tensorboard_v2.py:92: UserWarning: The TensorBoard callback `batch_size` ar
  warnings.warn('The TensorBoard callback `batch_size` argument '
```

```python
# determine Loss function and Optimizer
model_2.compile(loss='categorical_crossentropy',
             optimizer=Adam(),
             metrics=['accuracy'])
```

```python
history = model_2.fit_generator(img_train,
                  steps_per_epoch=steps,
                  epochs=epochs,
                  verbose=1,
```

```
                validation_data=(X_test, y_test),
                callbacks=callback_2)
```

```
WARNING:tensorflow:From <ipython-input-16-370768f647ea>:6: Model.fit_generator (from tensorflow.python.keras.engine.training) is d
Instructions for updating:
Please use Model.fit, which supports generators.
Epoch 1/100
781/781 [==============================] - ETA: 0s - loss: 1.7078 - accuracy: 0.3614
Epoch 00001: val_accuracy improved from -inf to 0.48190, saving model to weights_3.best.hdf5
781/781 [==============================] - 64s 82ms/step - loss: 1.7078 - accuracy: 0.3614 - val_loss: 1.4209 - val_accuracy: 0.48
Epoch 2/100
781/781 [==============================] - ETA: 0s - loss: 1.3442 - accuracy: 0.5120
Epoch 00002: val_accuracy did not improve from 0.48190
781/781 [==============================] - 60s 77ms/step - loss: 1.3442 - accuracy: 0.5120 - val_loss: 1.9366 - val_accuracy: 0.43
Epoch 3/100
781/781 [==============================] - ETA: 0s - loss: 1.1557 - accuracy: 0.5852
Epoch 00003: val_accuracy improved from 0.48190 to 0.62060, saving model to weights_3.best.hdf5
781/781 [==============================] - 61s 78ms/step - loss: 1.1557 - accuracy: 0.5852 - val_loss: 1.0709 - val_accuracy: 0.62
Epoch 4/100
781/781 [==============================] - ETA: 0s - loss: 1.0311 - accuracy: 0.6308
Epoch 00004: val_accuracy did not improve from 0.62060
781/781 [==============================] - 60s 76ms/step - loss: 1.0311 - accuracy: 0.6308 - val_loss: 1.1683 - val_accuracy: 0.60
Epoch 5/100
781/781 [==============================] - ETA: 0s - loss: 0.9492 - accuracy: 0.6617
Epoch 00005: val_accuracy improved from 0.62060 to 0.66110, saving model to weights_3.best.hdf5
781/781 [==============================] - 60s 77ms/step - loss: 0.9492 - accuracy: 0.6617 - val_loss: 1.0082 - val_accuracy: 0.66
Epoch 6/100
781/781 [==============================] - ETA: 0s - loss: 0.8879 - accuracy: 0.6873
Epoch 00006: val_accuracy did not improve from 0.66110
781/781 [==============================] - 60s 77ms/step - loss: 0.8879 - accuracy: 0.6873 - val_loss: 1.4715 - val_accuracy: 0.55
Epoch 7/100
781/781 [==============================] - ETA: 0s - loss: 0.8286 - accuracy: 0.7101
Epoch 00007: val_accuracy improved from 0.66110 to 0.68110, saving model to weights_3.best.hdf5
781/781 [==============================] - 60s 77ms/step - loss: 0.8286 - accuracy: 0.7101 - val_loss: 0.9707 - val_accuracy: 0.68
Epoch 8/100
781/781 [==============================] - ETA: 0s - loss: 0.7874 - accuracy: 0.7234
Epoch 00008: val_accuracy improved from 0.68110 to 0.74310, saving model to weights_3.best.hdf5
781/781 [==============================] - 59s 76ms/step - loss: 0.7874 - accuracy: 0.7234 - val_loss: 0.7654 - val_accuracy: 0.74
Epoch 9/100
781/781 [==============================] - ETA: 0s - loss: 0.7508 - accuracy: 0.7364
Epoch 00009: val_accuracy did not improve from 0.74310
781/781 [==============================] - 60s 77ms/step - loss: 0.7508 - accuracy: 0.7364 - val_loss: 0.9285 - val_accuracy: 0.69
Epoch 10/100
781/781 [==============================] - ETA: 0s - loss: 0.7149 - accuracy: 0.7495
Epoch 00010: val_accuracy did not improve from 0.74310
781/781 [==============================] - 60s 77ms/step - loss: 0.7149 - accuracy: 0.7495 - val_loss: 0.7695 - val_accuracy: 0.73
```

```
781/781 [                              ]  003 77ms/step - 1033  0.7119 - accuracy: 0.7199 - val_1033  0.7039 - val_accuracy: 0.73
Epoch 11/100
781/781 [==============================] - ETA: 0s - loss: 0.6894 - accuracy: 0.7604
Epoch 00011: val_accuracy did not improve from 0.74310
781/781 [==============================] - 58s 74ms/step - loss: 0.6894 - accuracy: 0.7604 - val_loss: 0.8247 - val_accuracy: 0.72
Epoch 12/100
781/781 [==============================] - ETA: 0s - loss: 0.6611 - accuracy: 0.7710
Epoch 00012: val_accuracy improved from 0.74310 to 0.78210, saving model to weights_3.best.hdf5
781/781 [==============================] - 59s 76ms/step - loss: 0.6611 - accuracy: 0.7710 - val_loss: 0.6519 - val_accuracy: 0.78
Epoch 13/100
781/781 [==============================] - ETA: 0s - loss: 0.6366 - accuracy: 0.7775
Epoch 00013: val_accuracy did not improve from 0.78210
781/781 [==============================] - 59s 76ms/step - loss: 0.6366 - accuracy: 0.7775 - val_loss: 0.7034 - val_accuracy: 0.76
Epoch 14/100
781/781 [==============================] - ETA: 0s - loss: 0.6210 - accuracy: 0.7850
Epoch 00014: val_accuracy did not improve from 0.78210
781/781 [==============================] - 59s 76ms/step - loss: 0.6210 - accuracy: 0.7850 - val_loss: 0.7885 - val_accuracy: 0.74
Epoch 15/100
781/781 [==============================] - ETA: 0s - loss: 0.6018 - accuracy: 0.7919
Epoch 00015: val_accuracy did not improve from 0.78210
781/781 [==============================] - 59s 76ms/step - loss: 0.6018 - accuracy: 0.7919 - val_loss: 0.6770 - val_accuracy: 0.77
Epoch 16/100
781/781 [==============================] - ETA: 0s - loss: 0.5843 - accuracy: 0.7977
Epoch 00016: val_accuracy did not improve from 0.78210
781/781 [==============================] - 59s 75ms/step - loss: 0.5843 - accuracy: 0.7977 - val_loss: 0.7066 - val_accuracy: 0.76
Epoch 17/100
781/781 [==============================] - ETA: 0s - loss: 0.5673 - accuracy: 0.8028
Epoch 00017: val_accuracy did not improve from 0.78210
781/781 [==============================] - 59s 75ms/step - loss: 0.5673 - accuracy: 0.8028 - val_loss: 0.7461 - val_accuracy: 0.76
Epoch 18/100
781/781 [==============================] - ETA: 0s - loss: 0.5511 - accuracy: 0.8087
Epoch 00018: val_accuracy improved from 0.78210 to 0.78360, saving model to weights_3.best.hdf5
781/781 [==============================] - 60s 76ms/step - loss: 0.5511 - accuracy: 0.8087 - val_loss: 0.6725 - val_accuracy: 0.78
Epoch 19/100
781/781 [==============================] - ETA: 0s - loss: 0.5400 - accuracy: 0.8120
Epoch 00019: val_accuracy did not improve from 0.78360
781/781 [==============================] - 59s 75ms/step - loss: 0.5400 - accuracy: 0.8120 - val_loss: 0.6973 - val_accuracy: 0.77
Epoch 20/100
781/781 [==============================] - ETA: 0s - loss: 0.5283 - accuracy: 0.8178
Epoch 00020: val_accuracy improved from 0.78360 to 0.82710, saving model to weights_3.best.hdf5
781/781 [==============================] - 59s 75ms/step - loss: 0.5283 - accuracy: 0.8178 - val_loss: 0.5188 - val_accuracy: 0.82
Epoch 21/100
781/781 [==============================] - ETA: 0s - loss: 0.5183 - accuracy: 0.8212
Epoch 00021: val accuracy did not improve from 0.82710
```

```
                  Epoch 00021: val_accuracy did not improve from 0.82710
781/781 [==============================] - 58s 75ms/step - loss: 0.5183 - accuracy: 0.8212 - val_loss: 0.5524 - val_accuracy: 0.81
Epoch 22/100
781/781 [==============================] - ETA: 0s - loss: 0.5082 - accuracy: 0.8248
Epoch 00022: val_accuracy did not improve from 0.82710
781/781 [==============================] - 59s 75ms/step - loss: 0.5082 - accuracy: 0.8248 - val_loss: 0.5533 - val_accuracy: 0.81
Epoch 23/100
781/781 [==============================] - ETA: 0s - loss: 0.4998 - accuracy: 0.8280
Epoch 00023: val_accuracy did not improve from 0.82710
781/781 [==============================] - 58s 74ms/step - loss: 0.4998 - accuracy: 0.8280 - val_loss: 0.6017 - val_accuracy: 0.79
Epoch 24/100
781/781 [==============================] - ETA: 0s - loss: 0.4863 - accuracy: 0.8311
Epoch 00024: val_accuracy did not improve from 0.82710
781/781 [==============================] - 58s 74ms/step - loss: 0.4863 - accuracy: 0.8311 - val_loss: 0.5809 - val_accuracy: 0.81
Epoch 25/100
781/781 [==============================] - ETA: 0s - loss: 0.4740 - accuracy: 0.8350
Epoch 00025: val_accuracy improved from 0.82710 to 0.83970, saving model to weights_3.best.hdf5
781/781 [==============================] - 59s 75ms/step - loss: 0.4740 - accuracy: 0.8350 - val_loss: 0.4766 - val_accuracy: 0.83
Epoch 26/100
781/781 [==============================] - ETA: 0s - loss: 0.4676 - accuracy: 0.8368
Epoch 00026: val_accuracy did not improve from 0.83970
781/781 [==============================] - 58s 74ms/step - loss: 0.4676 - accuracy: 0.8368 - val_loss: 0.5126 - val_accuracy: 0.83
Epoch 27/100
781/781 [==============================] - ETA: 0s - loss: 0.4541 - accuracy: 0.8422
Epoch 00027: val_accuracy did not improve from 0.83970
781/781 [==============================] - 58s 74ms/step - loss: 0.4541 - accuracy: 0.8422 - val_loss: 0.5352 - val_accuracy: 0.82
Epoch 28/100
781/781 [==============================] - ETA: 0s - loss: 0.4499 - accuracy: 0.8450
Epoch 00028: val_accuracy did not improve from 0.83970
781/781 [==============================] - 59s 76ms/step - loss: 0.4499 - accuracy: 0.8450 - val_loss: 0.5655 - val_accuracy: 0.81
Epoch 29/100
781/781 [==============================] - ETA: 0s - loss: 0.4430 - accuracy: 0.8473
Epoch 00029: val_accuracy did not improve from 0.83970
781/781 [==============================] - 59s 76ms/step - loss: 0.4430 - accuracy: 0.8473 - val_loss: 0.5665 - val_accuracy: 0.81
Epoch 30/100
781/781 [==============================] - ETA: 0s - loss: 0.4370 - accuracy: 0.8478
Epoch 00030: val_accuracy did not improve from 0.83970
781/781 [==============================] - 60s 77ms/step - loss: 0.4370 - accuracy: 0.8478 - val_loss: 0.4931 - val_accuracy: 0.83
Epoch 31/100
781/781 [==============================] - ETA: 0s - loss: 0.4283 - accuracy: 0.8511
Epoch 00031: val_accuracy improved from 0.83970 to 0.84450, saving model to weights_3.best.hdf5
781/781 [==============================] - 59s 76ms/step - loss: 0.4283 - accuracy: 0.8511 - val_loss: 0.4590 - val_accuracy: 0.84
Epoch 32/100
781/781 [==============================] - ETA: 0s - loss: 0.4161 - accuracy: 0.8543
```

```
Epoch 00032: val_accuracy did not improve from 0.84450
781/781 [==============================] - 58s 75ms/step - loss: 0.4161 - accuracy: 0.8543 - val_loss: 0.5467 - val_accuracy: 0.82
Epoch 33/100
781/781 [==============================] - ETA: 0s - loss: 0.4161 - accuracy: 0.8577
Epoch 00033: val_accuracy did not improve from 0.84450
781/781 [==============================] - 65s 83ms/step - loss: 0.4161 - accuracy: 0.8577 - val_loss: 0.6014 - val_accuracy: 0.81
Epoch 34/100
781/781 [==============================] - ETA: 0s - loss: 0.4128 - accuracy: 0.8566
Epoch 00034: val_accuracy improved from 0.84450 to 0.85680, saving model to weights_3.best.hdf5
781/781 [==============================] - 61s 78ms/step - loss: 0.4128 - accuracy: 0.8566 - val_loss: 0.4464 - val_accuracy: 0.85
Epoch 35/100
781/781 [==============================] - ETA: 0s - loss: 0.4032 - accuracy: 0.8600
Epoch 00035: val_accuracy did not improve from 0.85680
781/781 [==============================] - 61s 79ms/step - loss: 0.4032 - accuracy: 0.8600 - val_loss: 0.4822 - val_accuracy: 0.84
Epoch 36/100
781/781 [==============================] - ETA: 0s - loss: 0.4020 - accuracy: 0.8606
Epoch 00036: val_accuracy improved from 0.85680 to 0.85860, saving model to weights_3.best.hdf5
781/781 [==============================] - 61s 79ms/step - loss: 0.4020 - accuracy: 0.8606 - val_loss: 0.4317 - val_accuracy: 0.85
Epoch 37/100
781/781 [==============================] - ETA: 0s - loss: 0.3871 - accuracy: 0.8650
Epoch 00037: val_accuracy did not improve from 0.85860
781/781 [==============================] - 62s 80ms/step - loss: 0.3871 - accuracy: 0.8650 - val_loss: 0.5045 - val_accuracy: 0.83
Epoch 38/100
781/781 [==============================] - ETA: 0s - loss: 0.3900 - accuracy: 0.8647
Epoch 00038: val_accuracy did not improve from 0.85860
781/781 [==============================] - 62s 79ms/step - loss: 0.3900 - accuracy: 0.8647 - val_loss: 0.5285 - val_accuracy: 0.83
Epoch 39/100
781/781 [==============================] - ETA: 0s - loss: 0.3832 - accuracy: 0.8664
Epoch 00039: val_accuracy did not improve from 0.85860
781/781 [==============================] - 62s 79ms/step - loss: 0.3832 - accuracy: 0.8664 - val_loss: 0.4383 - val_accuracy: 0.85
Epoch 40/100
781/781 [==============================] - ETA: 0s - loss: 0.3812 - accuracy: 0.8674
Epoch 00040: val_accuracy improved from 0.85860 to 0.86980, saving model to weights_3.best.hdf5
781/781 [==============================] - 62s 79ms/step - loss: 0.3812 - accuracy: 0.8674 - val_loss: 0.3963 - val_accuracy: 0.86
Epoch 41/100
781/781 [==============================] - ETA: 0s - loss: 0.3710 - accuracy: 0.8700
Epoch 00041: val_accuracy did not improve from 0.86980
781/781 [==============================] - 61s 79ms/step - loss: 0.3710 - accuracy: 0.8700 - val_loss: 0.5843 - val_accuracy: 0.82
Epoch 42/100
781/781 [==============================] - ETA: 0s - loss: 0.3699 - accuracy: 0.8711
Epoch 00042: val_accuracy did not improve from 0.86980
781/781 [==============================] - 62s 79ms/step - loss: 0.3699 - accuracy: 0.8711 - val_loss: 0.3984 - val_accuracy: 0.86
Epoch 43/100
```

```
781/781 [==============================] - ETA: 0s - loss: 0.3648 - accuracy: 0.8736
Epoch 00043: val_accuracy did not improve from 0.86980
781/781 [==============================] - 60s 77ms/step - loss: 0.3648 - accuracy: 0.8736 - val_loss: 0.6063 - val_accuracy: 0.81
Epoch 44/100
781/781 [==============================] - ETA: 0s - loss: 0.3646 - accuracy: 0.8723
Epoch 00044: val_accuracy did not improve from 0.86980
781/781 [==============================] - 62s 79ms/step - loss: 0.3646 - accuracy: 0.8723 - val_loss: 0.4679 - val_accuracy: 0.84
Epoch 45/100
781/781 [==============================] - ETA: 0s - loss: 0.3592 - accuracy: 0.8761
Epoch 00045: val_accuracy did not improve from 0.86980
781/781 [==============================] - 62s 79ms/step - loss: 0.3592 - accuracy: 0.8761 - val_loss: 0.4600 - val_accuracy: 0.85
Epoch 46/100
781/781 [==============================] - ETA: 0s - loss: 0.3547 - accuracy: 0.8762
Epoch 00046: val_accuracy did not improve from 0.86980
781/781 [==============================] - 61s 79ms/step - loss: 0.3547 - accuracy: 0.8762 - val_loss: 0.4551 - val_accuracy: 0.85
Epoch 47/100
781/781 [==============================] - ETA: 0s - loss: 0.3490 - accuracy: 0.8789
Epoch 00047: val_accuracy did not improve from 0.86980
781/781 [==============================] - 61s 78ms/step - loss: 0.3490 - accuracy: 0.8789 - val_loss: 0.4614 - val_accuracy: 0.85
Epoch 48/100
781/781 [==============================] - ETA: 0s - loss: 0.3451 - accuracy: 0.8793
Epoch 00048: val_accuracy did not improve from 0.86980
781/781 [==============================] - 60s 77ms/step - loss: 0.3451 - accuracy: 0.8793 - val_loss: 0.4438 - val_accuracy: 0.85
Epoch 49/100
781/781 [==============================] - ETA: 0s - loss: 0.3427 - accuracy: 0.8803
Epoch 00049: val_accuracy did not improve from 0.86980
781/781 [==============================] - 60s 77ms/step - loss: 0.3427 - accuracy: 0.8803 - val_loss: 0.4451 - val_accuracy: 0.85
Epoch 50/100
781/781 [==============================] - ETA: 0s - loss: 0.3389 - accuracy: 0.8800
Epoch 00050: val_accuracy did not improve from 0.86980
781/781 [==============================] - 63s 80ms/step - loss: 0.3389 - accuracy: 0.8800 - val_loss: 0.4137 - val_accuracy: 0.86
Epoch 51/100
781/781 [==============================] - ETA: 0s - loss: 0.3379 - accuracy: 0.8820
Epoch 00051: val_accuracy improved from 0.86980 to 0.87440, saving model to weights_3.best.hdf5
781/781 [==============================] - 61s 79ms/step - loss: 0.3379 - accuracy: 0.8820 - val_loss: 0.3945 - val_accuracy: 0.87
Epoch 52/100
781/781 [==============================] - ETA: 0s - loss: 0.3324 - accuracy: 0.8853
Epoch 00052: val_accuracy improved from 0.87440 to 0.87600, saving model to weights_3.best.hdf5
781/781 [==============================] - 62s 79ms/step - loss: 0.3324 - accuracy: 0.8853 - val_loss: 0.3896 - val_accuracy: 0.87
Epoch 53/100
781/781 [==============================] - ETA: 0s - loss: 0.3265 - accuracy: 0.8859
Epoch 00053: val_accuracy did not improve from 0.87600
781/781 [==============================] - 62s 79ms/step - loss: 0.3265 - accuracy: 0.8859 - val_loss: 0.4545 - val_accuracy: 0.85
```

```
Epoch 54/100
781/781 [==============================] - ETA: 0s - loss: 0.3278 - accuracy: 0.8861
Epoch 00054: val_accuracy did not improve from 0.87600
781/781 [==============================] - 62s 80ms/step - loss: 0.3278 - accuracy: 0.8861 - val_loss: 0.4560 - val_accuracy: 0.85
Epoch 55/100
781/781 [==============================] - ETA: 0s - loss: 0.3245 - accuracy: 0.8866
Epoch 00055: val_accuracy did not improve from 0.87600
781/781 [==============================] - 62s 79ms/step - loss: 0.3245 - accuracy: 0.8866 - val_loss: 0.4278 - val_accuracy: 0.85
Epoch 56/100
781/781 [==============================] - ETA: 0s - loss: 0.3209 - accuracy: 0.8892
Epoch 00056: val_accuracy did not improve from 0.87600
781/781 [==============================] - 61s 79ms/step - loss: 0.3209 - accuracy: 0.8892 - val_loss: 0.4604 - val_accuracy: 0.85
Epoch 57/100
781/781 [==============================] - ETA: 0s - loss: 0.3170 - accuracy: 0.8896
Epoch 00057: val_accuracy did not improve from 0.87600
781/781 [==============================] - 60s 77ms/step - loss: 0.3170 - accuracy: 0.8896 - val_loss: 0.4487 - val_accuracy: 0.85
Epoch 58/100
781/781 [==============================] - ETA: 0s - loss: 0.3176 - accuracy: 0.8894
Epoch 00058: val_accuracy improved from 0.87600 to 0.88360, saving model to weights_3.best.hdf5
781/781 [==============================] - 61s 78ms/step - loss: 0.3176 - accuracy: 0.8894 - val_loss: 0.3539 - val_accuracy: 0.88
Epoch 59/100
781/781 [==============================] - ETA: 0s - loss: 0.3107 - accuracy: 0.8917
Epoch 00059: val_accuracy did not improve from 0.88360
781/781 [==============================] - 62s 79ms/step - loss: 0.3107 - accuracy: 0.8917 - val_loss: 0.4208 - val_accuracy: 0.86
Epoch 60/100
781/781 [==============================] - ETA: 0s - loss: 0.3079 - accuracy: 0.8923
Epoch 00060: val_accuracy did not improve from 0.88360
781/781 [==============================] - 61s 78ms/step - loss: 0.3079 - accuracy: 0.8923 - val_loss: 0.4984 - val_accuracy: 0.84
Epoch 61/100
781/781 [==============================] - ETA: 0s - loss: 0.3072 - accuracy: 0.8934
Epoch 00061: val_accuracy did not improve from 0.88360
781/781 [==============================] - 61s 78ms/step - loss: 0.3072 - accuracy: 0.8934 - val_loss: 0.4834 - val_accuracy: 0.84
Epoch 62/100
781/781 [==============================] - ETA: 0s - loss: 0.3088 - accuracy: 0.8913
Epoch 00062: val_accuracy did not improve from 0.88360
781/781 [==============================] - 61s 78ms/step - loss: 0.3088 - accuracy: 0.8913 - val_loss: 0.4529 - val_accuracy: 0.85
Epoch 63/100
781/781 [==============================] - ETA: 0s - loss: 0.2969 - accuracy: 0.8962
Epoch 00063: val_accuracy did not improve from 0.88360
781/781 [==============================] - 60s 77ms/step - loss: 0.2969 - accuracy: 0.8962 - val_loss: 0.5774 - val_accuracy: 0.83
Epoch 64/100
781/781 [==============================] - ETA: 0s - loss: 0.2995 - accuracy: 0.8939
Epoch 00064: val_accuracy did not improve from 0.88360
```

```
781/781 [==============================] - 60s 77ms/step - loss: 0.2995 - accuracy: 0.8939 - val_loss: 0.4959 - val_accuracy: 0.85
Epoch 65/100
781/781 [==============================] - ETA: 0s - loss: 0.2955 - accuracy: 0.8973
Epoch 00065: val_accuracy did not improve from 0.88360
781/781 [==============================] - 60s 77ms/step - loss: 0.2955 - accuracy: 0.8973 - val_loss: 0.3864 - val_accuracy: 0.87
Epoch 66/100
781/781 [==============================] - ETA: 0s - loss: 0.2900 - accuracy: 0.8999
Epoch 00066: val_accuracy did not improve from 0.88360
781/781 [==============================] - 60s 77ms/step - loss: 0.2900 - accuracy: 0.8999 - val_loss: 0.4035 - val_accuracy: 0.87
Epoch 67/100
781/781 [==============================] - ETA: 0s - loss: 0.2881 - accuracy: 0.9000
Epoch 00067: val_accuracy did not improve from 0.88360
781/781 [==============================] - 60s 77ms/step - loss: 0.2881 - accuracy: 0.9000 - val_loss: 0.4484 - val_accuracy: 0.85
Epoch 68/100
781/781 [==============================] - ETA: 0s - loss: 0.2893 - accuracy: 0.8982
Epoch 00068: val_accuracy did not improve from 0.88360
781/781 [==============================] - 59s 76ms/step - loss: 0.2893 - accuracy: 0.8982 - val_loss: 0.4436 - val_accuracy: 0.86
Epoch 69/100
781/781 [==============================] - ETA: 0s - loss: 0.2873 - accuracy: 0.8992
Epoch 00069: val_accuracy did not improve from 0.88360
781/781 [==============================] - 58s 74ms/step - loss: 0.2873 - accuracy: 0.8992 - val_loss: 0.5121 - val_accuracy: 0.84
Epoch 70/100
781/781 [==============================] - ETA: 0s - loss: 0.2857 - accuracy: 0.9015
Epoch 00070: val_accuracy did not improve from 0.88360
781/781 [==============================] - 59s 76ms/step - loss: 0.2857 - accuracy: 0.9015 - val_loss: 0.4016 - val_accuracy: 0.87
Epoch 71/100
```
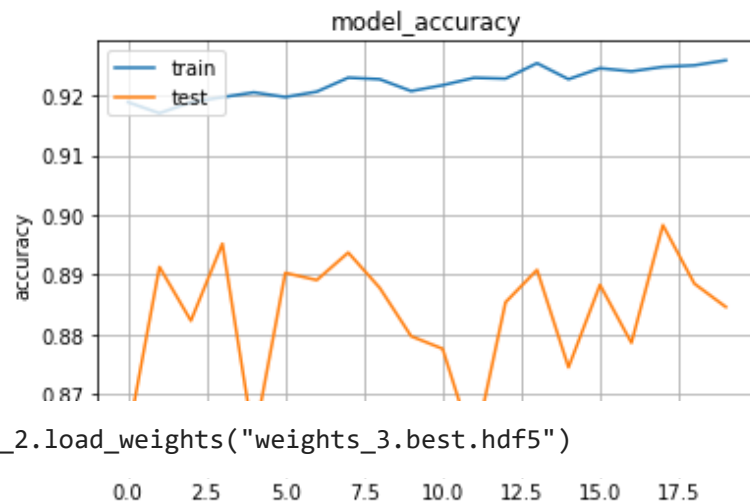
```python
import matplotlib.pyplot as plt
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.grid()
plt.title('model_accuracy')
plt.xlabel('Epochs')
plt.ylabel('accuracy')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```

⊳

```
model_2.load_weights("weights_3.best.hdf5")
```

```
model_2.compile(
loss='categorical_crossentropy',
optimizer=Adam(),
metrics=['accuracy'])
```

```
# Test the model
score = model_2.evaluate(X_test, y_test, verbose=1)
print('Test loss:', score[0])
print('Test accuracy:', score[1])
```

```
313/313 [==============================] - 4s 11ms/step - loss: 0.3292 - accuracy: 0.8983
Test loss: 0.32916033267974854
Test accuracy: 0.8982999920845032
```

Training for 20 more epochs for Better Accuracy

```
history = model_2.fit_generator(img_train,
                    steps_per_epoch=steps,
                    epochs=20,
                    verbose=1,
                    validation_data=(X_test, y_test),
                    callbacks=callback_2)
```

⤳

```
31/781 [==============================] - ETA: 0s - loss: 0.1997 - accuracy: 0.9312
poch 00013: val_accuracy did not improve from 0.90280
31/781 [==============================] - 60s 77ms/step - loss: 0.1997 - accuracy: 0.9312 - val_loss: 0.4479 - val_accuracy: 0.8747


poch 00013: val_accuracy did not improve from 0.90280
31/781 [==============================] - 60s 77ms/step - loss: 0.1997 - accuracy: 0.9312 - val_loss: 0.4479 - val_accuracy: 0.8747
poch 14/20
 1/781 [..............................] - ETA: 0s - loss: 0.2804 - accuracy: 0.9062Epoch 14/20
31/781 [==============================] - ETA: 0s - loss: 0.1958 - accuracy: 0.9304
poch 00014: val_accuracy did not improve from 0.90280
31/781 [==============================] - 60s 76ms/step - loss: 0.1958 - accuracy: 0.9304 - val_loss: 0.3857 - val_accuracy: 0.8926


poch 00014: val_accuracy did not improve from 0.90280
31/781 [==============================] - 60s 76ms/step - loss: 0.1958 - accuracy: 0.9304 - val_loss: 0.3857 - val_accuracy: 0.8926
poch 15/20
 1/781 [..............................] - ETA: 0s - loss: 0.2132 - accuracy: 0.9219Epoch 15/20
31/781 [==============================] - ETA: 0s - loss: 0.2012 - accuracy: 0.9280
poch 00015: val_accuracy did not improve from 0.90280
31/781 [==============================] - 60s 77ms/step - loss: 0.2012 - accuracy: 0.9280 - val_loss: 0.4102 - val_accuracy: 0.8886


poch 00015: val_accuracy did not improve from 0.90280
31/781 [==============================] - 60s 77ms/step - loss: 0.2012 - accuracy: 0.9280 - val_loss: 0.4102 - val_accuracy: 0.8886
poch 16/20
 1/781 [..............................] - ETA: 0s - loss: 0.1446 - accuracy: 0.9688Epoch 16/20
31/781 [==============================] - ETA: 0s - loss: 0.1980 - accuracy: 0.9311
poch 00016: val_accuracy did not improve from 0.90280
```

```python
model_2.load_weights("weights_3.best.hdf5")
```

```
# Test the model
score = model_2.evaluate(X_test, y_test, verbose=1)
print('Test loss:', score[0])
print('Test accuracy:', score[1])
```

```
313/313 [==============================] - 4s 11ms/step - loss: 0.3397 - accuracy: 0.9028
Test loss: 0.3396528363227844
Test accuracy: 0.9028000235557556

 1/781 [..............................] - ETA: 0s - loss: 0.2822 - accuracy: 0.9219Epoch 18/20
```

```
# Please compare all your models using Prettytable library
from prettytable import PrettyTable
```

```
x = PrettyTable()
x.field_names = ["Model", "kernel", "filters", "Train Accuracy(%)", "Test accuracy(%)"]
x.add_row(["Model 1", "3x3", 12, 80.19, 74.01]),
x.add_row(["Model 2", "3x3", 24, 93.13,90.28]),


print(x)
```

```
+---------+--------+---------+-------------------+------------------+
|  Model  | kernel | filters | Train Accuracy(%) | Test accuracy(%) |
+---------+--------+---------+-------------------+------------------+
| Model 1 |  3x3   |   12    |       80.19       |      74.01       |
| Model 2 |  3x3   |   24    |       93.13       |      90.28       |
+---------+--------+---------+-------------------+------------------+
```

```
31/781 [==============================] - 59s 76ms/step - loss: 0.1958 - accuracy: 0.9313 - val_loss: 0.3840 - val_accuracy: 0.8914
```