


```
import numpy as np
import tensorflow as tf
import matplotlib.pyplot as plt
```

✓ Creating directory for training data


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

```


 Mounted at /content/drive/

✓ Data preprocessing

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
trg = ImageDataGenerator(rescale=1/255)
tgenerator = trg.flow_from_directory(
    '/content/drive/MyDrive/data/train',
    target_size=(150, 150),
    batch_size=32,
    class_mode='categorical' # Ensure labels are categorical
)
```

 Found 1951 images belonging to 4 classes.

```
num_classes = len(tgenerator.class_indices)
print(f"Number of classes: {num_classes}")
```

 Number of classes: 4

```
xb1, yb1 = next(iter(tgenerator))
```

✓ Load InceptionV3

```
bm = tf.keras.applications.InceptionV3(input_shape=(150, 150, 3), include_top=False, weights='imagenet')
bm.trainable = False
```

 Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/inception_v3/inception_v3_weights_tf_dim_ordering_tf_87910968/87910968 [=====] - 1s 0us/step

```
xb = bm(xb1)
```

```
p1 = tf.keras.layers.Dense(num_classes)
global_average_layer = tf.keras.layers.GlobalAveragePooling2D()
feature_batch_average = global_average_layer(xb)
prediction_batch = p1(feature_batch_average)
```

```
feature_batch_average.shape
```

 TensorShape([32, 2048])

```
inp = tf.keras.Input(shape=(150, 150, 3))
x = bm(inp, training=False)
x = global_average_layer(x)
x = tf.keras.layers.Dropout(0.2)(x)
out = p1(x)
m = tf.keras.Model(inp, out)
```

```
m.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=0.001),
          loss=tf.keras.losses.CategoricalCrossentropy(from_logits=True),
          metrics=['accuracy'])
```

```
m.fit(tgenerator, epochs=5)
```

```
Epoch 1/5
61/61 [=====] - 458s 7s/step - loss: 0.6594 - accuracy: 0.7611
Epoch 2/5
61/61 [=====] - 86s 1s/step - loss: 0.3824 - accuracy: 0.8631
Epoch 3/5
61/61 [=====] - 87s 1s/step - loss: 0.2989 - accuracy: 0.9031
Epoch 4/5
61/61 [=====] - 86s 1s/step - loss: 0.2530 - accuracy: 0.9190
Epoch 5/5
61/61 [=====] - 86s 1s/step - loss: 0.2413 - accuracy: 0.9226
<keras.src.callbacks.History at 0x7fe81c64b1c0>
```

```
m.save('model.h5')
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/engine/training.py:3103: UserWarning: You are saving your model as an HDF5 file via `
saving_api.save_model(
```

```
m.summary()
```

```
Model: "model"
```

Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[(None, 150, 150, 3)]	0
inception_v3 (Functional)	(None, 3, 3, 2048)	21802784
global_average_pooling2d (GlobalAveragePooling2D)	(None, 2048)	0
dropout (Dropout)	(None, 2048)	0
dense (Dense)	(None, 4)	8196

```

Total params: 21810980 (83.20 MB)
Trainable params: 8196 (32.02 KB)
Non-trainable params: 21802784 (83.17 MB)

```

```
label = tgenerator.classes
len(label)
```

```
1951
```

```
label = ['Ascochyta Blight', 'Bacteria blight', 'Fresh leaf', 'Target spot', 'Cercospora']
```

```
tsg = ImageDataGenerator(rescale=1/255)
tsgenerator = trg.flow_from_directory(
    '/content/drive/MyDrive/test',
    target_size=(150, 150)
)
```

```
Found 36 images belonging to 5 classes.
```

```
q, _ = next(tsgenerator)
pred = m.predict(q)
```

```
1/1 [=====] - 3s 3s/step
```

```
a, b = label[np.argmax(pred[0])], label[np.argmax(pred[1])]
```

```
plt.subplot(2, 2, 1)
plt.imshow(q[0])
plt.xlabel(a)
plt.subplot(2, 2, 2)
plt.imshow(q[1])
plt.xlabel(b)
```

 Text(0.5, 0, 'Ascochyta Blight')

