Problem Statement

Create an image classifier to identify the actors in the

images within the test dataset with any of keras, tensorflow, pytorch or scikit-learn.

In [4]:
```
cd C:\Users\harsha.teja\Desktop\myg\Test\convertics\indian-actor-dataset
```

```
C:\Users\harsha.teja\Desktop\myg\Test\convertics\indian-actor-dataset
```

In [2]:
```
#Import all the Dependencies
import pandas as pd
import numpy as np
import tensorflow as tf
from tensorflow.keras import models,layers
import matplotlib.pyplot as plt
from tensorflow import keras
```

```
C:\Users\harsha.teja\Anaconda3\envs\deeplearning\lib\site-packages\numpy\_distributo
r_init.py:32: UserWarning: loaded more than 1 DLL from .libs:
C:\Users\harsha.teja\Anaconda3\envs\deeplearning\lib\site-packages\numpy\.libs\libop
enblas.PYQHXLVVQ7VESDPUVUADXEVJOBGHJPAY.gfortran-win_amd64.dll
C:\Users\harsha.teja\Anaconda3\envs\deeplearning\lib\site-packages\numpy\.libs\libop
enblas.QVLO2T66WEPI7JZ63PS3HMOHFEY472BC.gfortran-win_amd64.dll
C:\Users\harsha.teja\Anaconda3\envs\deeplearning\lib\site-packages\numpy\.libs\libop
enblas.WCDJNK7YVMPZQ2ME2ZZHJJRJ3JIKNDB7.gfortran-win_amd64.dll
  stacklevel=1)
```

In [3]:
```
#Set all the Constants
BATCH_SIZE = 32
IMAGE_SIZE = 160
CHANNELS=3
EPOCHS=50
```

Import Actors Dataset

In [5]:
```
dataset = tf.keras.preprocessing.image_dataset_from_directory(
    "train",
    seed=123,
    shuffle=True,
    image_size=(IMAGE_SIZE,IMAGE_SIZE),
    batch_size=BATCH_SIZE
)
```

```
Found 4694 files belonging to 135 classes.
```

In [6]:
```
class_names = dataset.class_names
print(class_names)
len(class_names)
```

```
['abhay_deol', 'adil_hussain', 'ajay_devgn', 'akshay_kumar', 'akshaye_khanna', 'amit
abh_bachchan', 'amjad_khan', 'amol_palekar', 'amole_gupte', 'amrish_puri', 'anil_kap
oor', 'annu_kapoor', 'anupam_kher', 'anushka_shetty', 'arshad_warsi', 'aruna_irani',
'ashish_vidyarthi', 'asrani', 'atul_kulkarni', 'ayushmann_khurrana', 'boman_irani',
```

```
'chiranjeevi', 'chunky_panday', 'danny_denzongpa', 'darsheel_safary', 'deepika_paduk
one', 'deepti_naval', 'dev_anand', 'dharmendra', 'dilip_kumar', 'dimple_kapadia', 'f
arhan_akhtar', 'farida_jalal', 'farooq_shaikh', 'girish_karnad', 'govinda', 'gulshan
_grover', 'hrithik_roshan', 'huma_qureshi', 'irrfan_khan', 'jaspal_bhatti', 'jeetend
ra', 'jimmy_sheirgill', 'johnny_lever', 'kader_khan', 'kajol', 'kalki_koechlin', 'ka
mal_haasan', 'kangana_ranaut', 'kay_kay_menon', 'konkona_sen_sharma', 'kulbhushan_kh
arbanda', 'lara_dutta', 'madhavan', 'madhuri_dixit', 'mammootty', 'manoj_bajpayee',
'manoj_pahwa', 'mehmood', 'mita_vashisht', 'mithun_chakraborty', 'mohanlal', 'mohnis
h_bahl', 'mukesh_khanna', 'mukul_dev', 'nagarjuna_akkineni', 'nana_patekar', 'nandit
a_das', 'nargis', 'naseeruddin_shah', 'navin_nischol', 'nawazuddin_siddiqui', 'neera
j_kabi', 'nirupa_roy', 'om_puri', 'pankaj_kapur', 'pankaj_tripathi', 'paresh_rawal',
'pawan_malhotra', 'pooja_bhatt', 'prabhas', 'prabhu_deva', 'prakash_raj', 'pran', 'p
rem_chopra', 'priyanka_chopra', 'raaj_kumar', 'radhika_apte', 'rahul_bose', 'raj_bab
bar', 'raj_kapoor', 'rajat_kapoor', 'rajesh_khanna', 'rajinikanth', 'rajit_kapoor',
'rajkummar_rao', 'rajpal_yadav', 'rakhee_gulzar', 'ramya_krishnan', 'ranbir_kapoor',
'randeep_hooda', 'rani_mukerji', 'ranveer_singh', 'ranvir_shorey', 'ratna_pathak_sha
h', 'rekha', 'richa_chadha', 'rishi_kapoor', 'riteish_deshmukh', 'sachin_khedekar',
'saeed_jaffrey', 'saif_ali_khan', 'salman_khan', 'sanjay_dutt', 'sanjay_mishra', 'sh
abana_azmi', 'shah_rukh_khan', 'sharman_joshi', 'sharmila_tagore', 'shashi_kapoor',
'shreyas_talpade', 'smita_patil', 'soumitra_chatterjee', 'sridevi', 'sunil_shetty',
'sunny_deol', 'tabu', 'tinnu_anand', 'utpal_dutt', 'varun_dhawan', 'vidya_balan', 'v
inod_khanna', 'waheeda_rehman', 'zarina_wahab', 'zeenat_aman']
```

Out[6]: 135

In [7]:
```
len(dataset)
```

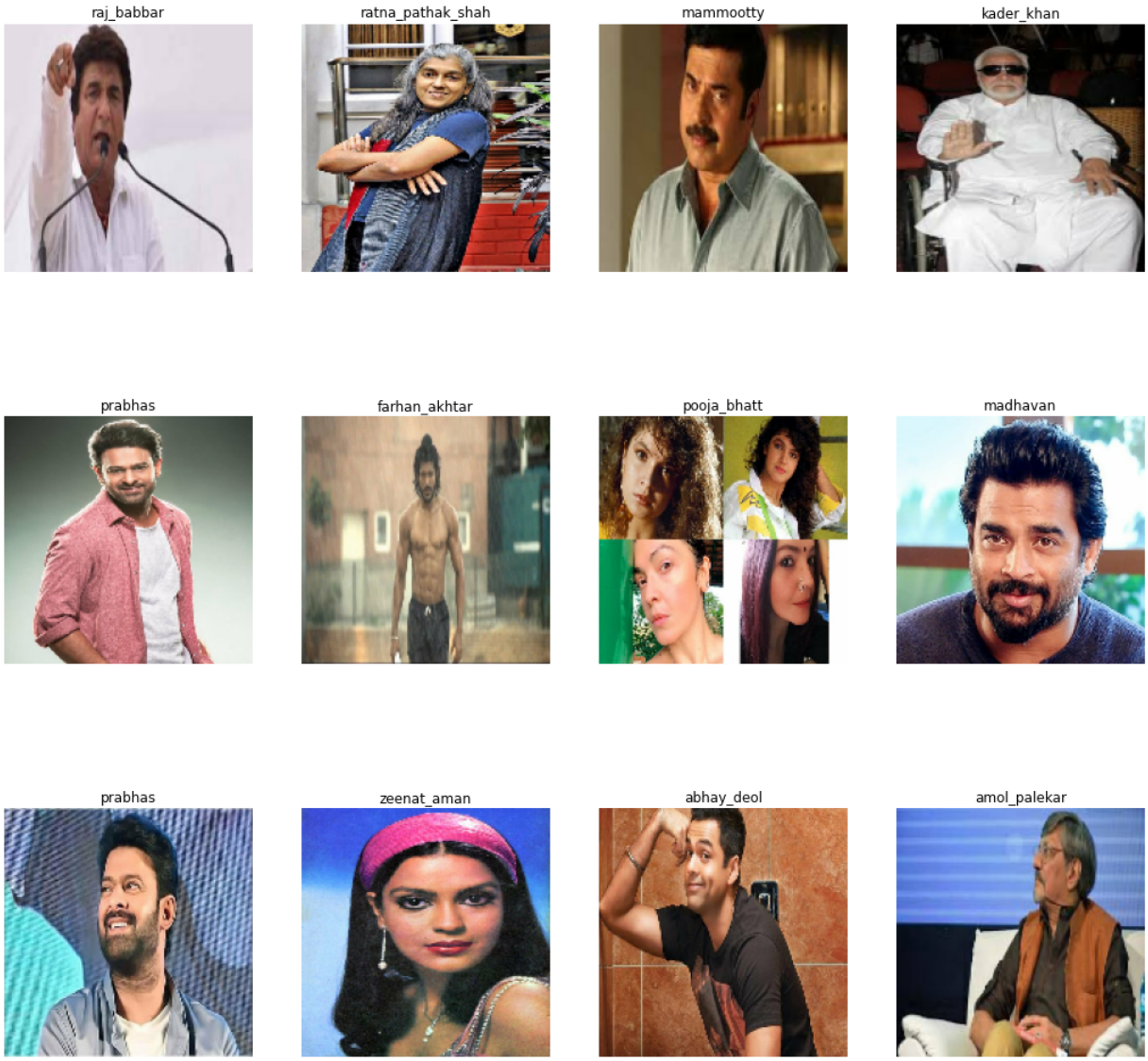
Out[7]: 147

In [8]:
```
class_names = np.array(dataset.class_names)
print(class_names)
```

```
['abhay_deol' 'adil_hussain' 'ajay_devgn' 'akshay_kumar' 'akshaye_khanna'
 'amitabh_bachchan' 'amjad_khan' 'amol_palekar' 'amole_gupte'
 'amrish_puri' 'anil_kapoor' 'annu_kapoor' 'anupam_kher' 'anushka_shetty'
 'arshad_warsi' 'aruna_irani' 'ashish_vidyarthi' 'asrani' 'atul_kulkarni'
 'ayushmann_khurrana' 'boman_irani' 'chiranjeevi' 'chunky_panday'
 'danny_denzongpa' 'darsheel_safary' 'deepika_padukone' 'deepti_naval'
 'dev_anand' 'dharmendra' 'dilip_kumar' 'dimple_kapadia' 'farhan_akhtar'
 'farida_jalal' 'farooq_shaikh' 'girish_karnad' 'govinda' 'gulshan_grover'
 'hrithik_roshan' 'huma_qureshi' 'irrfan_khan' 'jaspal_bhatti' 'jeetendra'
 'jimmy_sheirgill' 'johnny_lever' 'kader_khan' 'kajol' 'kalki_koechlin'
 'kamal_haasan' 'kangana_ranaut' 'kay_kay_menon' 'konkona_sen_sharma'
 'kulbhushan_kharbanda' 'lara_dutta' 'madhavan' 'madhuri_dixit'
 'mammootty' 'manoj_bajpayee' 'manoj_pahwa' 'mehmood' 'mita_vashisht'
 'mithun_chakraborty' 'mohanlal' 'mohnish_bahl' 'mukesh_khanna'
 'mukul_dev' 'nagarjuna_akkineni' 'nana_patekar' 'nandita_das' 'nargis'
 'naseeruddin_shah' 'navin_nischol' 'nawazuddin_siddiqui' 'neeraj_kabi'
 'nirupa_roy' 'om_puri' 'pankaj_kapur' 'pankaj_tripathi' 'paresh_rawal'
 'pawan_malhotra' 'pooja_bhatt' 'prabhas' 'prabhu_deva' 'prakash_raj'
 'pran' 'prem_chopra' 'priyanka_chopra' 'raaj_kumar' 'radhika_apte'
 'rahul_bose' 'raj_babbar' 'raj_kapoor' 'rajat_kapoor' 'rajesh_khanna'
 'rajinikanth' 'rajit_kapoor' 'rajkummar_rao' 'rajpal_yadav'
 'rakhee_gulzar' 'ramya_krishnan' 'ranbir_kapoor' 'randeep_hooda'
 'rani_mukerji' 'ranveer_singh' 'ranvir_shorey' 'ratna_pathak_shah'
 'rekha' 'richa_chadha' 'rishi_kapoor' 'riteish_deshmukh'
 'sachin_khedekar' 'saeed_jaffrey' 'saif_ali_khan' 'salman_khan'
 'sanjay_dutt' 'sanjay_mishra' 'shabana_azmi' 'shah_rukh_khan'
 'sharman_joshi' 'sharmila_tagore' 'shashi_kapoor' 'shreyas_talpade'
 'smita_patil' 'soumitra_chatterjee' 'sridevi' 'sunil_shetty' 'sunny_deol'
 'tabu' 'tinnu_anand' 'utpal_dutt' 'varun_dhawan' 'vidya_balan'
 'vinod_khanna' 'waheeda_rehman' 'zarina_wahab' 'zeenat_aman']
```

In [9]:
```
#plotting somesamples
plt.figure(figsize=(18, 18))
for image_batch, labels_batch in dataset.take(1):
    for i in range(12):
        ax = plt.subplot(3, 4, i + 1)
```

```
        plt.imshow(image_batch[i].numpy().astype("uint8"))
        plt.title(class_names[labels_batch[i]])
        plt.axis("off")
```



Function to Split Dataset

Dataset should be bifurcated into 3 subsets, namely:

Training: Dataset to be used while training Validation: Dataset to be tested against while training Test: Dataset to be tested against after we trained a model

In [14]:
```
def get_dataset_partitions_tf(ds, train_split=0.7, val_split=0.15, test_split=0.15,
    assert (train_split + test_split + val_split) == 1

    ds_size = len(ds)

    if shuffle:
        ds = ds.shuffle(shuffle_size, seed=12)

    train_size = int(train_split * ds_size)
    val_size = int(val_split * ds_size)

    train_ds = ds.take(train_size)
    val_ds = ds.skip(train_size).take(val_size)
    test_ds = ds.skip(train_size).skip(val_size)
```

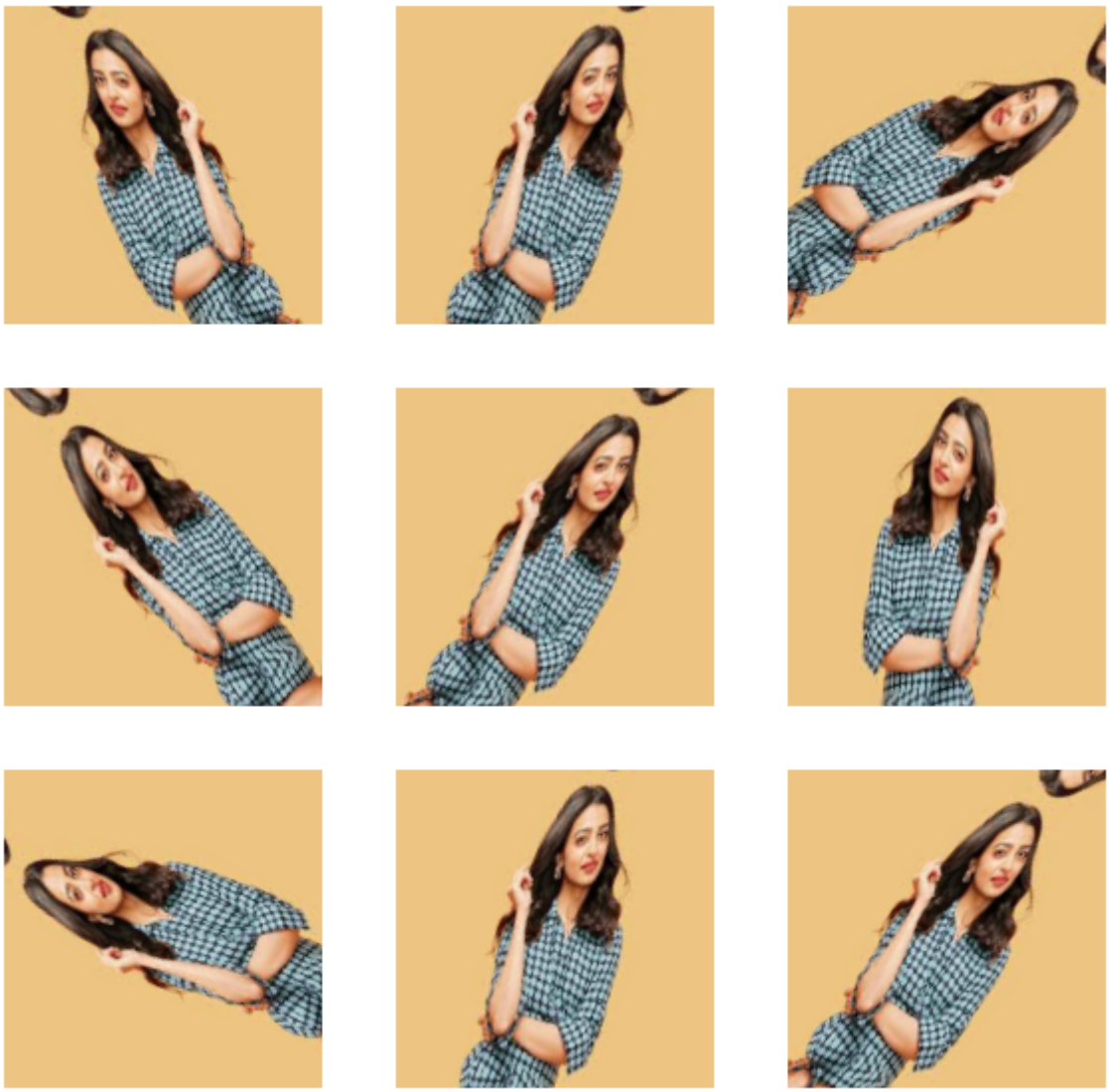
```
    return train_ds, val_ds, test_ds
```

In [15]:
```
train_dataset, validation_dataset , test_dataset = get_dataset_partitions_tf(dataset
```

In [54]:
```
AUTOTUNE = tf.data.AUTOTUNE

train_dataset = train_dataset.prefetch(buffer_size=AUTOTUNE)
validation_dataset = validation_dataset.prefetch(buffer_size=AUTOTUNE)
test_dataset = test_dataset.prefetch(buffer_size=AUTOTUNE)
```

In [55]:
```
#data_augmentation
data_augmentation = tf.keras.Sequential([
  tf.keras.layers.experimental.preprocessing.RandomFlip('horizontal'),
  tf.keras.layers.experimental.preprocessing.RandomRotation(0.2),
])
```

In [56]:
```
#plotting data_augmentation
for image, _ in train_dataset.take(2):
  plt.figure(figsize=(10, 10))
  first_image = image[0]
  for i in range(9):
    ax = plt.subplot(3, 3, i + 1)
    augmented_image = data_augmentation(tf.expand_dims(first_image, 0))
    plt.imshow(augmented_image[0] / 255)
    plt.axis('off')
```



In [60]:
```
#Downloading mobile v2
preprocess_input = tf.keras.applications.mobilenet_v2.preprocess_input
rescale = tf.keras.layers.experimental.preprocessing.Rescaling(1./127.5, offset= -1)
IMG_SIZE = (160,160)
# Create the base model from the pre-trained model MobileNet V2
IMG_SHAPE = IMG_SIZE + (3,)
base_model = tf.keras.applications.MobileNetV2(input_shape=IMG_SHAPE,
                                               include_top=False,
                                               weights='imagenet')
```

In [61]:
```
#checking shape
image_batch, label_batch = next(iter(train_dataset))
feature_batch = base_model(image_batch)
print(feature_batch.shape)
```

```
(32, 5, 5, 1280)
```

In [62]:

In [63]:
```
# Let's take a look at the base model architecture
base_model.trainable = False
base_model.summary()
```

```
Model: "mobilenetv2_1.00_160"
_________________________________________________________________________________________
Layer (type)                 Output Shape              Param #     Connected to
==========================================================================================
==============
```

```
input_3 (InputLayer)            [(None, 160, 160, 3) 0
__________________________________________________________________________________________________
Conv1 (Conv2D)                  (None, 80, 80, 32)   864         input_3[0][0]
__________________________________________________________________________________________________
bn_Conv1 (BatchNormalization)   (None, 80, 80, 32)   128         Conv1[0][0]
__________________________________________________________________________________________________
Conv1_relu (ReLU)               (None, 80, 80, 32)   0           bn_Conv1[0][0]
__________________________________________________________________________________________________
expanded_conv_depthwise (Depthw (None, 80, 80, 32)   288         Conv1_relu[0][0]
__________________________________________________________________________________________________
expanded_conv_depthwise_BN (Bat (None, 80, 80, 32)   128         expanded_conv_depth
wise[0][0]
__________________________________________________________________________________________________
expanded_conv_depthwise_relu (R (None, 80, 80, 32)   0           expanded_conv_depth
wise_BN[0][0]
__________________________________________________________________________________________________
expanded_conv_project (Conv2D)  (None, 80, 80, 16)   512         expanded_conv_depth
wise_relu[0][0
__________________________________________________________________________________________________
expanded_conv_project_BN (Batch (None, 80, 80, 16)   64          expanded_conv_proje
ct[0][0]
__________________________________________________________________________________________________
block_1_expand (Conv2D)         (None, 80, 80, 96)   1536        expanded_conv_proje
ct_BN[0][0]
__________________________________________________________________________________________________
block_1_expand_BN (BatchNormali (None, 80, 80, 96)   384         block_1_expand[0]
[0]
__________________________________________________________________________________________________
block_1_expand_relu (ReLU)      (None, 80, 80, 96)   0           block_1_expand_BN
[0][0]
__________________________________________________________________________________________________
block_1_pad (ZeroPadding2D)     (None, 81, 81, 96)   0           block_1_expand_relu
[0][0]
__________________________________________________________________________________________________
block_1_depthwise (DepthwiseCon (None, 40, 40, 96)   864         block_1_pad[0][0]
__________________________________________________________________________________________________
block_1_depthwise_BN (BatchNorm (None, 40, 40, 96)   384         block_1_depthwise
[0][0]
__________________________________________________________________________________________________
block_1_depthwise_relu (ReLU)   (None, 40, 40, 96)   0           block_1_depthwise_B
N[0][0]
__________________________________________________________________________________________________
block_1_project (Conv2D)        (None, 40, 40, 24)   2304        block_1_depthwise_r
elu[0][0]
__________________________________________________________________________________________________
block_1_project_BN (BatchNormal (None, 40, 40, 24)   96          block_1_project[0]
[0]
__________________________________________________________________________________________________
block_2_expand (Conv2D)         (None, 40, 40, 144)  3456        block_1_project_BN
[0][0]
__________________________________________________________________________________________________
```

```
_______________
block_2_expand_BN (BatchNormali (None, 40, 40, 144)  576         block_2_expand[0]
[0]
_____________________________________________________________________________
_______________
block_2_expand_relu (ReLU)      (None, 40, 40, 144)  0           block_2_expand_BN
[0][0]
_____________________________________________________________________________
_______________
block_2_depthwise (DepthwiseCon (None, 40, 40, 144)  1296        block_2_expand_relu
[0][0]
_____________________________________________________________________________
_______________
block_2_depthwise_BN (BatchNorm (None, 40, 40, 144)  576         block_2_depthwise
[0][0]
_____________________________________________________________________________
_______________
block_2_depthwise_relu (ReLU)   (None, 40, 40, 144)  0           block_2_depthwise_B
N[0][0]
_____________________________________________________________________________
_______________
block_2_project (Conv2D)        (None, 40, 40, 24)   3456        block_2_depthwise_r
elu[0][0]
_____________________________________________________________________________
_______________
block_2_project_BN (BatchNormal (None, 40, 40, 24)   96          block_2_project[0]
[0]
_____________________________________________________________________________
_______________
block_2_add (Add)               (None, 40, 40, 24)   0           block_1_project_BN
[0][0]
                                                                 block_2_project_BN
[0][0]
_____________________________________________________________________________
_______________
block_3_expand (Conv2D)         (None, 40, 40, 144)  3456        block_2_add[0][0]
_____________________________________________________________________________
_______________
block_3_expand_BN (BatchNormali (None, 40, 40, 144)  576         block_3_expand[0]
[0]
_____________________________________________________________________________
_______________
block_3_expand_relu (ReLU)      (None, 40, 40, 144)  0           block_3_expand_BN
[0][0]
_____________________________________________________________________________
_______________
block_3_pad (ZeroPadding2D)     (None, 41, 41, 144)  0           block_3_expand_relu
[0][0]
_____________________________________________________________________________
_______________
block_3_depthwise (DepthwiseCon (None, 20, 20, 144)  1296        block_3_pad[0][0]
_____________________________________________________________________________
_______________
block_3_depthwise_BN (BatchNorm (None, 20, 20, 144)  576         block_3_depthwise
[0][0]
_____________________________________________________________________________
_______________
block_3_depthwise_relu (ReLU)   (None, 20, 20, 144)  0           block_3_depthwise_B
N[0][0]
_____________________________________________________________________________
_______________
block_3_project (Conv2D)        (None, 20, 20, 32)   4608        block_3_depthwise_r
elu[0][0]
_____________________________________________________________________________
_______________
block_3_project_BN (BatchNormal (None, 20, 20, 32)   128         block_3_project[0]
[0]
_____________________________________________________________________________
_______________
```

```
block_4_expand (Conv2D)         (None, 20, 20, 192)  6144        block_3_project_BN
[0][0]
__________________________________________________________________________________
_______________
block_4_expand_BN (BatchNormali (None, 20, 20, 192)  768         block_4_expand[0]
[0]
__________________________________________________________________________________
_______________
block_4_expand_relu (ReLU)      (None, 20, 20, 192)  0           block_4_expand_BN
[0][0]
__________________________________________________________________________________
_______________
block_4_depthwise (DepthwiseCon (None, 20, 20, 192)  1728        block_4_expand_relu
[0][0]
__________________________________________________________________________________
_______________
block_4_depthwise_BN (BatchNorm (None, 20, 20, 192)  768         block_4_depthwise
[0][0]
__________________________________________________________________________________
_______________
block_4_depthwise_relu (ReLU)   (None, 20, 20, 192)  0           block_4_depthwise_B
N[0][0]
__________________________________________________________________________________
_______________
block_4_project (Conv2D)        (None, 20, 20, 32)   6144        block_4_depthwise_r
elu[0][0]
__________________________________________________________________________________
_______________
block_4_project_BN (BatchNormal (None, 20, 20, 32)   128         block_4_project[0]
[0]
__________________________________________________________________________________
_______________
block_4_add (Add)               (None, 20, 20, 32)   0           block_3_project_BN
[0][0]
                                                                 block_4_project_BN
[0][0]
__________________________________________________________________________________
_______________
block_5_expand (Conv2D)         (None, 20, 20, 192)  6144        block_4_add[0][0]
__________________________________________________________________________________
_______________
block_5_expand_BN (BatchNormali (None, 20, 20, 192)  768         block_5_expand[0]
[0]
__________________________________________________________________________________
_______________
block_5_expand_relu (ReLU)      (None, 20, 20, 192)  0           block_5_expand_BN
[0][0]
__________________________________________________________________________________
_______________
block_5_depthwise (DepthwiseCon (None, 20, 20, 192)  1728        block_5_expand_relu
[0][0]
__________________________________________________________________________________
_______________
block_5_depthwise_BN (BatchNorm (None, 20, 20, 192)  768         block_5_depthwise
[0][0]
__________________________________________________________________________________
_______________
block_5_depthwise_relu (ReLU)   (None, 20, 20, 192)  0           block_5_depthwise_B
N[0][0]
__________________________________________________________________________________
_______________
block_5_project (Conv2D)        (None, 20, 20, 32)   6144        block_5_depthwise_r
elu[0][0]
__________________________________________________________________________________
_______________
block_5_project_BN (BatchNormal (None, 20, 20, 32)   128         block_5_project[0]
[0]
__________________________________________________________________________________
_______________
```

```
block_5_add (Add)                (None, 20, 20, 32)   0           block_4_add[0][0]
                                                                  block_5_project_BN
[0][0]
__________________________________________________________________________________
_______________
block_6_expand (Conv2D)          (None, 20, 20, 192)  6144        block_5_add[0][0]
__________________________________________________________________________________
_______________
block_6_expand_BN (BatchNormali (None, 20, 20, 192)  768         block_6_expand[0]
[0]
__________________________________________________________________________________
_______________
block_6_expand_relu (ReLU)       (None, 20, 20, 192)  0           block_6_expand_BN
[0][0]
__________________________________________________________________________________
_______________
block_6_pad (ZeroPadding2D)      (None, 21, 21, 192)  0           block_6_expand_relu
[0][0]
__________________________________________________________________________________
_______________
block_6_depthwise (DepthwiseCon (None, 10, 10, 192)  1728        block_6_pad[0][0]
__________________________________________________________________________________
_______________
block_6_depthwise_BN (BatchNorm (None, 10, 10, 192)  768         block_6_depthwise
[0][0]
__________________________________________________________________________________
_______________
block_6_depthwise_relu (ReLU)    (None, 10, 10, 192)  0           block_6_depthwise_B
N[0][0]
__________________________________________________________________________________
_______________
block_6_project (Conv2D)         (None, 10, 10, 64)   12288       block_6_depthwise_r
elu[0][0]
__________________________________________________________________________________
_______________
block_6_project_BN (BatchNormal (None, 10, 10, 64)   256         block_6_project[0]
[0]
__________________________________________________________________________________
_______________
block_7_expand (Conv2D)          (None, 10, 10, 384)  24576       block_6_project_BN
[0][0]
__________________________________________________________________________________
_______________
block_7_expand_BN (BatchNormali (None, 10, 10, 384)  1536        block_7_expand[0]
[0]
__________________________________________________________________________________
_______________
block_7_expand_relu (ReLU)       (None, 10, 10, 384)  0           block_7_expand_BN
[0][0]
__________________________________________________________________________________
_______________
block_7_depthwise (DepthwiseCon (None, 10, 10, 384)  3456        block_7_expand_relu
[0][0]
__________________________________________________________________________________
_______________
block_7_depthwise_BN (BatchNorm (None, 10, 10, 384)  1536        block_7_depthwise
[0][0]
__________________________________________________________________________________
_______________
block_7_depthwise_relu (ReLU)    (None, 10, 10, 384)  0           block_7_depthwise_B
N[0][0]
__________________________________________________________________________________
_______________
block_7_project (Conv2D)         (None, 10, 10, 64)   24576       block_7_depthwise_r
elu[0][0]
__________________________________________________________________________________
_______________
block_7_project_BN (BatchNormal (None, 10, 10, 64)   256         block_7_project[0]
[0]
```

```
__________________________________________________________________________________________________
_______________
block_7_add (Add)               (None, 10, 10, 64)   0           block_6_project_BN
[0][0]
                                                                 block_7_project_BN
[0][0]
__________________________________________________________________________________________________
_______________
block_8_expand (Conv2D)         (None, 10, 10, 384)  24576       block_7_add[0][0]
__________________________________________________________________________________________________
_______________
block_8_expand_BN (BatchNormali (None, 10, 10, 384)  1536        block_8_expand[0]
[0]
__________________________________________________________________________________________________
_______________
block_8_expand_relu (ReLU)      (None, 10, 10, 384)  0           block_8_expand_BN
[0][0]
__________________________________________________________________________________________________
_______________
block_8_depthwise (DepthwiseCon (None, 10, 10, 384)  3456        block_8_expand_relu
[0][0]
__________________________________________________________________________________________________
_______________
block_8_depthwise_BN (BatchNorm (None, 10, 10, 384)  1536        block_8_depthwise
[0][0]
__________________________________________________________________________________________________
_______________
block_8_depthwise_relu (ReLU)   (None, 10, 10, 384)  0           block_8_depthwise_B
N[0][0]
__________________________________________________________________________________________________
_______________
block_8_project (Conv2D)        (None, 10, 10, 64)   24576       block_8_depthwise_r
elu[0][0]
__________________________________________________________________________________________________
_______________
block_8_project_BN (BatchNormal (None, 10, 10, 64)   256         block_8_project[0]
[0]
__________________________________________________________________________________________________
_______________
block_8_add (Add)               (None, 10, 10, 64)   0           block_7_add[0][0]
                                                                 block_8_project_BN
[0][0]
__________________________________________________________________________________________________
_______________
block_9_expand (Conv2D)         (None, 10, 10, 384)  24576       block_8_add[0][0]
__________________________________________________________________________________________________
_______________
block_9_expand_BN (BatchNormali (None, 10, 10, 384)  1536        block_9_expand[0]
[0]
__________________________________________________________________________________________________
_______________
block_9_expand_relu (ReLU)      (None, 10, 10, 384)  0           block_9_expand_BN
[0][0]
__________________________________________________________________________________________________
_______________
block_9_depthwise (DepthwiseCon (None, 10, 10, 384)  3456        block_9_expand_relu
[0][0]
__________________________________________________________________________________________________
_______________
block_9_depthwise_BN (BatchNorm (None, 10, 10, 384)  1536        block_9_depthwise
[0][0]
__________________________________________________________________________________________________
_______________
block_9_depthwise_relu (ReLU)   (None, 10, 10, 384)  0           block_9_depthwise_B
N[0][0]
__________________________________________________________________________________________________
_______________
block_9_project (Conv2D)        (None, 10, 10, 64)   24576       block_9_depthwise_r
elu[0][0]
```

```
__________________________________________________________________________________________________
_______________
block_9_project_BN (BatchNormal (None, 10, 10, 64)   256         block_9_project[0]
[0]
__________________________________________________________________________________________________
_______________
block_9_add (Add)               (None, 10, 10, 64)   0           block_8_add[0][0]
                                                                 block_9_project_BN
[0][0]
__________________________________________________________________________________________________
_______________
block_10_expand (Conv2D)        (None, 10, 10, 384)  24576       block_9_add[0][0]
__________________________________________________________________________________________________
_______________
block_10_expand_BN (BatchNormal (None, 10, 10, 384)  1536        block_10_expand[0]
[0]
__________________________________________________________________________________________________
_______________
block_10_expand_relu (ReLU)     (None, 10, 10, 384)  0           block_10_expand_BN
[0][0]
__________________________________________________________________________________________________
_______________
block_10_depthwise (DepthwiseCo (None, 10, 10, 384)  3456        block_10_expand_rel
u[0][0]
__________________________________________________________________________________________________
_______________
block_10_depthwise_BN (BatchNor (None, 10, 10, 384)  1536        block_10_depthwise
[0][0]
__________________________________________________________________________________________________
_______________
block_10_depthwise_relu (ReLU)  (None, 10, 10, 384)  0           block_10_depthwise_
BN[0][0]
__________________________________________________________________________________________________
_______________
block_10_project (Conv2D)       (None, 10, 10, 96)   36864       block_10_depthwise_
relu[0][0]
__________________________________________________________________________________________________
_______________
block_10_project_BN (BatchNorma (None, 10, 10, 96)   384         block_10_project[0]
[0]
__________________________________________________________________________________________________
_______________
block_11_expand (Conv2D)        (None, 10, 10, 576)  55296       block_10_project_BN
[0][0]
__________________________________________________________________________________________________
_______________
block_11_expand_BN (BatchNormal (None, 10, 10, 576)  2304        block_11_expand[0]
[0]
__________________________________________________________________________________________________
_______________
block_11_expand_relu (ReLU)     (None, 10, 10, 576)  0           block_11_expand_BN
[0][0]
__________________________________________________________________________________________________
_______________
block_11_depthwise (DepthwiseCo (None, 10, 10, 576)  5184        block_11_expand_rel
u[0][0]
__________________________________________________________________________________________________
_______________
block_11_depthwise_BN (BatchNor (None, 10, 10, 576)  2304        block_11_depthwise
[0][0]
__________________________________________________________________________________________________
_______________
block_11_depthwise_relu (ReLU)  (None, 10, 10, 576)  0           block_11_depthwise_
BN[0][0]
__________________________________________________________________________________________________
_______________
block_11_project (Conv2D)       (None, 10, 10, 96)   55296       block_11_depthwise_
relu[0][0]
__________________________________________________________________________________________________
```

```
_____________
block_11_project_BN (BatchNorma (None, 10, 10, 96)   384         block_11_project[0]
[0]
_____________________________________________________________________________________
_____________
block_11_add (Add)              (None, 10, 10, 96)   0           block_10_project_BN
[0][0]
                                                                 block_11_project_BN
[0][0]
_____________________________________________________________________________________
_____________
block_12_expand (Conv2D)        (None, 10, 10, 576)  55296       block_11_add[0][0]
_____________________________________________________________________________________
_____________
block_12_expand_BN (BatchNormal (None, 10, 10, 576)  2304        block_12_expand[0]
[0]
_____________________________________________________________________________________
_____________
block_12_expand_relu (ReLU)     (None, 10, 10, 576)  0           block_12_expand_BN
[0][0]
_____________________________________________________________________________________
_____________
block_12_depthwise (DepthwiseCo (None, 10, 10, 576)  5184        block_12_expand_rel
u[0][0]
_____________________________________________________________________________________
_____________
block_12_depthwise_BN (BatchNor (None, 10, 10, 576)  2304        block_12_depthwise
[0][0]
_____________________________________________________________________________________
_____________
block_12_depthwise_relu (ReLU)  (None, 10, 10, 576)  0           block_12_depthwise_
BN[0][0]
_____________________________________________________________________________________
_____________
block_12_project (Conv2D)       (None, 10, 10, 96)   55296       block_12_depthwise_
relu[0][0]
_____________________________________________________________________________________
_____________
block_12_project_BN (BatchNorma (None, 10, 10, 96)   384         block_12_project[0]
[0]
_____________________________________________________________________________________
_____________
block_12_add (Add)              (None, 10, 10, 96)   0           block_11_add[0][0]
                                                                 block_12_project_BN
[0][0]
_____________________________________________________________________________________
_____________
block_13_expand (Conv2D)        (None, 10, 10, 576)  55296       block_12_add[0][0]
_____________________________________________________________________________________
_____________
block_13_expand_BN (BatchNormal (None, 10, 10, 576)  2304        block_13_expand[0]
[0]
_____________________________________________________________________________________
_____________
block_13_expand_relu (ReLU)     (None, 10, 10, 576)  0           block_13_expand_BN
[0][0]
_____________________________________________________________________________________
_____________
block_13_pad (ZeroPadding2D)    (None, 11, 11, 576)  0           block_13_expand_rel
u[0][0]
_____________________________________________________________________________________
_____________
block_13_depthwise (DepthwiseCo (None, 5, 5, 576)    5184        block_13_pad[0][0]
_____________________________________________________________________________________
_____________
block_13_depthwise_BN (BatchNor (None, 5, 5, 576)    2304        block_13_depthwise
[0][0]
_____________________________________________________________________________________
_____________
```

```
block_13_depthwise_relu (ReLU)  (None, 5, 5, 576)    0           block_13_depthwise_
BN[0][0]
__________________________________________________________________________________
_______________
block_13_project (Conv2D)       (None, 5, 5, 160)    92160       block_13_depthwise_
relu[0][0]
__________________________________________________________________________________
_______________
block_13_project_BN (BatchNorma (None, 5, 5, 160)    640         block_13_project[0]
[0]
__________________________________________________________________________________
_______________
block_14_expand (Conv2D)        (None, 5, 5, 960)    153600      block_13_project_BN
[0][0]
__________________________________________________________________________________
_______________
block_14_expand_BN (BatchNormal (None, 5, 5, 960)    3840        block_14_expand[0]
[0]
__________________________________________________________________________________
_______________
block_14_expand_relu (ReLU)     (None, 5, 5, 960)    0           block_14_expand_BN
[0][0]
__________________________________________________________________________________
_______________
block_14_depthwise (DepthwiseCo (None, 5, 5, 960)    8640        block_14_expand_rel
u[0][0]
__________________________________________________________________________________
_______________
block_14_depthwise_BN (BatchNor (None, 5, 5, 960)    3840        block_14_depthwise
[0][0]
__________________________________________________________________________________
_______________
block_14_depthwise_relu (ReLU)  (None, 5, 5, 960)    0           block_14_depthwise_
BN[0][0]
__________________________________________________________________________________
_______________
block_14_project (Conv2D)       (None, 5, 5, 160)    153600      block_14_depthwise_
relu[0][0]
__________________________________________________________________________________
_______________
block_14_project_BN (BatchNorma (None, 5, 5, 160)    640         block_14_project[0]
[0]
__________________________________________________________________________________
_______________
block_14_add (Add)              (None, 5, 5, 160)    0           block_13_project_BN
[0][0]
                                                                 block_14_project_BN
[0][0]
__________________________________________________________________________________
_______________
block_15_expand (Conv2D)        (None, 5, 5, 960)    153600      block_14_add[0][0]
__________________________________________________________________________________
_______________
block_15_expand_BN (BatchNormal (None, 5, 5, 960)    3840        block_15_expand[0]
[0]
__________________________________________________________________________________
_______________
block_15_expand_relu (ReLU)     (None, 5, 5, 960)    0           block_15_expand_BN
[0][0]
__________________________________________________________________________________
_______________
block_15_depthwise (DepthwiseCo (None, 5, 5, 960)    8640        block_15_expand_rel
u[0][0]
__________________________________________________________________________________
_______________
block_15_depthwise_BN (BatchNor (None, 5, 5, 960)    3840        block_15_depthwise
[0][0]
__________________________________________________________________________________
_______________
```

```
block_15_depthwise_relu (ReLU)  (None, 5, 5, 960)    0           block_15_depthwise_
BN[0][0]
__________________________________________________________________________________
_______________
block_15_project (Conv2D)       (None, 5, 5, 160)    153600      block_15_depthwise_
relu[0][0]
__________________________________________________________________________________
_______________
block_15_project_BN (BatchNorma (None, 5, 5, 160)    640         block_15_project[0]
[0]
__________________________________________________________________________________
_______________
block_15_add (Add)              (None, 5, 5, 160)    0           block_14_add[0][0]
                                                                 block_15_project_BN
[0][0]
__________________________________________________________________________________
_______________
block_16_expand (Conv2D)        (None, 5, 5, 960)    153600      block_15_add[0][0]
__________________________________________________________________________________
_______________
block_16_expand_BN (BatchNormal (None, 5, 5, 960)    3840        block_16_expand[0]
[0]
__________________________________________________________________________________
_______________
block_16_expand_relu (ReLU)     (None, 5, 5, 960)    0           block_16_expand_BN
[0][0]
__________________________________________________________________________________
_______________
block_16_depthwise (DepthwiseCo (None, 5, 5, 960)    8640        block_16_expand_rel
u[0][0]
__________________________________________________________________________________
_______________
block_16_depthwise_BN (BatchNor (None, 5, 5, 960)    3840        block_16_depthwise
[0][0]
__________________________________________________________________________________
_______________
block_16_depthwise_relu (ReLU)  (None, 5, 5, 960)    0           block_16_depthwise_
BN[0][0]
__________________________________________________________________________________
_______________
block_16_project (Conv2D)       (None, 5, 5, 320)    307200      block_16_depthwise_
relu[0][0]
__________________________________________________________________________________
_______________
block_16_project_BN (BatchNorma (None, 5, 5, 320)    1280        block_16_project[0]
[0]
__________________________________________________________________________________
_______________
Conv_1 (Conv2D)                 (None, 5, 5, 1280)   409600      block_16_project_BN
[0][0]
__________________________________________________________________________________
_______________
Conv_1_bn (BatchNormalization)  (None, 5, 5, 1280)   5120        Conv_1[0][0]
__________________________________________________________________________________
_______________
out_relu (ReLU)                 (None, 5, 5, 1280)   0           Conv_1_bn[0][0]
==================================================================================
==============
Total params: 2,257,984
Trainable params: 0
Non-trainable params: 2,257,984
__________________________________________________________________________________
_______________
```

In [64]:

```
#global_average_layer
global_average_layer = tf.keras.layers.GlobalAveragePooling2D()
feature_batch_average = global_average_layer(feature_batch)
print(feature_batch_average.shape)
```

```
(32, 1280)
```

In [ ]:
```
#output Layer
prediction_layer = tf.keras.layers.Dense(135)
```

In [66]:
```
#model Bulding
inputs = tf.keras.Input(shape=(160, 160, 3))
x = data_augmentation(inputs)
x = preprocess_input(x)
x = base_model(x, training=False)
x = global_average_layer(x)

x =tf.keras.layers.Dense(1024, activation='relu')(x)
x = tf.keras.layers.Dropout(0.2)(x)
x = tf.keras.layers.Dense(1024, activation='relu')(x)
x = tf.keras.layers.Dropout(0.2)(x)
x = tf.keras.layers.Dense(1280, activation='relu')(x)

x = tf.keras.layers.Dropout(0.2)(x)
outputs = prediction_layer(x)
model = tf.keras.Model(inputs, outputs)
base_learning_rate = 0.0001
model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=base_learning_rate),
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])
```

In [68]:
```
model.summary()
```

```
Model: "model_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_4 (InputLayer)         [(None, 160, 160, 3)]     0
_________________________________________________________________
sequential_1 (Sequential)    (None, 160, 160, 3)       0
_________________________________________________________________
tf.math.truediv_1 (TFOpLambd (None, 160, 160, 3)       0
_________________________________________________________________
tf.math.subtract_1 (TFOpLamb (None, 160, 160, 3)       0
_________________________________________________________________
mobilenetv2_1.00_160 (Functi (None, 5, 5, 1280)        2257984
_________________________________________________________________
global_average_pooling2d_1 ( (None, 1280)              0
_________________________________________________________________
dense_5 (Dense)              (None, 1024)              1311744
_________________________________________________________________
dropout_3 (Dropout)          (None, 1024)              0
_________________________________________________________________
dense_6 (Dense)              (None, 1024)              1049600
_________________________________________________________________
dropout_4 (Dropout)          (None, 1024)              0
_________________________________________________________________
dense_7 (Dense)              (None, 1280)              1312000
_________________________________________________________________
dropout_5 (Dropout)          (None, 1280)              0
_________________________________________________________________
dense_4 (Dense)              (None, 135)               172935
=================================================================
Total params: 6,104,263
Trainable params: 3,846,279
Non-trainable params: 2,257,984
_________________________________________________________________
```

In [69]:
```
len(model.trainable_variables)
```

Out[69]: 8

In [70]:
```
initial_epochs = 100

loss0, accuracy0 = model.evaluate(validation_dataset)
```

```
22/22 [==============================] - 63s 797ms/step - loss: 4.9972 - accuracy:
0.0000e+00
```

In [71]:
```
print("initial loss: {:.2f}".format(loss0))
print("initial accuracy: {:.2f}".format(accuracy0))
```

```
initial loss: 5.00
initial accuracy: 0.00
```

In [ ]:
```
from tensordash.tensordash import Tensordash
histories = Tensordash(
    ModelName = 'Mobilenet',
    email = 'harshabolla@gmail.com',
    password = '*******')
```

In [ ]:
```
#callbacks
from keras.callbacks import ModelCheckpoint, EarlyStopping
checkpoint1 = ModelCheckpoint("mobilenet.h5", monitor='val_acc', verbose=1, save_bes
early1 = EarlyStopping(monitor='val_acc', min_delta=0, patience=20, verbose=1, mode=
```

In [73]:
```
#fiting the model
history = model.fit(train_dataset,validation_data=validation_dataset,
                    epochs=initial_epochs,callbacks=[checkpoint])
```

```
Epoch 1/100
102/102 [==============================] - 148s 1s/step - loss: 4.9504 - accuracy:
0.0098 - val_loss: 4.8157 - val_accuracy: 0.0185

Epoch 00001: val_accuracy improved from -inf to 0.01847, saving model to mobilenet.h
5
C:\Users\harsha.teja\Anaconda3\envs\deeplearning\lib\site-packages\keras\utils\gener
ic_utils.py:497: CustomMaskWarning: Custom mask layers require a config and must ove
rride get_config. When loading, the custom mask layer must be passed to the custom_o
bjects argument.
  category=CustomMaskWarning)
Epoch 2/100
102/102 [==============================] - 120s 1s/step - loss: 4.7948 - accuracy:
0.0251 - val_loss: 4.6445 - val_accuracy: 0.0668

Epoch 00002: val_accuracy improved from 0.01847 to 0.06676, saving model to mobilene
t.h5
Epoch 3/100
102/102 [==============================] - 88s 785ms/step - loss: 4.6233 - accuracy:
0.0433 - val_loss: 4.4078 - val_accuracy: 0.0793

Epoch 00003: val_accuracy improved from 0.06676 to 0.07925, saving model to mobilene
t.h5
Epoch 4/100
102/102 [==============================] - 87s 781ms/step - loss: 4.4391 - accuracy:
0.0535 - val_loss: 4.1674 - val_accuracy: 0.0909

Epoch 00004: val_accuracy improved from 0.07925 to 0.09091, saving model to mobilene
t.h5
Epoch 5/100
102/102 [==============================] - 89s 804ms/step - loss: 4.2452 - accuracy:
0.0756 - val_loss: 4.0039 - val_accuracy: 0.1236

Epoch 00005: val_accuracy improved from 0.09091 to 0.12358, saving model to mobilene
t.h5
Epoch 6/100
102/102 [==============================] - 89s 795ms/step - loss: 4.1094 - accuracy:
0.0950 - val_loss: 3.8330 - val_accuracy: 0.1671
```

```
Epoch 00006: val_accuracy improved from 0.12358 to 0.16715, saving model to mobilene
t.h5
Epoch 7/100
102/102 [==============================] - 91s 816ms/step - loss: 3.9690 - accuracy:
0.1170 - val_loss: 3.6761 - val_accuracy: 0.1705

Epoch 00007: val_accuracy improved from 0.16715 to 0.17045, saving model to mobilene
t.h5
Epoch 8/100
102/102 [==============================] - 92s 824ms/step - loss: 3.8311 - accuracy:
0.1369 - val_loss: 3.5984 - val_accuracy: 0.1903

Epoch 00008: val_accuracy improved from 0.17045 to 0.19034, saving model to mobilene
t.h5
Epoch 9/100
102/102 [==============================] - 91s 808ms/step - loss: 3.7434 - accuracy:
0.1550 - val_loss: 3.4777 - val_accuracy: 0.2219

Epoch 00009: val_accuracy improved from 0.19034 to 0.22190, saving model to mobilene
t.h5
Epoch 10/100
102/102 [==============================] - 92s 831ms/step - loss: 3.6015 - accuracy:
0.1721 - val_loss: 3.3282 - val_accuracy: 0.2571

Epoch 00010: val_accuracy improved from 0.22190 to 0.25710, saving model to mobilene
t.h5
Epoch 11/100
102/102 [==============================] - 91s 814ms/step - loss: 3.4910 - accuracy:
0.1773 - val_loss: 3.2028 - val_accuracy: 0.2812

Epoch 00011: val_accuracy improved from 0.25710 to 0.28125, saving model to mobilene
t.h5
Epoch 12/100
102/102 [==============================] - 92s 825ms/step - loss: 3.3910 - accuracy:
0.2087 - val_loss: 3.0634 - val_accuracy: 0.3153

Epoch 00012: val_accuracy improved from 0.28125 to 0.31534, saving model to mobilene
t.h5
Epoch 13/100
102/102 [==============================] - 92s 825ms/step - loss: 3.2864 - accuracy:
0.2256 - val_loss: 3.0210 - val_accuracy: 0.3224

Epoch 00013: val_accuracy improved from 0.31534 to 0.32244, saving model to mobilene
t.h5
Epoch 14/100
102/102 [==============================] - 92s 822ms/step - loss: 3.2437 - accuracy:
0.2420 - val_loss: 2.8835 - val_accuracy: 0.3239

Epoch 00014: val_accuracy improved from 0.32244 to 0.32386, saving model to mobilene
t.h5
Epoch 15/100
102/102 [==============================] - 92s 828ms/step - loss: 3.0881 - accuracy:
0.2784 - val_loss: 2.8801 - val_accuracy: 0.3199

Epoch 00015: val_accuracy did not improve from 0.32386
Epoch 16/100
102/102 [==============================] - 92s 825ms/step - loss: 3.0072 - accuracy:
0.2806 - val_loss: 2.7947 - val_accuracy: 0.3501

Epoch 00016: val_accuracy improved from 0.32386 to 0.35014, saving model to mobilene
t.h5
Epoch 17/100
102/102 [==============================] - 94s 847ms/step - loss: 2.9274 - accuracy:
0.2876 - val_loss: 2.5757 - val_accuracy: 0.3804

Epoch 00017: val_accuracy improved from 0.35014 to 0.38040, saving model to mobilene
t.h5
Epoch 18/100
```

```
102/102 [==============================] - 98s 888ms/step - loss: 2.8487 - accuracy:
0.3159 - val_loss: 2.4462 - val_accuracy: 0.4418

Epoch 00018: val_accuracy improved from 0.38040 to 0.44176, saving model to mobilene
t.h5
Epoch 19/100
102/102 [==============================] - 94s 850ms/step - loss: 2.7716 - accuracy:
0.3254 - val_loss: 2.4643 - val_accuracy: 0.4219

Epoch 00019: val_accuracy did not improve from 0.44176
Epoch 20/100
102/102 [==============================] - 119s 1s/step - loss: 2.6800 - accuracy:
0.3442 - val_loss: 2.3407 - val_accuracy: 0.4531

Epoch 00020: val_accuracy improved from 0.44176 to 0.45312, saving model to mobilene
t.h5
Epoch 21/100
102/102 [==============================] - 118s 1s/step - loss: 2.5762 - accuracy:
0.3679 - val_loss: 2.3920 - val_accuracy: 0.3991

Epoch 00021: val_accuracy did not improve from 0.45312
Epoch 22/100
102/102 [==============================] - 112s 979ms/step - loss: 2.5012 - accurac
y: 0.3958 - val_loss: 2.2823 - val_accuracy: 0.4531

Epoch 00022: val_accuracy did not improve from 0.45312
Epoch 23/100
102/102 [==============================] - 107s 961ms/step - loss: 2.4635 - accurac
y: 0.3949 - val_loss: 2.2116 - val_accuracy: 0.4602

Epoch 00023: val_accuracy improved from 0.45312 to 0.46023, saving model to mobilene
t.h5
Epoch 24/100
102/102 [==============================] - 110s 996ms/step - loss: 2.3951 - accurac
y: 0.4100 - val_loss: 2.1704 - val_accuracy: 0.4801

Epoch 00024: val_accuracy improved from 0.46023 to 0.48011, saving model to mobilene
t.h5
Epoch 25/100
102/102 [==============================] - 92s 838ms/step - loss: 2.3653 - accuracy:
0.4222 - val_loss: 2.1239 - val_accuracy: 0.4886

Epoch 00025: val_accuracy improved from 0.48011 to 0.48864, saving model to mobilene
t.h5
Epoch 26/100
102/102 [==============================] - 120s 1s/step - loss: 2.2899 - accuracy:
0.4275 - val_loss: 2.0230 - val_accuracy: 0.5213

Epoch 00026: val_accuracy improved from 0.48864 to 0.52131, saving model to mobilene
t.h5
Epoch 27/100
102/102 [==============================] - 106s 974ms/step - loss: 2.1790 - accurac
y: 0.4586 - val_loss: 1.9968 - val_accuracy: 0.5099

Epoch 00027: val_accuracy did not improve from 0.52131
Epoch 28/100
102/102 [==============================] - 116s 1s/step - loss: 2.1407 - accuracy:
0.4614 - val_loss: 1.8750 - val_accuracy: 0.5724

Epoch 00028: val_accuracy improved from 0.52131 to 0.57244, saving model to mobilene
t.h5
Epoch 29/100
102/102 [==============================] - 110s 987ms/step - loss: 2.0647 - accurac
y: 0.4822 - val_loss: 1.7997 - val_accuracy: 0.5668

Epoch 00029: val_accuracy did not improve from 0.57244
Epoch 30/100
102/102 [==============================] - 104s 920ms/step - loss: 2.0133 - accurac
y: 0.5003 - val_loss: 1.8143 - val_accuracy: 0.5634
```

```
Epoch 00030: val_accuracy did not improve from 0.57244
Epoch 31/100
102/102 [==============================] - 108s 949ms/step - loss: 1.9582 - accurac
y: 0.4948 - val_loss: 1.7461 - val_accuracy: 0.5668

Epoch 00031: val_accuracy did not improve from 0.57244
Epoch 32/100
102/102 [==============================] - 104s 917ms/step - loss: 1.9183 - accurac
y: 0.5025 - val_loss: 1.7282 - val_accuracy: 0.5850

Epoch 00032: val_accuracy improved from 0.57244 to 0.58501, saving model to mobilene
t.h5
Epoch 33/100
102/102 [==============================] - 111s 977ms/step - loss: 1.8517 - accurac
y: 0.5239 - val_loss: 1.7140 - val_accuracy: 0.5696

Epoch 00033: val_accuracy did not improve from 0.58501
Epoch 34/100
102/102 [==============================] - 99s 878ms/step - loss: 1.8000 - accuracy:
0.5375 - val_loss: 1.5576 - val_accuracy: 0.6193

Epoch 00034: val_accuracy improved from 0.58501 to 0.61932, saving model to mobilene
t.h5
Epoch 35/100
102/102 [==============================] - 102s 899ms/step - loss: 1.7705 - accurac
y: 0.5544 - val_loss: 1.6192 - val_accuracy: 0.5952

Epoch 00035: val_accuracy did not improve from 0.61932
Epoch 36/100
102/102 [==============================] - 105s 939ms/step - loss: 1.7197 - accurac
y: 0.5627 - val_loss: 1.5239 - val_accuracy: 0.6179

Epoch 00036: val_accuracy did not improve from 0.61932
Epoch 37/100
102/102 [==============================] - 106s 964ms/step - loss: 1.6965 - accurac
y: 0.5686 - val_loss: 1.4967 - val_accuracy: 0.6499

Epoch 00037: val_accuracy improved from 0.61932 to 0.64986, saving model to mobilene
t.h5
Epoch 38/100
102/102 [==============================] - 112s 1s/step - loss: 1.6256 - accuracy:
0.5843 - val_loss: 1.4120 - val_accuracy: 0.6690

Epoch 00038: val_accuracy improved from 0.64986 to 0.66903, saving model to mobilene
t.h5
Epoch 39/100
102/102 [==============================] - 124s 1s/step - loss: 1.5449 - accuracy:
0.5947 - val_loss: 1.3649 - val_accuracy: 0.6662

Epoch 00039: val_accuracy did not improve from 0.66903
Epoch 40/100
102/102 [==============================] - 109s 985ms/step - loss: 1.4960 - accurac
y: 0.6088 - val_loss: 1.3223 - val_accuracy: 0.6761

Epoch 00040: val_accuracy improved from 0.66903 to 0.67614, saving model to mobilene
t.h5
Epoch 41/100
102/102 [==============================] - 89s 803ms/step - loss: 1.4609 - accuracy:
0.6173 - val_loss: 1.3603 - val_accuracy: 0.6747

Epoch 00041: val_accuracy did not improve from 0.67614
Epoch 42/100
102/102 [==============================] - 111s 1s/step - loss: 1.4396 - accuracy:
0.6242 - val_loss: 1.3150 - val_accuracy: 0.6747

Epoch 00042: val_accuracy did not improve from 0.67614
Epoch 43/100
102/102 [==============================] - 105s 955ms/step - loss: 1.4116 - accurac
```

```
y: 0.6320 - val_loss: 1.3397 - val_accuracy: 0.6548

Epoch 00043: val_accuracy did not improve from 0.67614
Epoch 44/100
102/102 [==============================] - 105s 931ms/step - loss: 1.3942 - accurac
y: 0.6358 - val_loss: 1.2194 - val_accuracy: 0.7003

Epoch 00044: val_accuracy improved from 0.67614 to 0.70029, saving model to mobilene
t.h5
Epoch 45/100
102/102 [==============================] - 100s 902ms/step - loss: 1.3427 - accurac
y: 0.6484 - val_loss: 1.2499 - val_accuracy: 0.6818

Epoch 00045: val_accuracy did not improve from 0.70029
Epoch 46/100
102/102 [==============================] - 100s 903ms/step - loss: 1.3303 - accurac
y: 0.6515 - val_loss: 1.1790 - val_accuracy: 0.7074

Epoch 00046: val_accuracy improved from 0.70029 to 0.70739, saving model to mobilene
t.h5
Epoch 47/100
102/102 [==============================] - 100s 896ms/step - loss: 1.2710 - accurac
y: 0.6678 - val_loss: 1.1018 - val_accuracy: 0.7372

Epoch 00047: val_accuracy improved from 0.70739 to 0.73722, saving model to mobilene
t.h5
Epoch 48/100
102/102 [==============================] - 99s 889ms/step - loss: 1.2596 - accuracy:
0.6653 - val_loss: 1.0683 - val_accuracy: 0.7330

Epoch 00048: val_accuracy did not improve from 0.73722
Epoch 49/100
102/102 [==============================] - 105s 962ms/step - loss: 1.2115 - accurac
y: 0.6875 - val_loss: 1.0363 - val_accuracy: 0.7522

Epoch 00049: val_accuracy improved from 0.73722 to 0.75216, saving model to mobilene
t.h5
Epoch 50/100
102/102 [==============================] - 97s 874ms/step - loss: 1.1621 - accuracy:
0.6911 - val_loss: 1.0409 - val_accuracy: 0.7550

Epoch 00050: val_accuracy improved from 0.75216 to 0.75504, saving model to mobilene
t.h5
Epoch 51/100
102/102 [==============================] - 104s 921ms/step - loss: 1.1409 - accurac
y: 0.6998 - val_loss: 1.0289 - val_accuracy: 0.7543

Epoch 00051: val_accuracy did not improve from 0.75504
Epoch 52/100
102/102 [==============================] - 94s 833ms/step - loss: 1.1129 - accuracy:
0.7068 - val_loss: 1.0066 - val_accuracy: 0.7514

Epoch 00052: val_accuracy did not improve from 0.75504
Epoch 53/100
102/102 [==============================] - 98s 874ms/step - loss: 1.0729 - accuracy:
0.7135 - val_loss: 0.9834 - val_accuracy: 0.7642

Epoch 00053: val_accuracy improved from 0.75504 to 0.76420, saving model to mobilene
t.h5
Epoch 54/100
102/102 [==============================] - 106s 969ms/step - loss: 1.0247 - accurac
y: 0.7357 - val_loss: 1.0118 - val_accuracy: 0.7557

Epoch 00054: val_accuracy did not improve from 0.76420
Epoch 55/100
102/102 [==============================] - 101s 918ms/step - loss: 1.0599 - accurac
y: 0.7074 - val_loss: 0.9706 - val_accuracy: 0.7614

Epoch 00055: val_accuracy did not improve from 0.76420
```

```
Epoch 56/100
102/102 [==============================] - 97s 878ms/step - loss: 1.0241 - accuracy:
0.7253 - val_loss: 0.9577 - val_accuracy: 0.7723

Epoch 00056: val_accuracy improved from 0.76420 to 0.77233, saving model to mobilene
t.h5
Epoch 57/100
102/102 [==============================] - 103s 919ms/step - loss: 1.0078 - accurac
y: 0.7325 - val_loss: 0.8715 - val_accuracy: 0.7812

Epoch 00057: val_accuracy improved from 0.77233 to 0.78125, saving model to mobilene
t.h5
Epoch 58/100
102/102 [==============================] - 94s 839ms/step - loss: 0.9514 - accuracy:
0.7396 - val_loss: 0.8524 - val_accuracy: 0.7827

Epoch 00058: val_accuracy improved from 0.78125 to 0.78267, saving model to mobilene
t.h5
Epoch 59/100
102/102 [==============================] - 97s 861ms/step - loss: 0.9404 - accuracy:
0.7483 - val_loss: 0.8429 - val_accuracy: 0.7770

Epoch 00059: val_accuracy did not improve from 0.78267
Epoch 60/100
102/102 [==============================] - 98s 878ms/step - loss: 0.9060 - accuracy:
0.7600 - val_loss: 0.8964 - val_accuracy: 0.7756

Epoch 00060: val_accuracy did not improve from 0.78267
Epoch 61/100
102/102 [==============================] - 100s 903ms/step - loss: 0.8740 - accurac
y: 0.7619 - val_loss: 0.7978 - val_accuracy: 0.7884

Epoch 00061: val_accuracy improved from 0.78267 to 0.78835, saving model to mobilene
t.h5
Epoch 62/100
102/102 [==============================] - 92s 827ms/step - loss: 0.8328 - accuracy:
0.7778 - val_loss: 0.8365 - val_accuracy: 0.7770

Epoch 00062: val_accuracy did not improve from 0.78835
Epoch 63/100
102/102 [==============================] - 99s 883ms/step - loss: 0.8411 - accuracy:
0.7696 - val_loss: 0.6549 - val_accuracy: 0.8423

Epoch 00063: val_accuracy improved from 0.78835 to 0.84233, saving model to mobilene
t.h5
Epoch 64/100
102/102 [==============================] - 97s 872ms/step - loss: 0.8639 - accuracy:
0.7650 - val_loss: 0.7470 - val_accuracy: 0.8097

Epoch 00064: val_accuracy did not improve from 0.84233
Epoch 65/100
102/102 [==============================] - 97s 874ms/step - loss: 0.8218 - accuracy:
0.7763 - val_loss: 0.7883 - val_accuracy: 0.8168

Epoch 00065: val_accuracy did not improve from 0.84233
Epoch 66/100
102/102 [==============================] - 100s 900ms/step - loss: 0.7841 - accurac
y: 0.7855 - val_loss: 0.7389 - val_accuracy: 0.8111

Epoch 00066: val_accuracy did not improve from 0.84233
Epoch 67/100
102/102 [==============================] - 93s 830ms/step - loss: 0.7832 - accuracy:
0.7926 - val_loss: 0.7281 - val_accuracy: 0.8228

Epoch 00067: val_accuracy did not improve from 0.84233
Epoch 68/100
102/102 [==============================] - 97s 868ms/step - loss: 0.7463 - accuracy:
0.7904 - val_loss: 0.7341 - val_accuracy: 0.8153
```

```
Epoch 00068: val_accuracy did not improve from 0.84233
Epoch 69/100
102/102 [==============================] - 99s 880ms/step - loss: 0.7082 - accuracy:
0.8089 - val_loss: 0.6635 - val_accuracy: 0.8381

Epoch 00069: val_accuracy did not improve from 0.84233
Epoch 70/100
102/102 [==============================] - 98s 863ms/step - loss: 0.7409 - accuracy:
0.7938 - val_loss: 0.6491 - val_accuracy: 0.8310

Epoch 00070: val_accuracy did not improve from 0.84233
Epoch 71/100
102/102 [==============================] - 98s 881ms/step - loss: 0.7124 - accuracy:
0.8036 - val_loss: 0.6194 - val_accuracy: 0.8438

Epoch 00071: val_accuracy improved from 0.84233 to 0.84375, saving model to mobilene
t.h5
Epoch 72/100
102/102 [==============================] - 99s 899ms/step - loss: 0.6923 - accuracy:
0.8101 - val_loss: 0.6369 - val_accuracy: 0.8551

Epoch 00072: val_accuracy improved from 0.84375 to 0.85511, saving model to mobilene
t.h5
Epoch 73/100
102/102 [==============================] - 94s 838ms/step - loss: 0.6737 - accuracy:
0.8021 - val_loss: 0.6602 - val_accuracy: 0.8395

Epoch 00073: val_accuracy did not improve from 0.85511
Epoch 74/100
102/102 [==============================] - 98s 870ms/step - loss: 0.6506 - accuracy:
0.8239 - val_loss: 0.6621 - val_accuracy: 0.8338

Epoch 00074: val_accuracy did not improve from 0.85511
Epoch 75/100
102/102 [==============================] - 101s 919ms/step - loss: 0.6474 - accurac
y: 0.8254 - val_loss: 0.5204 - val_accuracy: 0.8707

Epoch 00075: val_accuracy improved from 0.85511 to 0.87074, saving model to mobilene
t.h5
Epoch 76/100
102/102 [==============================] - 96s 879ms/step - loss: 0.6218 - accuracy:
0.8279 - val_loss: 0.5136 - val_accuracy: 0.8835

Epoch 00076: val_accuracy improved from 0.87074 to 0.88352, saving model to mobilene
t.h5
Epoch 77/100
102/102 [==============================] - 97s 880ms/step - loss: 0.6093 - accuracy:
0.8344 - val_loss: 0.5895 - val_accuracy: 0.8551

Epoch 00077: val_accuracy did not improve from 0.88352
Epoch 78/100
102/102 [==============================] - 96s 867ms/step - loss: 0.6085 - accuracy:
0.8318 - val_loss: 0.5297 - val_accuracy: 0.8736

Epoch 00078: val_accuracy did not improve from 0.88352
Epoch 79/100
102/102 [==============================] - 95s 846ms/step - loss: 0.5801 - accuracy:
0.8441 - val_loss: 0.5407 - val_accuracy: 0.8565

Epoch 00079: val_accuracy did not improve from 0.88352
Epoch 80/100
102/102 [==============================] - 99s 883ms/step - loss: 0.5458 - accuracy:
0.8548 - val_loss: 0.4738 - val_accuracy: 0.8821

Epoch 00080: val_accuracy did not improve from 0.88352
Epoch 81/100
102/102 [==============================] - 97s 877ms/step - loss: 0.5484 - accuracy:
0.8494 - val_loss: 0.4569 - val_accuracy: 0.8864
```

```
Epoch 00081: val_accuracy improved from 0.88352 to 0.88636, saving model to mobilene
t.h5
Epoch 82/100
102/102 [==============================] - 96s 865ms/step - loss: 0.5204 - accuracy:
0.8560 - val_loss: 0.5027 - val_accuracy: 0.8750

Epoch 00082: val_accuracy did not improve from 0.88636
Epoch 83/100
102/102 [==============================] - 99s 899ms/step - loss: 0.5235 - accuracy:
0.8559 - val_loss: 0.4762 - val_accuracy: 0.8778

Epoch 00083: val_accuracy did not improve from 0.88636
Epoch 84/100
102/102 [==============================] - 96s 869ms/step - loss: 0.5060 - accuracy:
0.8562 - val_loss: 0.4658 - val_accuracy: 0.8807

Epoch 00084: val_accuracy did not improve from 0.88636
Epoch 85/100
102/102 [==============================] - 95s 828ms/step - loss: 0.5077 - accuracy:
0.8642 - val_loss: 0.4446 - val_accuracy: 0.8876

Epoch 00085: val_accuracy improved from 0.88636 to 0.88761, saving model to mobilene
t.h5
Epoch 86/100
102/102 [==============================] - 98s 877ms/step - loss: 0.5301 - accuracy:
0.8542 - val_loss: 0.4834 - val_accuracy: 0.8679

Epoch 00086: val_accuracy did not improve from 0.88761
Epoch 87/100
102/102 [==============================] - 98s 878ms/step - loss: 0.4901 - accuracy:
0.8654 - val_loss: 0.4603 - val_accuracy: 0.8949

Epoch 00087: val_accuracy improved from 0.88761 to 0.89489, saving model to mobilene
t.h5
Epoch 88/100
102/102 [==============================] - 95s 843ms/step - loss: 0.4761 - accuracy:
0.8651 - val_loss: 0.4297 - val_accuracy: 0.8963

Epoch 00088: val_accuracy improved from 0.89489 to 0.89631, saving model to mobilene
t.h5
Epoch 89/100
102/102 [==============================] - 101s 910ms/step - loss: 0.4561 - accurac
y: 0.8735 - val_loss: 0.3602 - val_accuracy: 0.9219

Epoch 00089: val_accuracy improved from 0.89631 to 0.92188, saving model to mobilene
t.h5
Epoch 90/100
102/102 [==============================] - 95s 861ms/step - loss: 0.4507 - accuracy:
0.8737 - val_loss: 0.4134 - val_accuracy: 0.8963

Epoch 00090: val_accuracy did not improve from 0.92188
Epoch 91/100
102/102 [==============================] - 96s 852ms/step - loss: 0.4985 - accuracy:
0.8597 - val_loss: 0.4457 - val_accuracy: 0.8949

Epoch 00091: val_accuracy did not improve from 0.92188
Epoch 92/100
102/102 [==============================] - 96s 847ms/step - loss: 0.4337 - accuracy:
0.8789 - val_loss: 0.4812 - val_accuracy: 0.8878

Epoch 00092: val_accuracy did not improve from 0.92188
Epoch 93/100
102/102 [==============================] - 100s 898ms/step - loss: 0.4253 - accurac
y: 0.8832 - val_loss: 0.3843 - val_accuracy: 0.8963

Epoch 00093: val_accuracy did not improve from 0.92188
Epoch 94/100
102/102 [==============================] - 102s 924ms/step - loss: 0.4416 - accurac
y: 0.8771 - val_loss: 0.3805 - val_accuracy: 0.9077
```

```
Epoch 00094: val_accuracy did not improve from 0.92188
Epoch 95/100
102/102 [==============================] - 96s 867ms/step - loss: 0.4040 - accuracy:
0.8841 - val_loss: 0.3835 - val_accuracy: 0.9048

Epoch 00095: val_accuracy did not improve from 0.92188
Epoch 96/100
102/102 [==============================] - 97s 873ms/step - loss: 0.4405 - accuracy:
0.8778 - val_loss: 0.3956 - val_accuracy: 0.9105

Epoch 00096: val_accuracy did not improve from 0.92188
Epoch 97/100
102/102 [==============================] - 102s 915ms/step - loss: 0.4054 - accurac
y: 0.8909 - val_loss: 0.3380 - val_accuracy: 0.9347

Epoch 00097: val_accuracy improved from 0.92188 to 0.93466, saving model to mobilene
t.h5
Epoch 98/100
102/102 [==============================] - 97s 877ms/step - loss: 0.3771 - accuracy:
0.8986 - val_loss: 0.3752 - val_accuracy: 0.9091

Epoch 00098: val_accuracy did not improve from 0.93466
Epoch 99/100
102/102 [==============================] - 97s 876ms/step - loss: 0.3797 - accuracy:
0.8943 - val_loss: 0.3469 - val_accuracy: 0.9332

Epoch 00099: val_accuracy did not improve from 0.93466
Epoch 100/100
102/102 [==============================] - 96s 870ms/step - loss: 0.3713 - accuracy:
0.8924 - val_loss: 0.3541 - val_accuracy: 0.9105

Epoch 00100: val_accuracy did not improve from 0.93466
```

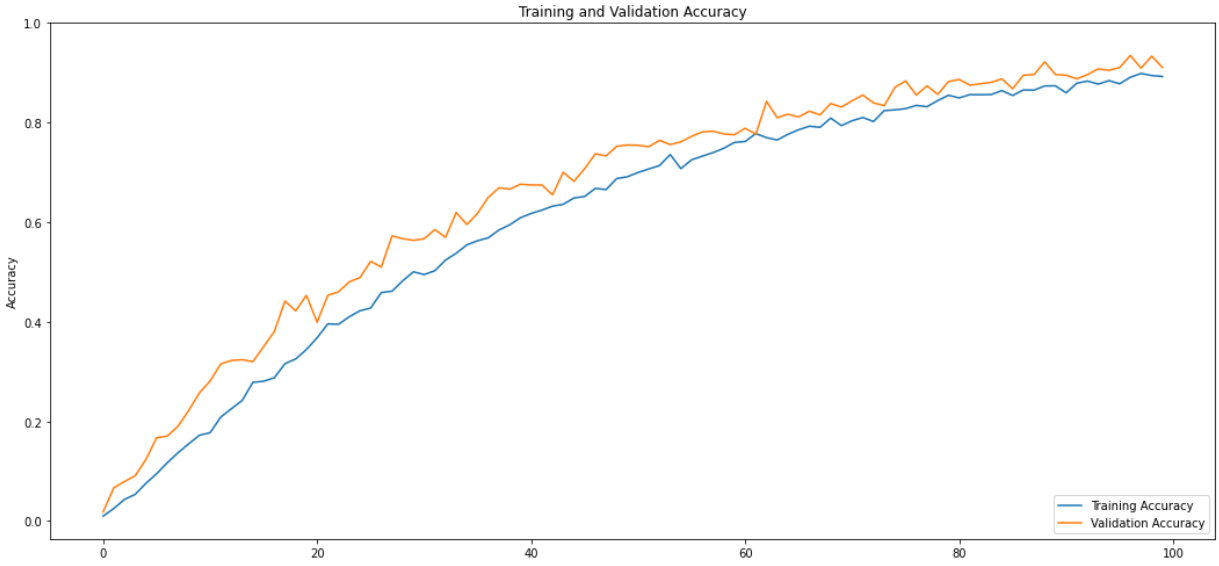
history = model.fit(train_dataset,validation_data=validation_dataset, epochs=initial_epochs, steps_per_epoch=len(train_dataset) // 32, validation_steps=len(validation_dataset) // 32, callbacks=[checkpoint,early])

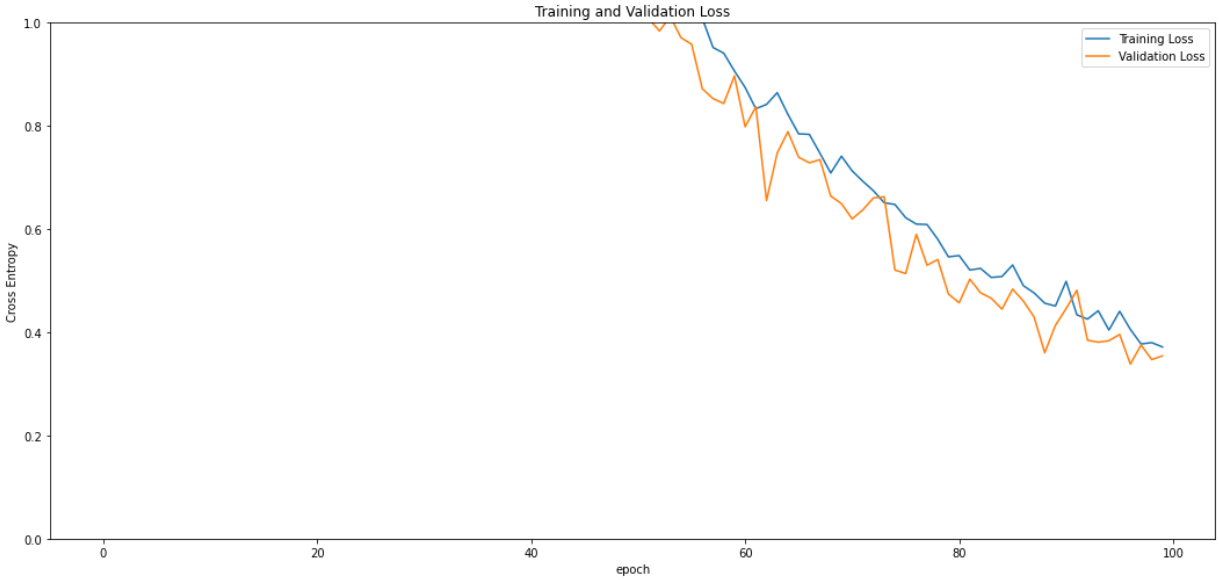
In [74]:
```
history.history.keys()
```

Out[74]:
```
dict_keys(['loss', 'accuracy', 'val_loss', 'val_accuracy'])
```

In [75]:
```
#ploting the results
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']

loss = history.history['loss']
val_loss = history.history['val_loss']

plt.figure(figsize=(18, 18))
plt.subplot(2, 1, 1)
plt.plot(acc, label='Training Accuracy')
plt.plot(val_acc, label='Validation Accuracy')
plt.legend(loc='lower right')
plt.ylabel('Accuracy')
plt.ylim([min(plt.ylim()),1])
plt.title('Training and Validation Accuracy')

plt.subplot(2, 1, 2)
plt.plot(loss, label='Training Loss')
plt.plot(val_loss, label='Validation Loss')
plt.legend(loc='upper right')
plt.ylabel('Cross Entropy')
plt.ylim([0,1.0])
plt.title('Training and Validation Loss')
```

```
plt.xlabel('epoch')
plt.show()
```





In [77]:
```
#fine Tuning the model
base_model.trainable = True
# Let's take a look to see how many layers are in the base model
print("Number of layers in the base model: ", len(base_model.layers))

# Fine-tune from this layer onwards
fine_tune_at = 100

# Freeze all the layers before the `fine_tune_at` layer
for layer in base_model.layers[:fine_tune_at]:
  layer.trainable =  False
```

```
Number of layers in the base model:  154
```

In [131…
```
#callbacks
from keras.callbacks import ModelCheckpoint, EarlyStopping
checkpoint1 = ModelCheckpoint("mobilenet_after_tune.h5", monitor='val_acc', verbose=
early1 = EarlyStopping(monitor='val_acc', min_delta=0, patience=20, verbose=1, mode=
```

In [106…
```
#initializing optimzer
base_learning_rate = 0.0001
model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=base_learning_rate/10
```

```
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])
```

In [108…

```
#fitting the model
fine_tune_epochs = 50
total_epochs =  initial_epochs + fine_tune_epochs

history_fine = model.fit(train_dataset,
                         epochs=total_epochs,
                         initial_epoch=history.epoch[-1],
                         validation_data=validation_dataset,callbacks=[checkpoint1])
```

```
Epoch 100/150
102/102 [==============================] - 120s 1s/step - loss: 0.1061 - accuracy:
0.9678 - val_loss: 0.0550 - val_accuracy: 0.9886
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 101/150
102/102 [==============================] - 123s 1s/step - loss: 0.1118 - accuracy:
0.9662 - val_loss: 0.0514 - val_accuracy: 0.9872
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 102/150
102/102 [==============================] - 122s 1s/step - loss: 0.1074 - accuracy:
0.9671 - val_loss: 0.0564 - val_accuracy: 0.9858
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 103/150
102/102 [==============================] - 126s 1s/step - loss: 0.1087 - accuracy:
0.9694 - val_loss: 0.0345 - val_accuracy: 0.9929
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 104/150
102/102 [==============================] - 125s 1s/step - loss: 0.1036 - accuracy:
0.9697 - val_loss: 0.0694 - val_accuracy: 0.9773
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 105/150
102/102 [==============================] - 126s 1s/step - loss: 0.1006 - accuracy:
0.9717 - val_loss: 0.0492 - val_accuracy: 0.9915
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 106/150
102/102 [==============================] - 129s 1s/step - loss: 0.0960 - accuracy:
0.9730 - val_loss: 0.0506 - val_accuracy: 0.9901
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 107/150
102/102 [==============================] - 129s 1s/step - loss: 0.1215 - accuracy:
0.9653 - val_loss: 0.0641 - val_accuracy: 0.9901
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 108/150
102/102 [==============================] - 132s 1s/step - loss: 0.0916 - accuracy:
0.9733 - val_loss: 0.0532 - val_accuracy: 0.9872
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 109/150
102/102 [==============================] - 129s 1s/step - loss: 0.0931 - accuracy:
0.9711 - val_loss: 0.0517 - val_accuracy: 0.9885
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 110/150
102/102 [==============================] - 128s 1s/step - loss: 0.0901 - accuracy:
0.9739 - val_loss: 0.0479 - val_accuracy: 0.9929
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 111/150
102/102 [==============================] - 134s 1s/step - loss: 0.0850 - accuracy:
0.9770 - val_loss: 0.0510 - val_accuracy: 0.9844
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 112/150
102/102 [==============================] - 128s 1s/step - loss: 0.0986 - accuracy:
0.9721 - val_loss: 0.0534 - val_accuracy: 0.9872
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 113/150
102/102 [==============================] - 129s 1s/step - loss: 0.0773 - accuracy:
0.9803 - val_loss: 0.0281 - val_accuracy: 0.9957
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
```

```
Epoch 114/150
102/102 [==============================] - 134s 1s/step - loss: 0.0872 - accuracy:
0.9720 - val_loss: 0.0387 - val_accuracy: 0.9929
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 115/150
102/102 [==============================] - 129s 1s/step - loss: 0.0952 - accuracy:
0.9720 - val_loss: 0.0474 - val_accuracy: 0.9901
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 116/150
102/102 [==============================] - 129s 1s/step - loss: 0.0821 - accuracy:
0.9776 - val_loss: 0.0470 - val_accuracy: 0.9886
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 117/150
102/102 [==============================] - 134s 1s/step - loss: 0.1003 - accuracy:
0.9711 - val_loss: 0.0498 - val_accuracy: 0.9885
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 118/150
102/102 [==============================] - 129s 1s/step - loss: 0.0855 - accuracy:
0.9748 - val_loss: 0.0467 - val_accuracy: 0.9885
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 119/150
102/102 [==============================] - 133s 1s/step - loss: 0.0788 - accuracy:
0.9795 - val_loss: 0.0309 - val_accuracy: 0.9957
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 120/150
102/102 [==============================] - 129s 1s/step - loss: 0.0852 - accuracy:
0.9745 - val_loss: 0.0323 - val_accuracy: 0.9957
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 121/150
102/102 [==============================] - 134s 1s/step - loss: 0.0834 - accuracy:
0.9776 - val_loss: 0.0278 - val_accuracy: 0.9972
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 122/150
102/102 [==============================] - 136s 1s/step - loss: 0.0957 - accuracy:
0.9714 - val_loss: 0.0395 - val_accuracy: 0.9957
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 123/150
102/102 [==============================] - 134s 1s/step - loss: 0.0783 - accuracy:
0.9797 - val_loss: 0.0343 - val_accuracy: 0.9886
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 124/150
102/102 [==============================] - 136s 1s/step - loss: 0.0809 - accuracy:
0.9766 - val_loss: 0.0318 - val_accuracy: 0.9929
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 125/150
102/102 [==============================] - 129s 1s/step - loss: 0.0843 - accuracy:
0.9751 - val_loss: 0.0477 - val_accuracy: 0.9858
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 126/150
102/102 [==============================] - 128s 1s/step - loss: 0.0808 - accuracy:
0.9776 - val_loss: 0.0266 - val_accuracy: 0.9957
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 127/150
102/102 [==============================] - 130s 1s/step - loss: 0.0866 - accuracy:
0.9739 - val_loss: 0.0410 - val_accuracy: 0.9872
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 128/150
102/102 [==============================] - 135s 1s/step - loss: 0.0809 - accuracy:
0.9785 - val_loss: 0.0405 - val_accuracy: 0.9943
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 129/150
102/102 [==============================] - 136s 1s/step - loss: 0.0704 - accuracy:
0.9816 - val_loss: 0.0266 - val_accuracy: 0.9957
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 130/150
102/102 [==============================] - 145s 1s/step - loss: 0.0656 - accuracy:
0.9816 - val_loss: 0.0214 - val_accuracy: 0.9957
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 131/150
```

```
102/102 [==============================] - 141s 1s/step - loss: 0.0732 - accuracy:
0.9794 - val_loss: 0.0226 - val_accuracy: 0.9957
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 132/150
102/102 [==============================] - 129s 1s/step - loss: 0.0739 - accuracy:
0.9767 - val_loss: 0.0293 - val_accuracy: 0.9943
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 133/150
102/102 [==============================] - 131s 1s/step - loss: 0.0775 - accuracy:
0.9776 - val_loss: 0.0351 - val_accuracy: 0.9899
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 134/150
102/102 [==============================] - 127s 1s/step - loss: 0.0774 - accuracy:
0.9794 - val_loss: 0.0450 - val_accuracy: 0.9914
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 135/150
102/102 [==============================] - 131s 1s/step - loss: 0.0723 - accuracy:
0.9801 - val_loss: 0.0348 - val_accuracy: 0.9929
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 136/150
102/102 [==============================] - 131s 1s/step - loss: 0.0674 - accuracy:
0.9825 - val_loss: 0.0240 - val_accuracy: 0.9957
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 137/150
102/102 [==============================] - 133s 1s/step - loss: 0.0720 - accuracy:
0.9801 - val_loss: 0.0251 - val_accuracy: 0.9942
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 138/150
102/102 [==============================] - 136s 1s/step - loss: 0.0728 - accuracy:
0.9785 - val_loss: 0.0254 - val_accuracy: 0.9943
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 139/150
102/102 [==============================] - 128s 1s/step - loss: 0.0834 - accuracy:
0.9736 - val_loss: 0.0222 - val_accuracy: 0.9972
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 140/150
102/102 [==============================] - 132s 1s/step - loss: 0.0725 - accuracy:
0.9819 - val_loss: 0.0236 - val_accuracy: 0.9929
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 141/150
102/102 [==============================] - 132s 1s/step - loss: 0.0712 - accuracy:
0.9791 - val_loss: 0.0180 - val_accuracy: 0.9972
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 142/150
102/102 [==============================] - 132s 1s/step - loss: 0.0550 - accuracy:
0.9862 - val_loss: 0.0235 - val_accuracy: 0.9972
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 143/150
102/102 [==============================] - 127s 1s/step - loss: 0.0726 - accuracy:
0.9776 - val_loss: 0.0284 - val_accuracy: 0.9943
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 144/150
102/102 [==============================] - 131s 1s/step - loss: 0.0674 - accuracy:
0.9822 - val_loss: 0.0239 - val_accuracy: 0.9957
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 145/150
102/102 [==============================] - 132s 1s/step - loss: 0.0612 - accuracy:
0.9849 - val_loss: 0.0217 - val_accuracy: 0.9972
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 146/150
102/102 [==============================] - 131s 1s/step - loss: 0.0647 - accuracy:
0.9838 - val_loss: 0.0260 - val_accuracy: 0.9943
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 147/150
102/102 [==============================] - 132s 1s/step - loss: 0.0722 - accuracy:
0.9809 - val_loss: 0.0258 - val_accuracy: 0.9957
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 148/150
102/102 [==============================] - 132s 1s/step - loss: 0.0687 - accuracy:
```

```
0.9806 - val_loss: 0.0232 - val_accuracy: 0.9972
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 149/150
102/102 [==============================] - 146s 1s/step - loss: 0.0673 - accuracy:
0.9779 - val_loss: 0.0290 - val_accuracy: 0.9972
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
Epoch 150/150
102/102 [==============================] - 174s 2s/step - loss: 0.0713 - accuracy:
0.9831 - val_loss: 0.0170 - val_accuracy: 0.9972
WARNING:tensorflow:Can save best model only with val_acc available, skipping.
```

In [123…
```
#calclating loss and accuracy on test dataset
loss, accuracy = model.evaluate(test_dataset)
print('Test accuracy :', accuracy)
```

```
23/23 [==============================] - 55s 601ms/step - loss: 0.0289 - accuracy:
0.9932
Test accuracy : 0.9932065010070801
```

In [164…
```
# make a prediction
predictions = model.predict(test_dataset, steps=len(validation_dataset), verbose=1)
```

```
22/22 [==============================] - 33s 817ms/step
```

In [125…
```
import time
t = time.time()

export_path = "/saved_models/{}".format(int(t))
model.save(export_path)

export_path
```

```
INFO:tensorflow:Assets written to: /saved_models/1634361181\assets
C:\Users\harsha.teja\Anaconda3\envs\deeplearning\lib\site-packages\keras\utils\gener
ic_utils.py:497: CustomMaskWarning: Custom mask layers require a config and must ove
rride get_config. When loading, the custom mask layer must be passed to the custom_o
bjects argument.
  category=CustomMaskWarning)
```

Out[125…
```
'/saved_models/1634361181'
```

In [11]:
```
#predicating on Test dataset
img_height, img_width = 160,160
def predict(model, img):
    img_array = tf.keras.preprocessing.image.img_to_array(images[i].numpy())
    img_array = tf.expand_dims(img_array, 0)

    predictions = model.predict(img_array)

    predicted_class = class_names[np.argmax(predictions[0])]
    #confidence = round(100 * (np.max(predictions[0])), 2)
    confidence = tf.nn.softmax(predictions[0])
    confidence = round(100 * (np.max(confidence)),2)
    return predicted_class, confidence
```

In [16]:
```
plt.figure(figsize=(15, 15))
for images, labels in test_dataset.take(10):
    for i in range(9):
        ax = plt.subplot(3, 3, i + 1)
        plt.imshow(images[i].numpy().astype("uint8"))

        predicted_class, confidence = predict(model, images[i].numpy())
        actual_class = class_names[labels[i]]

        plt.title(f"Actual: {actual_class},\n Predicted: {predicted_class}.\n Confid
```

```
        plt.axis("off")
```

```
C:\Users\harsha.teja\Anaconda3\envs\deeplearning\lib\site-packages\ipykernel_launche
r.py:4: MatplotlibDeprecationWarning: Adding an axes using the same arguments as a p
revious axes currently reuses the earlier instance.  In a future version, a new inst
ance will always be created and returned.  Meanwhile, this warning can be suppresse
d, and the future behavior ensured, by passing a unique label to each axes instance.
  after removing the cwd from sys.path.
```

In [17]:

```
plt.figure(figsize=(15, 15))
for images, labels in test_dataset.take(10):
    for i in range(9):
        ax = plt.subplot(3, 3, i + 1)
        plt.imshow(images[i].numpy().astype("uint8"))

        predicted_class, confidence = predict(model, images[i].numpy())
        actual_class = class_names[labels[i]]

        plt.title(f"Actual: {actual_class},\n Predicted: {predicted_class}.\n Confid

        plt.axis("off")
```

```
C:\Users\harsha.teja\Anaconda3\envs\deeplearning\lib\site-packages\ipykernel_launche
r.py:4: MatplotlibDeprecationWarning: Adding an axes using the same arguments as a p
revious axes currently reuses the earlier instance.  In a future version, a new inst
```

```
ance will always be created and returned.  Meanwhile, this warning can be suppresse
d, and the future behavior ensured, by passing a unique label to each axes instance.
  after removing the cwd from sys.path.
```

In [ ]:
```
#predication of test Images will be done another file
```

By Harsha Teja Bolla