```python
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```python
!unzip "/content/drive/MyDrive/AUTOMATED_WEATHER_CLASSIFICATION_USING_TRANSFER_LEARNING/archive.zip"
```

```
  inflating: Multi-class Weather Dataset/Sunrise/sunrise208.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise209.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise21.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise210.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise211.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise212.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise213.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise214.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise215.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise216.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise217.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise218.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise219.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise22.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise220.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise221.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise222.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise223.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise224.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise225.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise226.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise227.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise228.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise229.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise23.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise230.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise231.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise232.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise233.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise234.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise235.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise236.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise237.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise238.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise239.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise24.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise240.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise241.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise242.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise243.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise244.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise245.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise246.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise247.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise248.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise249.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise25.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise250.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise251.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise252.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise253.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise254.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise255.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise256.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise257.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise258.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise259.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise26.jpg
  inflating: Multi-class Weather Dataset/Sunrise/sunrise260.jpg
```

```python
from tensorflow.keras.layers import Dense,Flatten,Input
from tensorflow.keras.models import Model
from tensorflow.keras.preprocessing import image
from tensorflow.keras.preprocessing.image import ImageDataGenerator, load_img
import numpy as np
```

```python
base_dir="/content/Multi-class Weather Dataset"
```

```python
train_gen = ImageDataGenerator(rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    validation_split=0.2)

train = train_gen.flow_from_directory(
    base_dir,
    target_size=(224,224),
    batch_size= 32,
    class_mode='categorical',
    subset='training')

validation_gen = train_gen.flow_from_directory(
    base_dir,
    target_size=(224, 224),
    batch_size= 32,
    class_mode='categorical',
    subset='validation')
```

```
Found 901 images belonging to 4 classes.
Found 224 images belonging to 4 classes.
```

VGG16

```python
from tensorflow.keras.applications.vgg16 import VGG16, preprocess_input
```

```python
# Adding the preprocessing layer to the front of vgg
vgg = VGG16(include_top=False,weights='imagenet',input_shape=(224,224,3))
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/vgg16/vgg16_weights_tf_dim_ordering_tf_kernels_notop.h5
58889256/58889256 [==============================] - 0s 0us/step
```

```python
# Train model with existing weights
for layer in vgg.layers:
  print(layer)
```

```
<keras.engine.input_layer.InputLayer object at 0x7f9b29053e80>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b29081900>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b29080f70>
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x7f9b29082c50>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b29083790>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b290838b0>
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x7f9b28508b80>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b290837f0>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b2850a440>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b28509720>
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x7f9b2850ba90>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b2850ba00>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b28509930>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b28525780>
```

```
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x7f9b28526830>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b28527220>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b28527490>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b285275b0>
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x7f9b28538970>
```

```python
# Train model with existing weights
for layer in vgg.layers:
    layer.trainable=False
```

```python
x = Flatten()(vgg.output)
```

```python
# output layer
prediction = Dense(4,activation='softmax')(x)
# Create Vgg16 model
model = Model(inputs=vgg.input,outputs=prediction)
```

```python
model.summary()
```

```
Model: "model"
_____
 Layer (type)              Output Shape             Param #
===============================================================
 input_1 (InputLayer)      [(None, 224, 224, 3)]    0

 block1_conv1 (Conv2D)     (None, 224, 224, 64)     1792

 block1_conv2 (Conv2D)     (None, 224, 224, 64)     36928

 block1_pool (MaxPooling2D) (None, 112, 112, 64)    0

 block2_conv1 (Conv2D)     (None, 112, 112, 128)    73856

 block2_conv2 (Conv2D)     (None, 112, 112, 128)    147584

 block2_pool (MaxPooling2D) (None, 56, 56, 128)     0

 block3_conv1 (Conv2D)     (None, 56, 56, 256)      295168

 block3_conv2 (Conv2D)     (None, 56, 56, 256)      590080

 block3_conv3 (Conv2D)     (None, 56, 56, 256)      590080

 block3_pool (MaxPooling2D) (None, 28, 28, 256)     0

 block4_conv1 (Conv2D)     (None, 28, 28, 512)      1180160

 block4_conv2 (Conv2D)     (None, 28, 28, 512)      2359808

 block4_conv3 (Conv2D)     (None, 28, 28, 512)      2359808

 block4_pool (MaxPooling2D) (None, 14, 14, 512)     0

 block5_conv1 (Conv2D)     (None, 14, 14, 512)      2359808

 block5_conv2 (Conv2D)     (None, 14, 14, 512)      2359808

 block5_conv3 (Conv2D)     (None, 14, 14, 512)      2359808

 block5_pool (MaxPooling2D) (None, 7, 7, 512)       0

 flatten_1 (Flatten)       (None, 25088)            0

 dense (Dense)             (None, 4)                100356

===============================================================
Total params: 14,815,044
Trainable params: 100,356
Non-trainable params: 14,714,688
_____
```

```python
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

```python
model.fit_generator(train,validation_data=validation_gen,epochs=10,steps_per_epoch=len(train),
                    validation_steps=len(validation_gen))
```

```
<ipython-input-16-3b4b868b8d22>:1: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Model.fit`, which supports generators.
  model.fit_generator(train,validation_data=validation_gen,epochs=10,steps_per_epoch=len(train),
Epoch 1/10
29/29 [==============================] - 35s 764ms/step - loss: 0.9922 - accuracy: 0.6471 - val_loss: 1.1540 - val_accuracy: 0.6473
Epoch 2/10
29/29 [==============================] - 26s 904ms/step - loss: 0.3346 - accuracy: 0.8746 - val_loss: 0.6467 - val_accuracy: 0.7902
Epoch 3/10
29/29 [==============================] - 21s 710ms/step - loss: 0.1826 - accuracy: 0.9445 - val_loss: 0.4057 - val_accuracy: 0.8839
Epoch 4/10
29/29 [==============================] - 20s 690ms/step - loss: 0.1305 - accuracy: 0.9689 - val_loss: 0.4452 - val_accuracy: 0.8571
Epoch 5/10
29/29 [==============================] - 20s 688ms/step - loss: 0.0989 - accuracy: 0.9811 - val_loss: 0.3957 - val_accuracy: 0.8705
Epoch 6/10
29/29 [==============================] - 21s 710ms/step - loss: 0.1014 - accuracy: 0.9778 - val_loss: 0.3264 - val_accuracy: 0.8795
Epoch 7/10
29/29 [==============================] - 21s 708ms/step - loss: 0.0893 - accuracy: 0.9811 - val_loss: 0.4530 - val_accuracy: 0.8304
Epoch 8/10
29/29 [==============================] - 21s 746ms/step - loss: 0.0851 - accuracy: 0.9867 - val_loss: 0.2498 - val_accuracy: 0.9241
Epoch 9/10
29/29 [==============================] - 21s 725ms/step - loss: 0.0628 - accuracy: 0.9900 - val_loss: 0.4094 - val_accuracy: 0.8393
Epoch 10/10
29/29 [==============================] - 20s 687ms/step - loss: 0.0526 - accuracy: 0.9900 - val_loss: 0.3020 - val_accuracy: 0.8973
<keras.callbacks.History at 0x7f9b2857f430>
```

```python
model.save("Automated_weather_classification.h5")
```

RESNET50

```python
#RESNET 50
from tensorflow.keras.applications.resnet50 import ResNet50
resnet = ResNet50(include_top=False,input_shape=(224,224,3))
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/resnet/resnet50_weights_tf_dim_ordering_tf_kernels_notop.h5
94765736/94765736 [==============================] - 3s 0us/step
```

```python
for layer in resnet.layers:
    layer.trainable=False
```

```python
for layer in resnet.layers:
    print(layer)
```

```
<keras.layers.core.activation.Activation object at 0x7f9b182a7160>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b18174820>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b181774f0>
<keras.layers.core.activation.Activation object at 0x7f9b18177c40>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b18177f40>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b18174fd0>
<keras.layers.merging.add.Add object at 0x7f9b18175180>
<keras.layers.core.activation.Activation object at 0x7f9b1819c520>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b1819c610>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b1819fa00>
<keras.layers.core.activation.Activation object at 0x7f9b1819ff10>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b1819fb50>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b1819ec80>
<keras.layers.core.activation.Activation object at 0x7f9b1819d840>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b181b4df0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b181b7160>
<keras.layers.merging.add.Add object at 0x7f9b1819f370>
<keras.layers.core.activation.Activation object at 0x7f9b181b4b50>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b181b6410>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b181b7dc0>
<keras.layers.core.activation.Activation object at 0x7f9b181b7070>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b181d0c40>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b181d28f0>
<keras.layers.core.activation.Activation object at 0x7f9b181d3100>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b181d1f30>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b181d3310>
<keras.layers.merging.add.Add object at 0x7f9b181d1e10>
<keras.layers.core.activation.Activation object at 0x7f9b181d3b80>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b181d3cd0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b181b4cd0>
<keras.layers.core.activation.Activation object at 0x7f9b181b6b90>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b181d2bf0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b181b6530>
<keras.layers.core.activation.Activation object at 0x7f9b181b7760>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b182a5630>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b18135840>
<keras.layers.merging.add.Add object at 0x7f9b18136230>
<keras.layers.core.activation.Activation object at 0x7f9b181fd1e0>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b181ff5e0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b181fe650>
<keras.layers.core.activation.Activation object at 0x7f9b181ffa90>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b181ffc40>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b181ff5b0>
<keras.layers.core.activation.Activation object at 0x7f9b181ff700>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b181fdae0>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b18210d60>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b181ff220>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9b18213670>
<keras.layers.merging.add.Add object at 0x7f9b181fcfd0>
<keras.layers.core.activation.Activation object at 0x7f9b18213d60>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9b180242b0>
```

```python
x = Flatten()(resnet.output)
```

```python
out = Dense(4, activation='softmax')(x)
```

```python
res_model = Model(inputs=resnet.input,outputs=out)
```

```python
res_model.summary()
```

conv4_block5_3_bn (BatchNormal ization)	(None, 14, 14, 1024)	4096	['conv4_block5_3_conv[0][0]']
conv4_block5_add (Add)	(None, 14, 14, 1024)	0	['conv4_block4_out[0][0]', 'conv4_block5_3_bn[0][0]']
conv4_block5_out (Activation)	(None, 14, 14, 1024)	0	['conv4_block5_add[0][0]']
conv4_block6_1_conv (Conv2D)	(None, 14, 14, 256)	262400	['conv4_block5_out[0][0]']
conv4_block6_1_bn (BatchNormal ization)	(None, 14, 14, 256)	1024	['conv4_block6_1_conv[0][0]']
conv4_block6_1_relu (Activatio n)	(None, 14, 14, 256)	0	['conv4_block6_1_bn[0][0]']
conv4_block6_2_conv (Conv2D)	(None, 14, 14, 256)	590080	['conv4_block6_1_relu[0][0]']
conv4_block6_2_bn (BatchNormal ization)	(None, 14, 14, 256)	1024	['conv4_block6_2_conv[0][0]']
conv4_block6_2_relu (Activatio n)	(None, 14, 14, 256)	0	['conv4_block6_2_bn[0][0]']
conv4_block6_3_conv (Conv2D)	(None, 14, 14, 1024)	263168	['conv4_block6_2_relu[0][0]']
conv4_block6_3_bn (BatchNormal ization)	(None, 14, 14, 1024)	4096	['conv4_block6_3_conv[0][0]']
conv4_block6_add (Add)	(None, 14, 14, 1024)	0	['conv4_block5_out[0][0]', 'conv4_block6_3_bn[0][0]']
conv4_block6_out (Activation)	(None, 14, 14, 1024)	0	['conv4_block6_add[0][0]']
conv5_block1_1_conv (Conv2D)	(None, 7, 7, 512)	524800	['conv4_block6_out[0][0]']
conv5_block1_1_bn (BatchNormal ization)	(None, 7, 7, 512)	2048	['conv5_block1_1_conv[0][0]']
conv5_block1_1_relu (Activatio n)	(None, 7, 7, 512)	0	['conv5_block1_1_bn[0][0]']
conv5_block1_2_conv (Conv2D)	(None, 7, 7, 512)	2359808	['conv5_block1_1_relu[0][0]']
conv5_block1_2_bn (BatchNormal ization)	(None, 7, 7, 512)	2048	['conv5_block1_2_conv[0][0]']
conv5_block1_2_relu (Activatio n)	(None, 7, 7, 512)	0	['conv5_block1_2_bn[0][0]']
conv5_block1_0_conv (Conv2D)	(None, 7, 7, 2048)	2099200	['conv4_block6_out[0][0]']
conv5_block1_3_conv (Conv2D)	(None, 7, 7, 2048)	1050624	['conv5_block1_2_relu[0][0]']
conv5_block1_0_bn (BatchNormal ization)	(None, 7, 7, 2048)	8192	['conv5_block1_0_conv[0][0]']

```python
res_model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

```python
res_model.fit(train,epochs=5,validation_data=validation_gen,steps_per_epoch=len(train),
              validation_steps=len(validation_gen))
```

```
Epoch 1/5
29/29 [==============================] - 27s 772ms/step - loss: 4.5890 - accuracy: 0.2963 - val_loss: 1.4691 - val_accuracy: 0.3616
Epoch 2/5
29/29 [==============================] - 20s 697ms/step - loss: 1.4455 - accuracy: 0.3685 - val_loss: 1.3120 - val_accuracy: 0.3304
Epoch 3/5
29/29 [==============================] - 20s 697ms/step - loss: 1.2164 - accuracy: 0.5017 - val_loss: 1.0394 - val_accuracy: 0.5536
Epoch 4/5
29/29 [==============================] - 21s 743ms/step - loss: 1.1845 - accuracy: 0.5305 - val_loss: 1.3555 - val_accuracy: 0.4062
```

```
Epoch 5/5
29/29 [==============================] - 21s 726ms/step - loss: 1.1225 - accuracy: 0.5361 - val_loss: 1.1372 - val_accuracy: 0.4866
<keras.callbacks.History at 0x7f9b181d3dc0>
```

INCEPTIONV3

```
#inseption-v3
from tensorflow.keras.applications.inception_v3 import InceptionV3
inc = InceptionV3(include_top=False,weights="imagenet",input_shape=(224, 224, 3))
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/inception_v3/inception_v3_weights_tf_dim_ordering_tf_kernels_notop.h5
87910968/87910968 [==============================] - 0s 0us/step
```

```
for layer in inc.layers:
  layer.trainable=False
```

```
for layer in inc.layers:
  print(layer)
```

```
<keras.layers.pooling.average_pooling2d.AveragePooling2D object at 0x7f9a960f4f40>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a962b7610>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a960cecb0>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a960f4580>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a960f7460>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a962b79d0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a960cf370>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a960e3a60>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a960f7a60>
<keras.layers.core.activation.Activation object at 0x7f9a962b4b20>
<keras.layers.core.activation.Activation object at 0x7f9a960cd210>
<keras.layers.core.activation.Activation object at 0x7f9a960f6740>
<keras.layers.core.activation.Activation object at 0x7f9a960f77c0>
<keras.layers.merging.concatenate.Concatenate object at 0x7f9a9610c400>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a9610da50>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a962a7d00>
<keras.layers.core.activation.Activation object at 0x7f9b0057fd00>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a9610ef50>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a96129630>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a9610f670>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a9612a2f0>
<keras.layers.core.activation.Activation object at 0x7f9a9610c8e0>
<keras.layers.core.activation.Activation object at 0x7f9a9612a4a0>
<keras.layers.pooling.average_pooling2d.AveragePooling2D object at 0x7f9a96131480>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a9610d810>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a9610c7f0>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a96129e70>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a96131ff0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a960f75b0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a960cea10>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a96129c30>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a96132c50>
<keras.layers.core.activation.Activation object at 0x7f9a9610d9f0>
<keras.layers.core.activation.Activation object at 0x7f9a960cc460>
<keras.layers.core.activation.Activation object at 0x7f9a9612b4f0>
<keras.layers.core.activation.Activation object at 0x7f9a96132530>
<keras.layers.merging.concatenate.Concatenate object at 0x7f9a96131570>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a96141ae0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a96143010>
<keras.layers.core.activation.Activation object at 0x7f9a961402e0>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a961437f0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a961426b0>
<keras.layers.core.activation.Activation object at 0x7f9a96142680>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a961403a0>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a96158850>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a96130370>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a9615ad70>
<keras.layers.core.activation.Activation object at 0x7f9a96131300>
<keras.layers.core.activation.Activation object at 0x7f9a9615b850>
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x7f9a9615a320>
<keras.layers.merging.concatenate.Concatenate object at 0x7f9a961598d0>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a9616dc60>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a96133160>
<keras.layers.core.activation.Activation object at 0x7f9a96159d50>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a9616c340>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a9619ad10>
<keras.layers.core.activation.Activation object at 0x7f9a96199de0>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a9616e560>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a9619bf70>
```

```
x1= Flatten()(inc.output)
```

```
pred = Dense(4, activation='softmax')(x1)
```

```
inception_model = Model(inputs=inc.input,outputs=pred)
```

```
inception_model.summary()
```

```
                                                    'activation_68[0][0]',
                                                    'activation_69[0][0]']

 conv2d_72 (Conv2D)              (None, 12, 12, 192)  147456      ['mixed7[0][0]']

 batch_normalization_72 (BatchN  (None, 12, 12, 192)  576         ['conv2d_72[0][0]']
 ormalization)

 activation_72 (Activation)      (None, 12, 12, 192)  0           ['batch_normalization_72[0][0]']

 conv2d_73 (Conv2D)              (None, 12, 12, 192)  258048      ['activation_72[0][0]']

 batch_normalization_73 (BatchN  (None, 12, 12, 192)  576         ['conv2d_73[0][0]']
 ormalization)
```

```python
inception_model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

```python
inception_model.fit(train,epochs=5,validation_data=validation_gen,steps_per_epoch=len(train),
            validation_steps=len(validation_gen))
```

```
Epoch 1/5
29/29 [==============================] - 28s 763ms/step - loss: 3.1814 - accuracy: 0.6948 - val_loss: 1.0506 - val_accuracy: 0.8616
Epoch 2/5
29/29 [==============================] - 20s 679ms/step - loss: 0.5123 - accuracy: 0.9201 - val_loss: 0.7987 - val_accuracy: 0.8616
Epoch 3/5
29/29 [==============================] - 20s 669ms/step - loss: 0.1471 - accuracy: 0.9656 - val_loss: 1.1385 - val_accuracy: 0.8125
Epoch 4/5
29/29 [==============================] - 20s 689ms/step - loss: 0.2070 - accuracy: 0.9534 - val_loss: 0.9986 - val_accuracy: 0.8348
Epoch 5/5
29/29 [==============================] - 20s 690ms/step - loss: 0.1463 - accuracy: 0.9634 - val_loss: 0.5372 - val_accuracy: 0.8884
<keras.callbacks.History at 0x7f9a89558190>
```

XCEPTION

```python
from keras.applications.xception import Xception
```

```python
xception = Xception(include_top=False,input_shape=(224,224,3))
```

```python
for layer in xception.layers:
  layer.trainable=False
```

```python
for layer in xception.layers:
  print(layer)
```

```
<keras.engine.input_layer.InputLayer object at 0x7f99e879c580>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a0479d210>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a142919c0>
<keras.layers.core.activation.Activation object at 0x7f9a044f6890>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a0479df30>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a04745690>
<keras.layers.core.activation.Activation object at 0x7f9a0479f820>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a0479eef0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a04747580>
<keras.layers.core.activation.Activation object at 0x7f9a0477bf70>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a0477a770>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a724af040>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a0479f5e0>
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x7f9a0477a950>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a0479e410>
<keras.layers.merging.add.Add object at 0x7f9a0477b1c0>
<keras.layers.core.activation.Activation object at 0x7f9a047875b0>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a04786cb0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a0477be50>
<keras.layers.core.activation.Activation object at 0x7f9a04785f60>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a04786800>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a0477af80>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f9a0477a920>
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x7f9a04778df0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a04778ac0>
<keras.layers.merging.add.Add object at 0x7f9a0477a980>
<keras.layers.core.activation.Activation object at 0x7f9a142128c0>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f99f41dee60>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a14281f60>
<keras.layers.core.activation.Activation object at 0x7f9a04795900>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a0479df90>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a047973d0>
<keras.layers.convolutional.conv2d.Conv2D object at 0x7f99f41e6200>
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x7f9a04797370>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a04787eb0>
<keras.layers.merging.add.Add object at 0x7f9a04797eb0>
<keras.layers.core.activation.Activation object at 0x7f9a0456c5e0>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a047976a0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a04797dc0>
<keras.layers.core.activation.Activation object at 0x7f9a0456efb0>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a047969b0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a0456fdc0>
<keras.layers.core.activation.Activation object at 0x7f9a0456efe0>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a0456fdf0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a0456e710>
<keras.layers.merging.add.Add object at 0x7f9a04575a20>
<keras.layers.core.activation.Activation object at 0x7f9a04574640>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a04576980>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a04577d60>
<keras.layers.core.activation.Activation object at 0x7f9a04577370>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a045761d0>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a04577580>
<keras.layers.core.activation.Activation object at 0x7f9a0450a590>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a04574790>
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x7f9a04575150>
<keras.layers.merging.add.Add object at 0x7f9a0450b460>
<keras.layers.core.activation.Activation object at 0x7f9a0450aaa0>
<keras.layers.convolutional.separable_conv2d.SeparableConv2D object at 0x7f9a04508f40>,
```

```python
x = Flatten()(xception.output)
```

```python
predic = Dense(4, activation='softmax')(x)
```

```python
xce_model = Model(inputs=xception.input,outputs=predic)
```

```python
xce_model.summary()
```

```
block8_sepconv3_bn (BatchNorma   (None, 14, 14, 728)   2912      ['block8_sepconv3[0][0]']
lization)

add_174 (Add)                    (None, 14, 14, 728)   0         ['block8_sepconv3_bn[0][0]',
                                                                   'add_173[0][0]']

block9_sepconv1_act (Activatio   (None, 14, 14, 728)   0         ['add_174[0][0]']
n)

block9_sepconv1 (SeparableConv   (None, 14, 14, 728)   536536    ['block9_sepconv1_act[0][0]']
2D)

block9_sepconv1_bn (BatchNorma   (None, 14, 14, 728)   2912      ['block9_sepconv1[0][0]']
lization)

block9_sepconv2_act (Activatio   (None, 14, 14, 728)   0         ['block9_sepconv1_bn[0][0]']
n)

block9_sepconv2 (SeparableConv   (None, 14, 14, 728)   536536    ['block9_sepconv2_act[0][0]']
2D)

block9_sepconv2_bn (BatchNorma   (None, 14, 14, 728)   2912      ['block9_sepconv2[0][0]']
lization)

block9_sepconv3_act (Activatio   (None, 14, 14, 728)   0         ['block9_sepconv2_bn[0][0]']
n)

block9_sepconv3 (SeparableConv   (None, 14, 14, 728)   536536    ['block9_sepconv3_act[0][0]']
2D)

block9_sepconv3_bn (BatchNorma   (None, 14, 14, 728)   2912      ['block9_sepconv3[0][0]']
lization)

add_175 (Add)                    (None, 14, 14, 728)   0         ['block9_sepconv3_bn[0][0]',
                                                                   'add_174[0][0]']

block10_sepconv1_act (Activati   (None, 14, 14, 728)   0         ['add_175[0][0]']
on)

block10_sepconv1 (SeparableCon   (None, 14, 14, 728)   536536    ['block10_sepconv1_act[0][0]']
v2D)

block10_sepconv1_bn (BatchNorm   (None, 14, 14, 728)   2912      ['block10_sepconv1[0][0]']
alization)
```

```python
xce_model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

```python
xce_model.fit(train,epochs=5,validation_data=validation_gen,steps_per_epoch=len(train),
              validation_steps=len(validation_gen))
```

```
Epoch 1/5
29/29 [==============================] - 25s 783ms/step - loss: 0.9935 - accuracy: 0.7858 - val_loss: 0.9955 - val_accuracy: 0.7902
Epoch 2/5
29/29 [==============================] - 21s 730ms/step - loss: 0.3842 - accuracy: 0.9134 - val_loss: 0.8885 - val_accuracy: 0.8304
Epoch 3/5
29/29 [==============================] - 21s 725ms/step - loss: 0.4068 - accuracy: 0.9267 - val_loss: 0.7422 - val_accuracy: 0.8750
Epoch 4/5
29/29 [==============================] - 21s 701ms/step - loss: 0.2901 - accuracy: 0.9412 - val_loss: 0.9252 - val_accuracy: 0.8661
Epoch 5/5
29/29 [==============================] - 20s 678ms/step - loss: 0.1218 - accuracy: 0.9656 - val_loss: 0.6201 - val_accuracy: 0.8795
<keras.callbacks.History at 0x7f9a76cf2650>
```

✓ 3m 6s   completed at 11:18 AM