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

```
Mounted at /content/gdrive
```

```
import os
import cv2
import numpy as np
from sklearn.utils import shuffle
from random import randint

import tensorflow as tf
import matplotlib.pyplot as plt
from keras.preprocessing.image import ImageDataGenerator,img_to_array,load_img
from PIL import Image
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from keras.callbacks import ReduceLROnPlateau
from keras.applications.inception_v3 import InceptionV3
from keras.applications.mobilenet_v2 import MobileNetV2
```

```
train_DIR = "/content/gdrive/MyDrive/Colab-Notebooks/Semester 3/CV/Asgt 3/Dataset/Manua

train_datagen = ImageDataGenerator(shear_range=0.2,
                                   zoom_range=0.2,
                                   fill_mode="nearest",
                                   horizontal_flip=True,
                                   vertical_flip=True,rotation_range=90)


train_generator = train_datagen.flow_from_directory(train_DIR,
                                                    batch_size=32,
                                                    class_mode='categorical',
                                                    target_size=(150, 150))

test_DIR = "/content/gdrive/MyDrive/Colab-Notebooks/Semester 3/CV/Asgt 3/Dataset/Manual

validation_datagen = ImageDataGenerator(width_shift_range=0.25,
                                        height_shift_range=0.25,rotation_range=90)


validation_generator = validation_datagen.flow_from_directory(test_DIR,
                                                    batch_size=128,
                                                    class_mode='categorical',
                                                    target_size=(150, 150))

```

```
Found 14034 images belonging to 6 classes.
Found 3000 images belonging to 6 classes.
```

```
inceptionV3 = InceptionV3(include_top= False, input_shape=(150,150,3))
for layer in inceptionV3.layers:
  layer.trainable = False
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/i
87916544/87910968 [==============================] - 1s 0us/step
```

```
last_layer = inceptionV3.get_layer('mixed9')
print('last layer output shape: ', last_layer.output_shape)
last_output = last_layer.output
```

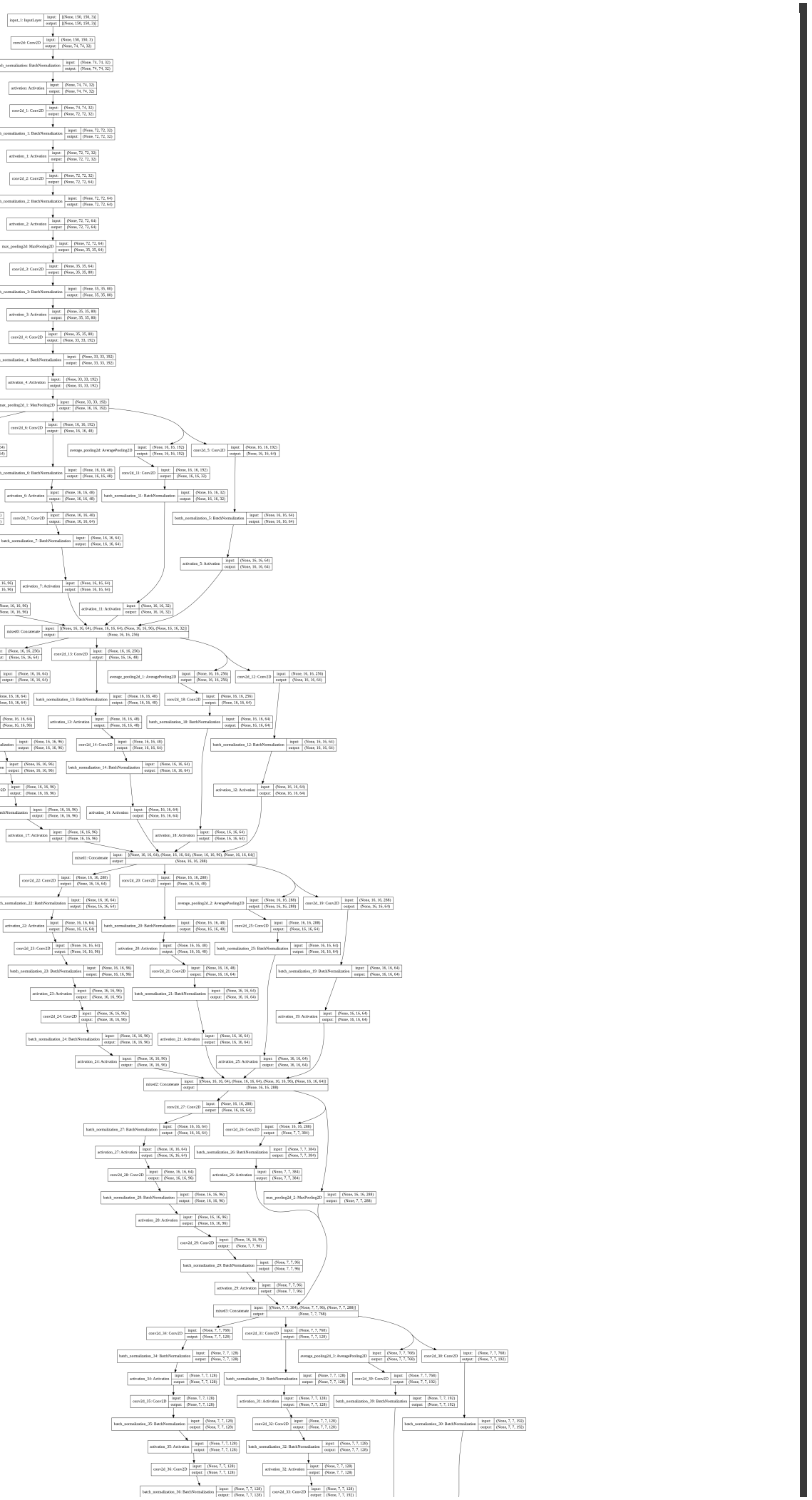
```
last layer output shape:  (None, 3, 3, 2048)
```

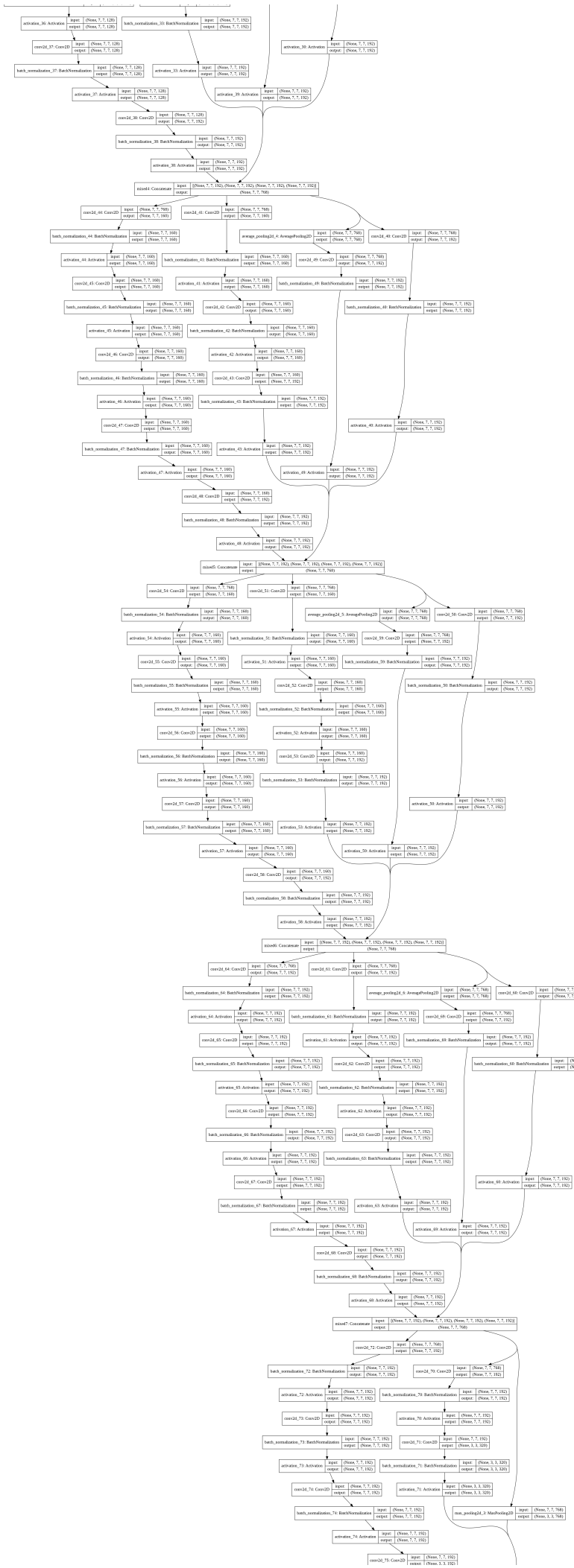
```
x = tf.keras.layers.Flatten()(last_output)
x = tf.keras.layers.Dense(units = 1024, activation = tf.nn.relu)(x)
x = tf.keras.layers.Dropout(0.3)(x)
x = tf.keras.layers.Dense  (6, activation = tf.nn.softmax)(x)

model = tf.keras.Model( inceptionV3.input, x)

learning_rate_reduction = ReduceLROnPlateau(monitor='val_acc',
                                            patience=1,
                                            verbose=1,
                                            factor=0.30,
                                            min_lr=0.0003)

model.compile(loss = 'categorical_crossentropy', optimizer= tf.keras.optimizers.Adam(),

model.summary()
```

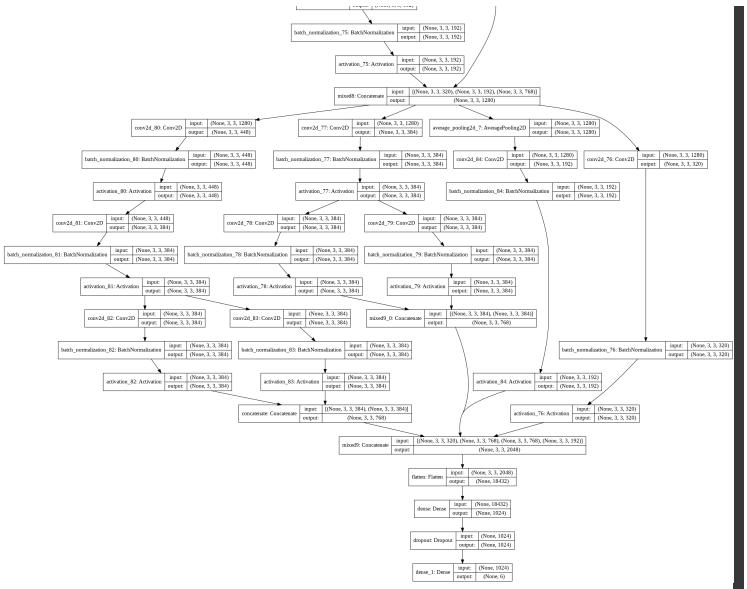
```
activation_81 (Activation)      (None, 3, 3, 384)    0           batch_normalizati
conv2d_78 (Conv2D)              (None, 3, 3, 384)    442368      activation_77[0][
conv2d_79 (Conv2D)              (None, 3, 3, 384)    442368      activation_77[0][
conv2d_82 (Conv2D)              (None, 3, 3, 384)    442368      activation_81[0][
conv2d_83 (Conv2D)              (None, 3, 3, 384)    442368      activation_81[0][
average_pooling2d_7 (AveragePoo (None, 3, 3, 1280)   0           mixed8[0][0]
conv2d_76 (Conv2D)              (None, 3, 3, 320)    409600      mixed8[0][0]
batch_normalization_78 (BatchNo (None, 3, 3, 384)    1152        conv2d_78[0][0]
batch_normalization_79 (BatchNo (None, 3, 3, 384)    1152        conv2d_79[0][0]
batch_normalization_82 (BatchNo (None, 3, 3, 384)    1152        conv2d_82[0][0]
batch_normalization_83 (BatchNo (None, 3, 3, 384)    1152        conv2d_83[0][0]
conv2d_84 (Conv2D)              (None, 3, 3, 192)    245760      average_pooling2d
batch_normalization_76 (BatchNo (None, 3, 3, 320)    960         conv2d_76[0][0]
activation_78 (Activation)      (None, 3, 3, 384)    0           batch_normalizati
activation_79 (Activation)      (None, 3, 3, 384)    0           batch_normalizati
```

```
activation_82 (Activation)      (None, 3, 3, 384)    0           batch_normalizati
activation_83 (Activation)      (None, 3, 3, 384)    0           batch_normalizati
batch_normalization_84 (BatchNo (None, 3, 3, 192)    576         conv2d_84[0][0]
activation_76 (Activation)      (None, 3, 3, 320)    0           batch_normalizati
mixed9_0 (Concatenate)          (None, 3, 3, 768)    0           activation_78[0][
                                                                 activation_79[0][
concatenate (Concatenate)       (None, 3, 3, 768)    0           activation_82[0][
                                                                 activation_83[0][
activation_84 (Activation)      (None, 3, 3, 192)    0           batch_normalizati
mixed9 (Concatenate)            (None, 3, 3, 2048)   0           activation_76[0][
                                                                 mixed9_0[0][0]
                                                                 concatenate[0][0]
                                                                 activation_84[0][
flatten (Flatten)               (None, 18432)        0           mixed9[0][0]
dense (Dense)                   (None, 1024)         18875392    flatten[0][0]
dropout (Dropout)               (None, 1024)         0           dense[0][0]
dense_1 (Dense)                 (None, 6)            6150        dropout[0][0]
```

```
1 tf.keras.utils.plot_model(model, to_file='RMSprop.png', show_shapes=True, show_layer_na
```


input_1: InputLayer
conv2d: Conv2D
batch_normalization: BatchNormalization
activation: Activation
conv2d_1: Conv2D
batch_normalization_1: BatchNormalization
activation_1: Activation
conv2d_2: Conv2D
batch_normalization_2: BatchNormalization
activation_2: Activation
max_pooling2d: MaxPooling2D
conv2d_3: Conv2D
batch_normalization_3: BatchNormalization
activation_3: Activation
conv2d_4: Conv2D
batch_normalization_4: BatchNormalization
activation_4: Activation
max_pooling2d_1: MaxPooling2D
conv2d_8: Conv2D
conv2d_6: Conv2D
average_pooling2d: AveragePooling2D
conv2d_5: Conv2D
batch_normalization_8: BatchNormalization
batch_normalization_6: BatchNormalization
conv2d_11: Conv2D
activation_8: Activation
activation_6: Activation
batch_normalization_11: BatchNormalization
conv2d_9: Conv2D
batch_normalization_5: BatchNormalization
batch_normalization_9: BatchNormalization
conv2d_7: Conv2D
activation_5: Activation
activation_9: Activation
batch_normalization_7: BatchNormalization
conv2d_10: Conv2D
batch_normalization_10: BatchNormalization
activation_7: Activation
activation_10: Activation
activation_11: Activation
mixed0: Concatenate
conv2d_15: Conv2D
conv2d_13: Conv2D
batch_normalization_15: BatchNormalization
average_pooling2d_1: AveragePooling2D
conv2d_12: Conv2D
activation_15: Activation
batch_normalization_13: BatchNormalization
conv2d_18: Conv2D
conv2d_16: Conv2D
activation_13: Activation
batch_normalization_18: BatchNormalization
batch_normalization_16: BatchNormalization
conv2d_14: Conv2D
batch_normalization_12: BatchNormalization
activation_16: Activation
batch_normalization_14: BatchNormalization
conv2d_17: Conv2D
activation_12: Activation
batch_normalization_17: BatchNormalization
activation_14: Activation
activation_17: Activation
activation_18: Activation
mixed1: Concatenate
conv2d_22: Conv2D
conv2d_20: Conv2D
batch_normalization_22: BatchNormalization
average_pooling2d_2: AveragePooling2D
conv2d_19: Conv2D
activation_22: Activation
batch_normalization_20: BatchNormalization
conv2d_25: Conv2D
conv2d_23: Conv2D
activation_20: Activation
batch_normalization_25: BatchNormalization
batch_normalization_23: BatchNormalization
conv2d_21: Conv2D
batch_normalization_19: BatchNormalization
activation_23: Activation
batch_normalization_21: BatchNormalization
conv2d_24: Conv2D
activation_19: Activation
batch_normalization_24: BatchNormalization
activation_21: Activation
activation_24: Activation
activation_25: Activation
mixed2: Concatenate
conv2d_27: Conv2D
batch_normalization_27: BatchNormalization
conv2d_26: Conv2D
activation_27: Activation
batch_normalization_26: BatchNormalization
conv2d_28: Conv2D
activation_26: Activation
batch_normalization_28: BatchNormalization
max_pooling2d_2: MaxPooling2D
activation_28: Activation
conv2d_29: Conv2D
batch_normalization_29: BatchNormalization
activation_29: Activation
mixed3: Concatenate
conv2d_34: Conv2D
conv2d_31: Conv2D
batch_normalization_34: BatchNormalization
average_pooling2d_3: AveragePooling2D
conv2d_30: Conv2D
activation_34: Activation
batch_normalization_31: BatchNormalization
conv2d_39: Conv2D
conv2d_35: Conv2D
activation_31: Activation
batch_normalization_39: BatchNormalization
batch_normalization_35: BatchNormalization
conv2d_32: Conv2D
batch_normalization_30: BatchNormalization
activation_35: Activation
batch_normalization_32: BatchNormalization
conv2d_36: Conv2D
activation_32: Activation
batch_normalization_36: BatchNormalization
conv2d_33: Conv2D


activation_36: Activation
batch_normalization_33: BatchNormalization
conv2d_37: Conv2D
activation_30: Activation
batch_normalization_37: BatchNormalization
activation_33: Activation
activation_37: Activation
activation_39: Activation
conv2d_38: Conv2D
batch_normalization_38: BatchNormalization
activation_38: Activation
mixed4: Concatenate
conv2d_44: Conv2D
conv2d_41: Conv2D
average_pooling2d_4: AveragePooling2D
conv2d_40: Conv2D
batch_normalization_44: BatchNormalization
batch_normalization_41: BatchNormalization
conv2d_49: Conv2D
activation_44: Activation
activation_41: Activation
batch_normalization_49: BatchNormalization
conv2d_45: Conv2D
batch_normalization_40: BatchNormalization
batch_normalization_45: BatchNormalization
conv2d_42: Conv2D
activation_45: Activation
batch_normalization_42: BatchNormalization
conv2d_46: Conv2D
activation_42: Activation
batch_normalization_46: BatchNormalization
conv2d_43: Conv2D
activation_46: Activation
batch_normalization_43: BatchNormalization
activation_40: Activation
conv2d_47: Conv2D
batch_normalization_47: BatchNormalization
activation_43: Activation
activation_47: Activation
activation_49: Activation
conv2d_48: Conv2D
batch_normalization_48: BatchNormalization
activation_48: Activation
mixed5: Concatenate
conv2d_54: Conv2D
conv2d_51: Conv2D
batch_normalization_54: BatchNormalization
average_pooling2d_5: AveragePooling2D
conv2d_50: Conv2D
activation_54: Activation
batch_normalization_51: BatchNormalization
conv2d_59: Conv2D
conv2d_55: Conv2D
activation_51: Activation
batch_normalization_59: BatchNormalization
batch_normalization_55: BatchNormalization
conv2d_52: Conv2D
batch_normalization_50: BatchNormalization
activation_55: Activation
batch_normalization_52: BatchNormalization
conv2d_56: Conv2D
activation_52: Activation
batch_normalization_56: BatchNormalization
conv2d_53: Conv2D
activation_56: Activation
batch_normalization_53: BatchNormalization
conv2d_57: Conv2D
activation_50: Activation
batch_normalization_57: BatchNormalization
activation_53: Activation
activation_57: Activation
activation_59: Activation
conv2d_58: Conv2D
batch_normalization_58: BatchNormalization
activation_58: Activation
mixed6: Concatenate
conv2d_64: Conv2D
conv2d_61: Conv2D
batch_normalization_64: BatchNormalization
average_pooling2d_6: AveragePooling2D
conv2d_60: Conv2D
activation_64: Activation
batch_normalization_61: BatchNormalization
conv2d_69: Conv2D
conv2d_65: Conv2D
activation_61: Activation
batch_normalization_69: BatchNormalization
batch_normalization_65: BatchNormalization
conv2d_62: Conv2D
batch_normalization_60: BatchNormalization
activation_65: Activation
batch_normalization_62: BatchNormalization
conv2d_66: Conv2D
activation_62: Activation
batch_normalization_66: BatchNormalization
conv2d_63: Conv2D
activation_66: Activation
batch_normalization_63: BatchNormalization
activation_60: Activation
conv2d_67: Conv2D
batch_normalization_67: BatchNormalization
activation_63: Activation
activation_67: Activation
activation_69: Activation
conv2d_68: Conv2D
batch_normalization_68: BatchNormalization
activation_68: Activation
mixed7: Concatenate
conv2d_72: Conv2D
batch_normalization_72: BatchNormalization
conv2d_70: Conv2D
activation_72: Activation
batch_normalization_70: BatchNormalization
conv2d_73: Conv2D
activation_70: Activation
batch_normalization_73: BatchNormalization
conv2d_71: Conv2D
activation_73: Activation
batch_normalization_71: BatchNormalization
conv2d_74: Conv2D
activation_71: Activation
batch_normalization_74: BatchNormalization
max_pooling2d_3: MaxPooling2D
activation_74: Activation
conv2d_75: Conv2D


batch_normalization_75: BatchNormalization
input: (None, 3, 3, 192)
output: (None, 3, 3, 192)
activation_75: Activation
input: (None, 3, 3, 192)
output: (None, 3, 3, 192)
mixed8: Concatenate
input: [(None, 3, 3, 320), (None, 3, 3, 192), (None, 3, 3, 768)]
output: (None, 3, 3, 1280)
conv2d_80: Conv2D
input: (None, 3, 3, 1280)
output: (None, 3, 3, 448)
conv2d_77: Conv2D
input: (None, 3, 3, 1280)
output: (None, 3, 3, 384)
average_pooling2d_7: AveragePooling2D
input: (None, 3, 3, 1280)
output: (None, 3, 3, 1280)
batch_normalization_80: BatchNormalization
input: (None, 3, 3, 448)
output: (None, 3, 3, 448)
batch_normalization_77: BatchNormalization
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
conv2d_84: Conv2D
input: (None, 3, 3, 1280)
output: (None, 3, 3, 192)
conv2d_76: Conv2D
input: (None, 3, 3, 1280)
output: (None, 3, 3, 320)
activation_80: Activation
input: (None, 3, 3, 448)
output: (None, 3, 3, 448)
activation_77: Activation
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
batch_normalization_84: BatchNormalization
input: (None, 3, 3, 192)
output: (None, 3, 3, 192)
conv2d_81: Conv2D
input: (None, 3, 3, 448)
output: (None, 3, 3, 384)
conv2d_78: Conv2D
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
conv2d_79: Conv2D
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
batch_normalization_81: BatchNormalization
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
batch_normalization_78: BatchNormalization
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
batch_normalization_79: BatchNormalization
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
activation_81: Activation
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
activation_78: Activation
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
activation_79: Activation
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
conv2d_82: Conv2D
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
conv2d_83: Conv2D
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
mixed9_0: Concatenate
input: [(None, 3, 3, 384), (None, 3, 3, 384)]
output: (None, 3, 3, 768)
batch_normalization_82: BatchNormalization
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
batch_normalization_83: BatchNormalization
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
batch_normalization_76: BatchNormalization
input: (None, 3, 3, 320)
output: (None, 3, 3, 320)
activation_82: Activation
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
activation_83: Activation
input: (None, 3, 3, 384)
output: (None, 3, 3, 384)
activation_84: Activation
input: (None, 3, 3, 192)
output: (None, 3, 3, 192)
concatenate: Concatenate
input: [(None, 3, 3, 384), (None, 3, 3, 384)]
output: (None, 3, 3, 768)
activation_76: Activation
input: (None, 3, 3, 320)
output: (None, 3, 3, 320)
mixed9: Concatenate
input: [(None, 3, 3, 320), (None, 3, 3, 768), (None, 3, 3, 768), (None, 3, 3, 192)]
output: (None, 3, 3, 2048)
flatten: Flatten
input: (None, 3, 3, 2048)
output: (None, 18432)
dense: Dense
input: (None, 18432)
output: (None, 1024)
dropout: Dropout
input: (None, 1024)
output: (None, 1024)
dense_1: Dense
input: (None, 1024)
output: (None, 6)

```
history = model.fit(train_generator,
                    epochs = 10,
                    verbose = 1,
                   validation_data = validation_generator,
                   callbacks=[learning_rate_reduction])
```

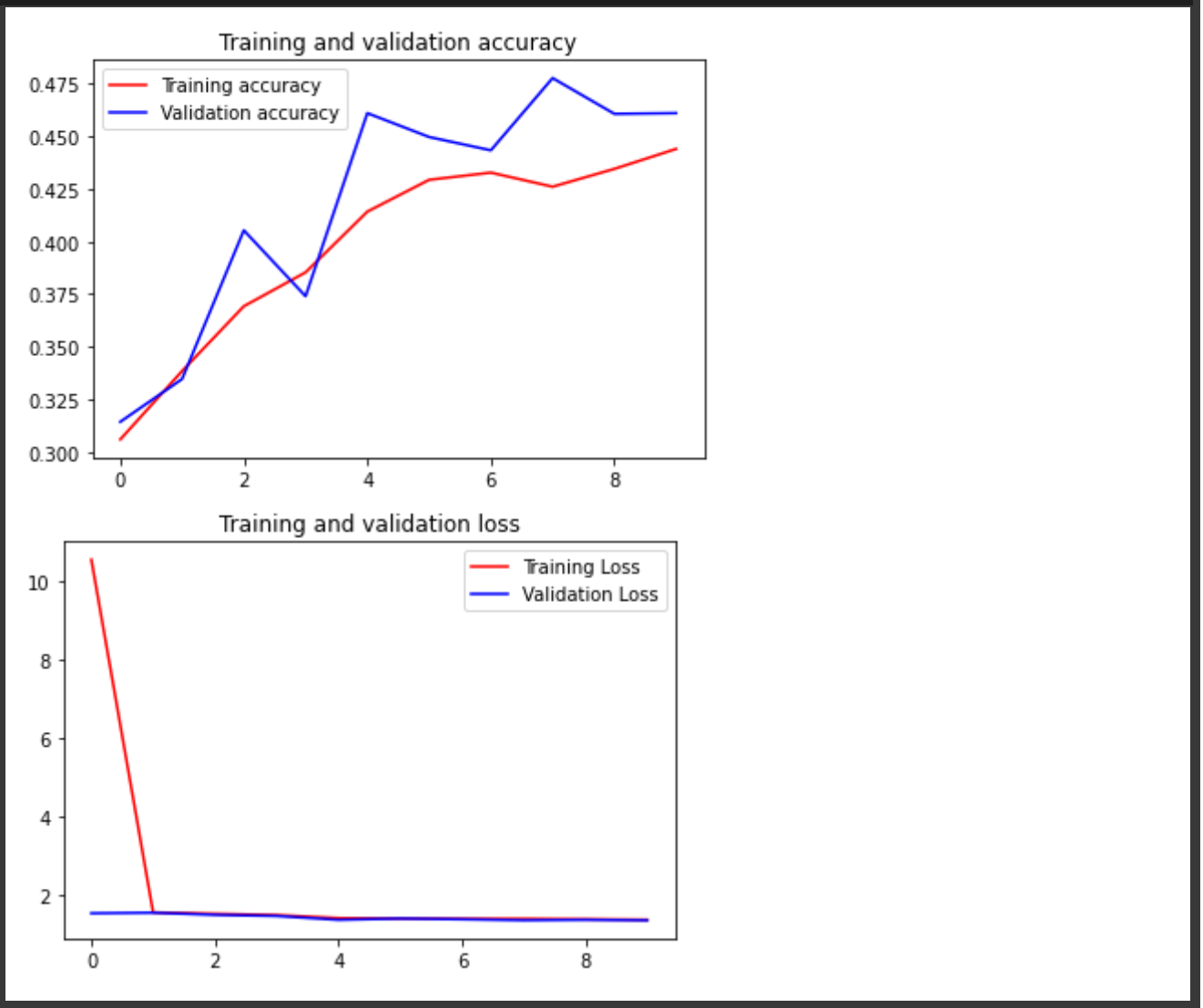
```
Epoch 1/10
439/439 [==============================] - 3879s 9s/step - loss: 37.7457 - acc: 0.29
Epoch 2/10
439/439 [==============================] - 123s 280ms/step - loss: 1.5583 - acc: 0.3
Epoch 3/10
439/439 [==============================] - 121s 277ms/step - loss: 1.5203 - acc: 0.3
Epoch 4/10
439/439 [==============================] - 122s 278ms/step - loss: 1.4866 - acc: 0.3

Epoch 00004: ReduceLROnPlateau reducing learning rate to 0.0003000000142492354.
Epoch 5/10
439/439 [==============================] - 119s 272ms/step - loss: 1.4154 - acc: 0.4
Epoch 6/10
439/439 [==============================] - 118s 269ms/step - loss: 1.3912 - acc: 0.4

Epoch 00006: ReduceLROnPlateau reducing learning rate to 0.0003.
Epoch 7/10
439/439 [==============================] - 118s 268ms/step - loss: 1.3881 - acc: 0.4

Epoch 00007: ReduceLROnPlateau reducing learning rate to 0.0003.
Epoch 8/10
439/439 [==============================] - 116s 265ms/step - loss: 1.3839 - acc: 0.4
Epoch 9/10
439/439 [==============================] - 117s 266ms/step - loss: 1.3656 - acc: 0.4

Epoch 00009: ReduceLROnPlateau reducing learning rate to 0.0003.
Epoch 10/10
439/439 [==============================] - 116s 264ms/step - loss: 1.3574 - acc: 0.4

Epoch 00010: ReduceLROnPlateau reducing learning rate to 0.0003.
```

```
model.save('/content/gdrive/MyDrive/Colab-Notebooks/Semester 3/CV/Asgt 3/Weights/Incept
```

```
from tensorflow import keras
model = keras.models.load_model('/content/gdrive/MyDrive/Colab-Notebooks/Semester 3/CV/
```

```
%matplotlib inline
acc = history.history['acc']
val_acc = history.history['val_acc']
loss = history.history['loss']
val_loss = history.history['val_loss']

epochs = range(len(acc))

```

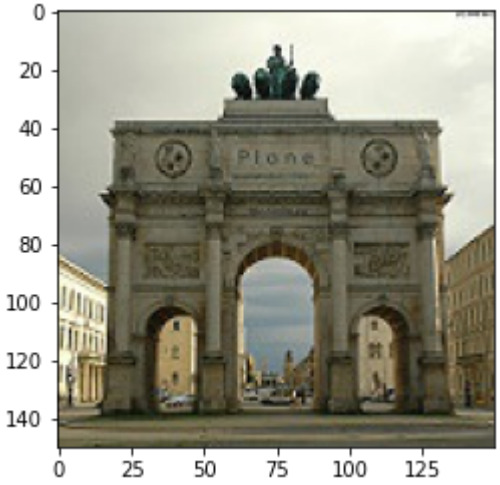
```
plt.plot(epochs, acc, 'r', label='Training accuracy')
plt.plot(epochs, val_acc, 'b', label='Validation accuracy')
plt.title('Training and validation accuracy')
plt.legend()
plt.figure()

plt.plot(epochs, loss, 'r', label='Training Loss')
plt.plot(epochs, val_loss, 'b', label='Validation Loss')
plt.title('Training and validation loss')
plt.legend()

plt.show()
```




```
import numpy as np
from keras.preprocessing import image

file_path =  '/content/gdrive/MyDrive/Colab-Notebooks/Semester 3/CV/Asgt 3/Dataset/Manu

test_image = image.load_img(file_path, target_size = (150, 150))
plt.imshow(test_image)
test_image = image.img_to_array(test_image)
test_image = np.expand_dims(test_image, axis = 0)
result = model.predict(test_image)

classes=train_generator.class_indices
```

```
14 classes=['buildings','forest','glacier','mountain','sea','street']
15
16 print("Prediction for this image is :",classes[np.argmax(result)])
```

```
Prediction for this image is : buildings
```




```
1
```