**APPENDIX**

from google.colab import drive

drive.mount('/content/drive')

!unzip /content/drive/MyDrive/archive.zip\_dl=0

import numpy as np

import matplotlib.pyplot as plt

import keras

import pandas

from keras.preprocessing.image import img\_to\_array

import os

from keras.preprocessing.image import load\_img

from keras.preprocessing.image import ImageDataGenerator

from keras.applications.vgg19 import VGG19,preprocess\_input,decode\_predictions

training\_data\_generator= ImageDataGenerator(zoom\_range=0.5, shear\_range=0.3, rescale=1/255, horizontal\_flip=True)

validation\_data\_generator= ImageDataGenerator(rescale= 1/255)

train = training\_data\_generator.flow\_from\_directory(directory="/content/new plant diseases dataset(augmented)/New Plant Diseases Dataset(Augmented)/train",target\_size=(256,256),batch\_size=32)

val = validation\_data\_generator.flow\_from\_directory(directory="/content/new plant diseases dataset(augmented)/New Plant Diseases Dataset(Augmented)/valid",target\_size=(256,256),batch\_size=32)

from keras.layers import Dense, Flatten

from keras.models import Model

from keras.applications.vgg19 import VGG19

import keras

base\_model =VGG19(input\_shape=(256,256,3),include\_top=False)

for layer in base\_model.layers:

layer.trainable=False

x =Flatten()(base\_model.output)

x= Dense(units=38, activation='softmax')(x)

model =Model(base\_model.input, x)

model.compile(optimizer='adam',loss=keras.losses.categorical\_crossentropy,metrics=['accuracy'])

from keras.callbacks import ModelCheckpoint, EarlyStopping

es =EarlyStopping(monitor='val\_accuracy',min\_delta=0.01,patience=3,verbose=1)

mc =ModelCheckpoint(filepath="best\_model.h",monitor='val\_accuracy',min\_delta=0.01,patience=3,verbose=1,save\_best\_only=True)

cb=[es,mc]

his = model.fit\_generator(train,steps\_per\_epoch=16,epochs=50,verbose=1,callbacks=cb,validation\_data=val,validation\_steps=16)

from keras.callbacks import ModelCheckpoint, EarlyStopping

es =EarlyStopping(monitor='val\_accuracy',min\_delta=0.01,patience=3,verbose=1)

mc =ModelCheckpoint(filepath="best\_model.h",monitor='val\_accuracy',min\_delta=0.01,patience=3,verbose=1,save\_best\_only=True)

cb=[es,mc]

from keras.models import load\_model

model=load\_model('/content/best\_model.h')

acc =model.evaluate\_generator(val)[1]

print(acc)

ref=dict(zip(list(train.class\_indices.values()),list(train.class\_indices.keys())))

def prediction(path):

img=load\_img(path,target\_size=(256,256))

i=img\_to\_array(img)

im=preprocess\_input(i)

img=np.expand\_dims(im,axis=0)

pred =np.argmax(model.predict(img))

print(pred)

print(f"The plant diagnosed as{ref[pred]}")

path="/content/drive/MyDrive/precaution/"+f'{pred}'+".txt"

f=open(path)

print(f.read())