Tomato Leaf Health Dataset

The data analysis and modeling will be done for the tomato leaf health dataset, which can be found here Tomato Leaf Health Data.

Introduction

My intention for this project is to demonstrate a methodology for generating an accurate model for problem. In this case taking images of tomato leaves and classifying them into one of eleven buckets; one healthy and the rest diseased in various ways. I will start with a small convolutional neural network (CNN), see how it performs, and add data augmentation to see if that improves its accuracy. Afterwards, I will use pretrained image classification models with transfer learning and fine-tuning to see if I can get a model that is substantially more accurate than the original CNN. Finally, after obtaining a better accuracy model from the transfer learning approach I will revisit the CNN and try training using knowledge distillation where the transfer learning model will work as the teacher, and the CNN will function as the student. Hopefully this will result in a much smaller model with improved accuracy. The purpose of doing this is to try and find a model that is small enough for offline edge devices to use that is also accurate enough to be useful.

Exploratory Data Analysis

In [14]:
```
import collections
import os

import keras.losses
from PIL import Image
from keras import backend
from keras import models
from keras import layers
from keras import optimizers
from keras.utils import to_categorical
import numpy as np
import matplotlib.pyplot as plt
from keras import regularizers
from keras.layers import BatchNormalization
from keras.callbacks import EarlyStopping
from sklearn.model_selection import train_test_split
from keras.utils import to_categorical
from PIL import Image
from keras.preprocessing.image import ImageDataGenerator
from keras.preprocessing import image
import os, shutil
np.random.seed(1)
```

In [15]:
```
# Data Details for EDA
def describe_data():
```

```
    train_path = "../data/train"
    test_path = "../data/valid"
    img_shapes = []
    training_files = get_file_count(train_path, img_shapes)
    print(f'Train Count: {training_files}')  # 69h X 130w
    test_files = get_file_count(test_path, img_shapes)
    print(f'Test Count: {test_files}')  # 100h X 150w

    test_x, test_y = img_shapes[0]
    print(f'{test_x}x{test_y}')
    x_avg, x_min, x_max = 0, 10000, 0
    y_avg, y_min, y_max = 0, 10000, 0

    x_s = []
    y_s = []

    for x, y in img_shapes:
        x_s.append(x)
        x_avg += x
        x_min = min(x_min, x)
        x_max = max(x_max, x)
        y_s.append(y)
        y_avg += y
        y_min = min(y_min, y)
        y_max = max(y_max, y)

    x_mode = collections.Counter(x_s).most_common()[0][0]
    y_mode = collections.Counter(y_s).most_common()[0][0]

    x_avg = x_avg // len(img_shapes)
    y_avg = y_avg // len(img_shapes)

    print(f'Avg: {x_avg}x{y_avg}')
    print(f'Mode: {x_mode}x{y_mode}')
    print(f'Min: {x_min}x{y_min}')
    print(f'Max: {x_max}x{y_max}')
```

In [16]:
```
def get_file_count(directory, shapes):
    count = 0
    for path in os.listdir(directory):
        if os.path.isfile(os.path.join(directory, path)):
            im = Image.open(os.path.join(directory, path))
            shapes.append(im.size)
            count += 1
        if os.path.isdir(os.path.join(directory, path)):
            add = get_file_count(os.path.join(directory, path), shapes)
            count += add
    return count
```

In [17]:
```
describe_data()
```

```
Train Count: 25848
Test Count: 6683
256x256
Avg: 292x303
Mode: 256x256
Min: 130x69
Max: 6000x6000
```

Looks like our training set is over 25k images, and that the most common image size is 256 by 256. I will start using that as my input shape, and we can adjust if needed.

In [27]:
```
categories = ['Bacterial_spot', 'Early_blight', 'healthy', 'Late_blight', 'Leaf_Mold',
              'powdery_mildew', 'Septoria_leaf_spot', 'Spider_mites Two-spotted_spider_m
              'Target_Spot', 'Tomato_mosaic_virus', 'Tomato_Yellow_Leaf_Curl_Virus']
train_dir = "../data/train"
test_dir = "../data/valid"
val_dir = "../data/val"
```

In [21]:
```
# Only run once
for category in categories:
  os.makedirs(val_dir + category)
  print(os.listdir(val_dir + category))
  files = os.listdir(train_dir + category)
  np.random.shuffle(files)
  num_val_samples = int(0.2*len(files))
  val_files = files[-num_val_samples:]
  for fname in val_files:
    shutil.move("../data/train/" + category + "/" + fname,
                "../data/val/" + category + "/" + fname)
```

```
---------------------------------------------------------------------------
FileExistsError                           Traceback (most recent call last)
Cell In [21], line 2
      1 for category in categories:
----> 2   os.makedirs("../data/val/" + category)
      3   print(os.listdir("../data/val/" + category))
      4   files = os.listdir("../data/train/" + category)

File ~\anaconda3\envs\deep-learning-cv\lib\os.py:225, in makedirs(name, mode, exist_o
k)
    223         return
    224 try:
--> 225     mkdir(name, mode)
    226 except OSError:
    227     # Cannot rely on checking for EEXIST, since the operating system
    228     # could give priority to other errors like EACCES or EROFS
    229     if not exist_ok or not path.isdir(name):

FileExistsError: [WinError 183] Cannot create a file when that file already exists:
'../data/val/Bacterial_spot'
```

In [23]:
```
# Build Category Size Display Function
def describe_categories(cats, directory):
    for cat in cats:
        full_dir = os.path.join(directory, cat)
        print(len(os.listdir(full_dir)))
```

In [24]:
```
# Checking the number of files in each class directory
describe_categories(categories, train_dir)
```

```
2261
1964
2441
2491
2204
804
2306
1398
1462
1723
1629
```

In [28]:
```python
describe_categories(categories, val_dir)
```

```
565
491
610
622
550
200
576
349
365
430
407
```

In [26]:
```python
describe_categories(categories, test_dir)
```

```
732
643
805
792
739
252
746
435
457
584
498
```

The distribution of the data appears to be fairly consistent, with a small outlier for the powdery mildew class. It likely isn't enough to cause problems, but if we find ourselves unable to classify correctly we should revisit.

In [29]:
```python
train_datagen = ImageDataGenerator(rescale=1./255)
test_datagen = ImageDataGenerator(rescale=1./255)
```

In [30]:
```python
train_generator = train_datagen.flow_from_directory(
    train_dir,
    target_size=(256, 256), # Choosing our mode image values
    batch_size=20,
    class_mode='categorical') # Since we have 11 classes as output

val_generator = train_datagen.flow_from_directory(
    val_dir,
    target_size=(256, 256),
    batch_size=20,
    class_mode='categorical')
```

```
test_generator = test_datagen.flow_from_directory(
    test_dir,
    target_size=(256, 256),
    batch_size=20,
    class_mode='categorical')
```

```
Found 20683 images belonging to 11 classes.
Found 5165 images belonging to 11 classes.
Found 6683 images belonging to 11 classes.
```

Now we have a validation set, and all 11 classes are represented. We will want to make sure that our neural network that we design has a softmax activation function, and 11 outputs.

In [40]:

```
# Build a plotting function
def plot_history(history):
  history_dict = history.history
  loss_values = history_dict['loss'] or history_dict['student_loss']
  val_loss_values = history_dict['val_loss'] or history_dict['val_student_loss']
  acc_values = history_dict['accuracy'] or history_dict['categorical_accuracy']
  val_acc_values = history_dict['val_accuracy'] or history_dict['val_categorical_accur
  epochs = range(1, len(history_dict['accuracy']) + 1)

  plt.plot(epochs, loss_values, 'bo', label = 'Training loss')
  plt.plot(epochs, val_loss_values, 'b', label = 'Validation loss')
  plt.title('Training and validation loss')
  plt.xlabel('Epochs')
  plt.ylabel('Loss')
  plt.legend()
  plt.show()

  plt.plot(epochs, acc_values, 'bo', label = 'Training accuracy')
  plt.plot(epochs, val_acc_values, 'b', label = 'Validation accuracy')
  plt.title('Training and validation accuracy')
  plt.xlabel('Epochs')
  plt.ylabel('Accuracy')
  plt.legend()
  return plt.show()
```

Deep Learning Model

I will be going down the path of experimenting with deep learning models, as they tend to work well on computer vision problems. I will start with a fairly classic architecture for my model with a sequential model using convolutional layers, max pooling, batch normalization into a top that flattens into dense layers with dropout to our softmax classifier. I will be using the Adam optimizer for scheduling, categorical crossentropy for loss, and accuracy for our metric. We could tune these hyperparameters but the proof of concept that I am looking for is taking a fairly small and inaccurate model and through knowledge distillation increasing its accuracy while keeping it small.

In [41]:

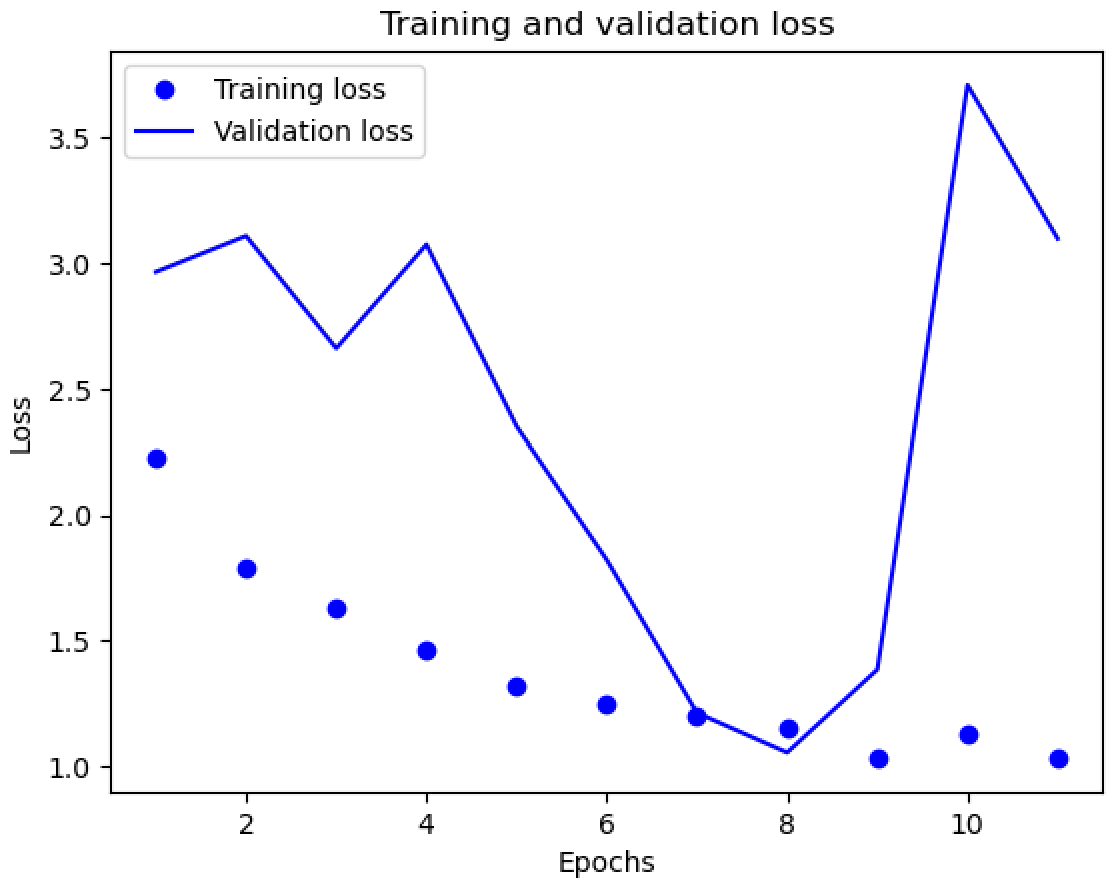
```
# Build a model
def Base_CNN():
  backend.clear_session()
  model = models.Sequential()
```

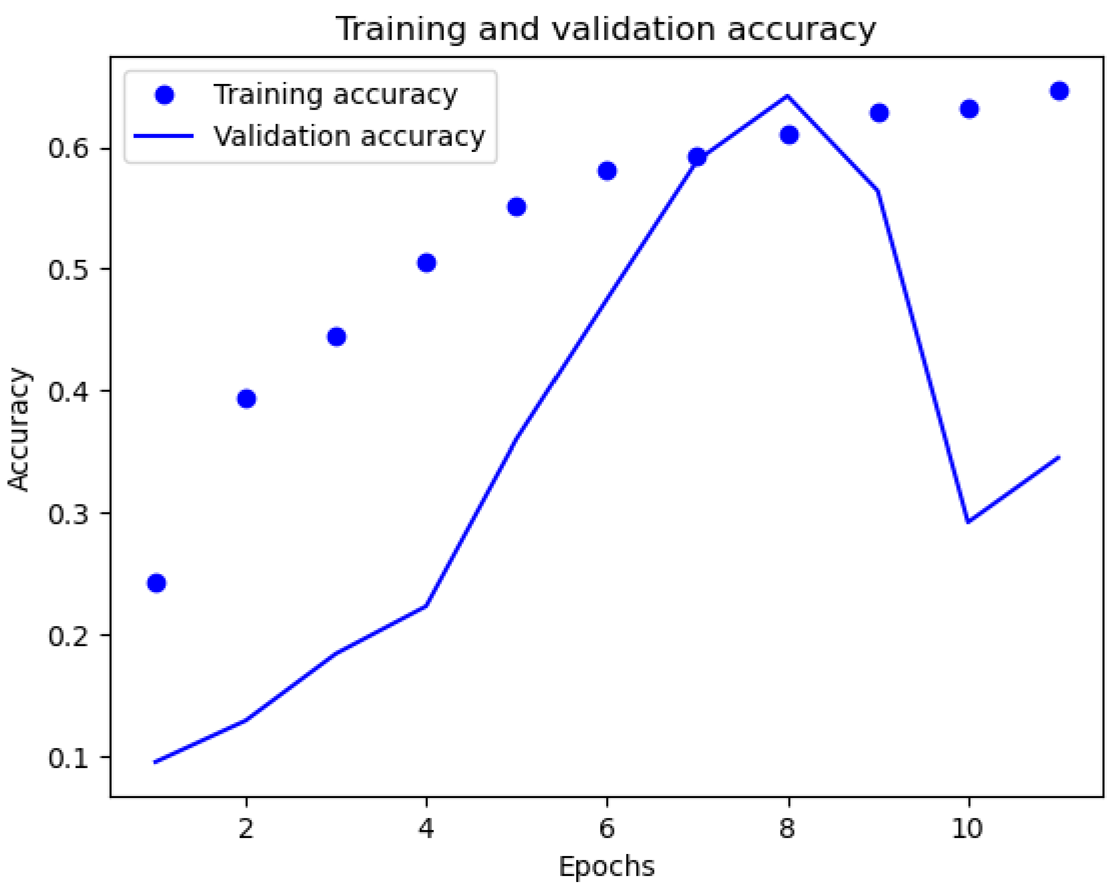
```
    model.add(layers.Conv2D(32, (3,3), activation = 'relu', input_shape = (256, 256, 3))
    model.add(layers.MaxPool2D((2,2)))
    model.add(BatchNormalization())

    model.add(layers.Conv2D(32, (3,3), activation = 'relu'))
    model.add(layers.MaxPool2D((2,2)))
    model.add(BatchNormalization())

    model.add(layers.Conv2D(32, (3,3), activation = 'relu'))
    model.add(layers.MaxPool2D((2,2)))
    model.add(BatchNormalization())

    model.add(layers.Conv2D(32, (3,3), activation = 'relu'))
    model.add(layers.MaxPool2D((2,2)))
    model.add(BatchNormalization())

    model.add(layers.Flatten())
    model.add(layers.Dense(64, activation='relu'))
    model.add(layers.Dense(64, activation='relu'))
    model.add(layers.Dense(64, activation='relu'))
    model.add(layers.Dropout(0.5))

    model.add(layers.Dense(11, activation='softmax'))

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

In [42]:
```
model = Base_CNN()
```

In [43]:
```
history = model.fit( # The image data must come from the image generator that takes th
    train_generator,
    steps_per_epoch=100,
    epochs=100,
    validation_data=val_generator,
    validation_steps=50,
    verbose = 1,
    callbacks=[EarlyStopping(monitor='val_accuracy', patience=5, restore_best_weights

plot_history(history) # Use our plot function to plot the loss and accuracy.

test_loss, test_acc =model.evaluate(test_generator, steps = 100)

print('test_acc:', test_acc)
```

```
Epoch 1/100
100/100 [==============================] - 23s 221ms/step - loss: 2.2259 - accuracy:
0.2430 - val_loss: 2.9652 - val_accuracy: 0.0950
Epoch 2/100
100/100 [==============================] - 22s 220ms/step - loss: 1.7937 - accuracy:
0.3940 - val_loss: 3.1084 - val_accuracy: 0.1290
Epoch 3/100
100/100 [==============================] - 22s 219ms/step - loss: 1.6326 - accuracy:
0.4450 - val_loss: 2.6607 - val_accuracy: 0.1840
Epoch 4/100
100/100 [==============================] - 23s 228ms/step - loss: 1.4627 - accuracy:
0.5050 - val_loss: 3.0743 - val_accuracy: 0.2230
Epoch 5/100
100/100 [==============================] - 22s 218ms/step - loss: 1.3240 - accuracy:
0.5510 - val_loss: 2.3513 - val_accuracy: 0.3600
Epoch 6/100
100/100 [==============================] - 22s 219ms/step - loss: 1.2499 - accuracy:
0.5810 - val_loss: 1.8223 - val_accuracy: 0.4750
Epoch 7/100
100/100 [==============================] - 22s 218ms/step - loss: 1.1999 - accuracy:
0.5930 - val_loss: 1.2130 - val_accuracy: 0.5890
Epoch 8/100
100/100 [==============================] - 21s 213ms/step - loss: 1.1554 - accuracy:
0.6105 - val_loss: 1.0553 - val_accuracy: 0.6420
Epoch 9/100
100/100 [==============================] - 23s 228ms/step - loss: 1.0341 - accuracy:
0.6295 - val_loss: 1.3857 - val_accuracy: 0.5640
Epoch 10/100
100/100 [==============================] - 22s 220ms/step - loss: 1.1286 - accuracy:
0.6315 - val_loss: 3.7098 - val_accuracy: 0.2920
Epoch 11/100
100/100 [==============================] - 22s 222ms/step - loss: 1.0355 - accuracy:
0.6470 - val_loss: 3.0973 - val_accuracy: 0.3450
```






```
100/100 [==============================] - 8s 84ms/step - loss: 1.0379 - accuracy: 0.
6475
test_acc: 0.6474999785423279
```

Looks like our accuracy peaked at almost 65% and quickly fell off. We could extend the patience for a few more epochs to see if it recovers but we can move on to trying some data augmentation first.

In [51]:
```
model_path = 'tomato_leaf_classifier_base_cnn.h5'
model.save(model_path)
```

In [52]:
```
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d (Conv2D)             (None, 254, 254, 32)      896       
                                                                 
 max_pooling2d (MaxPooling2D  (None, 127, 127, 32)     0         
 )                                                               
                                                                 
 batch_normalization (BatchN  (None, 127, 127, 32)     128       
 ormalization)                                                   
                                                                 
 conv2d_1 (Conv2D)           (None, 125, 125, 32)      9248      
                                                                 
 max_pooling2d_1 (MaxPooling  (None, 62, 62, 32)       0         
 2D)                                                             
                                                                 
 batch_normalization_1 (Batc  (None, 62, 62, 32)       128       
 hNormalization)                                                 
                                                                 
 conv2d_2 (Conv2D)           (None, 60, 60, 32)        9248      
                                                                 
 max_pooling2d_2 (MaxPooling  (None, 30, 30, 32)       0         
 2D)                                                             
                                                                 
 batch_normalization_2 (Batc  (None, 30, 30, 32)       128       
 hNormalization)                                                 
                                                                 
 conv2d_3 (Conv2D)           (None, 28, 28, 32)        9248      
                                                                 
 max_pooling2d_3 (MaxPooling  (None, 14, 14, 32)       0         
 2D)                                                             
                                                                 
 batch_normalization_3 (Batc  (None, 14, 14, 32)       128       
 hNormalization)                                                 
                                                                 
 flatten (Flatten)           (None, 6272)              0         
                                                                 
 dense (Dense)               (None, 64)                401472    
                                                                 
 dense_1 (Dense)             (None, 64)                4160      
                                                                 
 dense_2 (Dense)             (None, 64)                4160      
                                                                 
 dropout (Dropout)           (None, 64)                0         
                                                                 
 dense_3 (Dense)             (None, 11)                715       
                                                                 
=================================================================
Total params: 439,659
Trainable params: 439,403
Non-trainable params: 256
_________________________________________________________________
```

In [55]:

```python
# Size the h5 files, we can also change data type to make it smaller before inference
def model_size(path):
    size = os.path.getsize(path) / 1000000
    print(f'Model is approx {size} MB')
```

In [56]:
```
model_size(model_path)
```

```
Model is approx 5.387932 MB
```

Adding Data Augmentation

We likely have enough images that it won't change our accuracy much using data augmentation, but perhaps we can improve our fit, lets continue training our model on the augmented dataset and see if it does anything.

In [58]:
```
datagen = ImageDataGenerator(
    rotation_range=40,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    fill_mode='nearest')
```

First lets take a look at the data augmentation results for making additional images for classification.

In [ ]:
```
from tensorflow.keras.preprocessing import image
example_dir = os.path.join(train_dir, categories[0])
img = image.load_img(os.path.join(example_dir, os.listdir(example_dir)[1]), target_siz
x = image.img_to_array(img)
x = x.reshape((1,) + x.shape)
i = 0
for batch in datagen.flow(x, batch_size=1):
    plt.figure(i)
    imgplot = plt.imshow(image.array_to_img(batch[0]))
    i += 1
    if i % 4 == 0:
        break
plt.show()
```

This should give us a good range of rotations and leaf locations, so it should at least give us more variety to train on.

In [64]:
```
# Apply the data augmentation to our data.
train_datagen2 = ImageDataGenerator(
    rescale=1./255,
    rotation_range=40,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True)

test_datagen2 = ImageDataGenerator(rescale=1./255) #Never apply data augmentation to t

train_generator2 = train_datagen2.flow_from_directory(
    train_dir,
    target_size=(256, 256),
    batch_size=20,
```

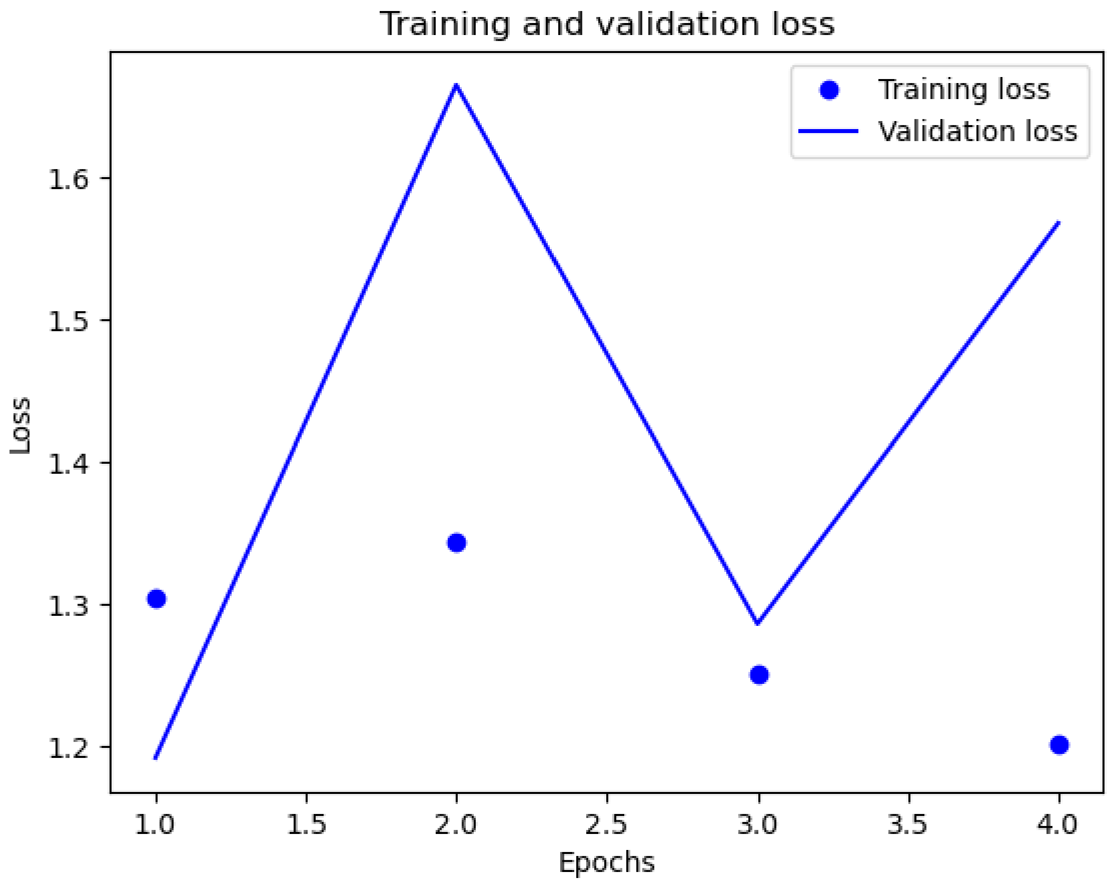
```
    class_mode='categorical')

validation_generator2 = train_datagen2.flow_from_directory(
    val_dir,
    target_size=(256, 256),
    batch_size=20,
    class_mode='categorical')

test_generator2 = test_datagen2.flow_from_directory( # Resize test data
    test_dir,
    target_size=(256, 256),
    batch_size=20,
    class_mode='categorical')
```

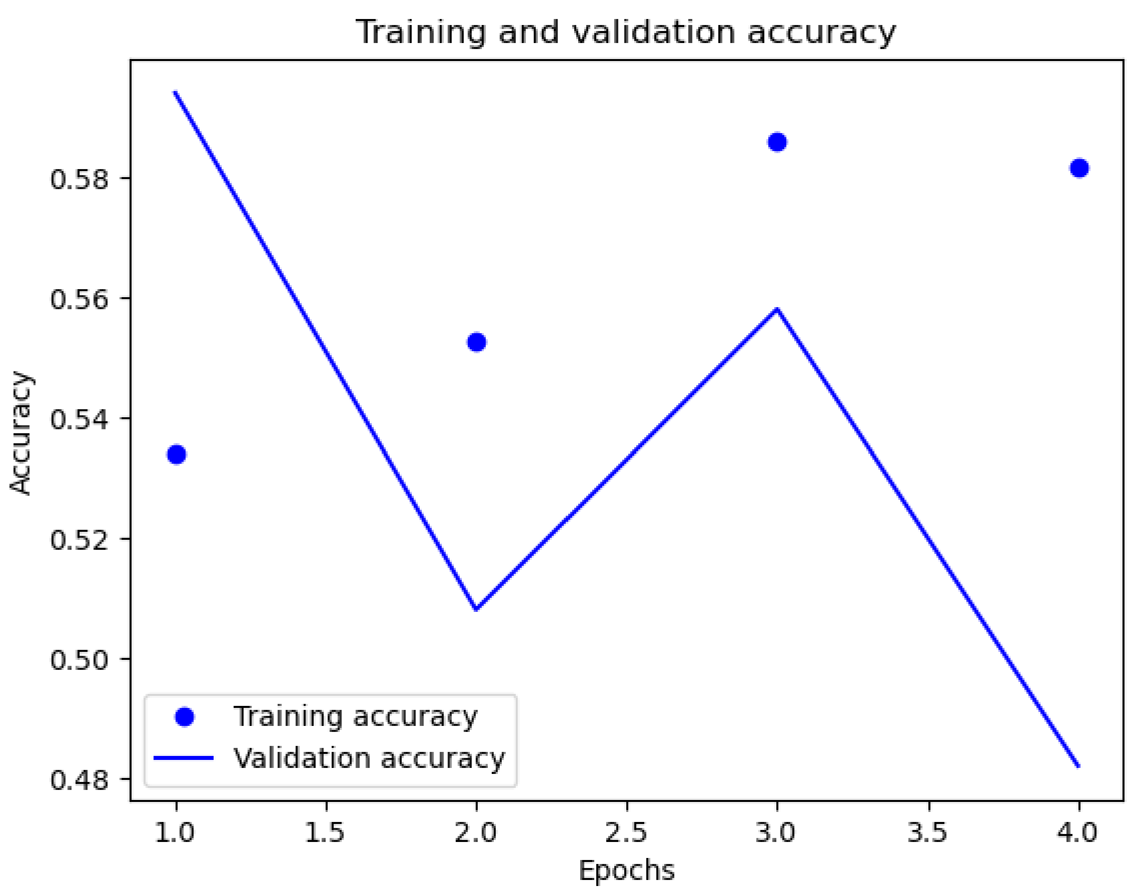
```
Found 20683 images belonging to 11 classes.
Found 5165 images belonging to 11 classes.
Found 6683 images belonging to 11 classes.
```

In [66]:

```
history = model.fit(
    train_generator2,
    steps_per_epoch=100,
    epochs=100,
    validation_data=validation_generator2,
    validation_steps=50,
    verbose = 1,
    callbacks=[EarlyStopping(monitor='val_accuracy', patience=5, restore_best_weights


plot_history(history)

test_loss, test_acc = model.evaluate(test_generator2, steps = 100)
print('test_acc:', test_acc)
model_path = "tomato_leaf_classifier_augmented_cnn.h5"
model.save(model_path)
```

```
Epoch 1/100
100/100 [==============================] - 55s 553ms/step - loss: 1.3046 - accuracy:
0.5340 - val_loss: 1.1921 - val_accuracy: 0.5940
Epoch 2/100
100/100 [==============================] - 55s 555ms/step - loss: 1.3434 - accuracy:
0.5525 - val_loss: 1.6648 - val_accuracy: 0.5080
Epoch 3/100
100/100 [==============================] - 55s 554ms/step - loss: 1.2509 - accuracy:
0.5860 - val_loss: 1.2862 - val_accuracy: 0.5580
Epoch 4/100
100/100 [==============================] - 55s 548ms/step - loss: 1.2020 - accuracy:
0.5815 - val_loss: 1.5677 - val_accuracy: 0.4820
```






```
100/100 [==============================] - 9s 88ms/step - loss: 1.0232 - accuracy: 0.
6445
test_acc: 0.6445000171661377
```

Our network immediately exited out from patience, so it does not look like we will be getting an improvement by using data augmentation. Next I will make next size up version of our CNN, which we can use for testing our knowledge distillation methods at the end. We won't go through testing it now as it is meant as something we can scale up to later if our small model cannot get us the desired accuracy.

In [151...

```
def Next_CNN():
  backend.clear_session()
  model = models.Sequential()

  model.add(layers.Conv2D(64, (3,3), activation = 'relu', input_shape = (256, 256, 3))
  model.add(layers.MaxPool2D((2,2)))
  model.add(BatchNormalization())

  model.add(layers.Conv2D(64, (3,3), activation = 'relu'))
  model.add(layers.MaxPool2D((2,2)))
  model.add(BatchNormalization())

  model.add(layers.Conv2D(64, (3,3), activation = 'relu'))
  model.add(layers.MaxPool2D((2,2)))
  model.add(BatchNormalization())

  model.add(layers.Conv2D(64, (3,3), activation = 'relu'))
  model.add(layers.MaxPool2D((2,2)))
  model.add(BatchNormalization())

  model.add(layers.Flatten())
  model.add(layers.Dense(128, activation='relu'))
  model.add(layers.Dense(128, activation='relu'))
  model.add(layers.Dense(128, activation='relu'))
  model.add(layers.Dropout(0.5))

  model.add(layers.Dense(11, activation='softmax'))

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

Transfer Learning Models

Here we will try using a few different pretrained models fine-tuned to our specific problem. The problem involves a lot more nuanced detail differences of leaves for the same plant. Hopefully that is not too narrow for this approach to work. My plan is to try our a few different sizes and types of pretrained image classifiers: ResNet, EfficientNet, and DenseNet in particular to try and get a higher accuracy prediction since our classic CNN only gave us 64% accuracy.

In [120...

```
from keras.applications import VGG16 # Tested and dropped due to accuracy
from keras.applications import ResNet152V2
from keras.applications import ResNet50 # Tested and dropped due to accuracy
from keras.applications import EfficientNetB0
from keras.applications import EfficientNetB7 # Tested and dropped due to accuracy
from keras.applications import DenseNet201
```

In [91]:
```
# Apply the data augmentation to our data for transfer learning case, again these are
train_datagen3 = ImageDataGenerator(
    rescale=1./255,
    rotation_range=10,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.02,
    zoom_range=0.1,
    horizontal_flip=True,
    vertical_flip=True)

test_datagen3 = ImageDataGenerator(rescale=1./255)

train_generator3 = train_datagen3.flow_from_directory(
    train_dir,
    target_size=(256, 256),
    batch_size=20,
    class_mode='categorical')

validation_generator3 = train_datagen3.flow_from_directory(
    val_dir,
    target_size=(256, 256),
    batch_size=20,
    class_mode='categorical')

test_generator3 = test_datagen3.flow_from_directory(
    test_dir,
    target_size=(256, 256),
    batch_size=20,
    class_mode='categorical')
```

```
Found 20683 images belonging to 11 classes.
Found 5165 images belonging to 11 classes.
Found 6683 images belonging to 11 classes.
```

In [74]:
```
# Now we can freeze all the resnet weights except the last few, and train those before
backend.clear_session()
resnet_base = ResNet152V2(weights = 'imagenet', include_top = False, input_shape = (25

# Here we freeze all the layers except the last 4.
for layer in resnet_base.layers[:-4]:
  layer.trainable = False
for layer in resnet_base.layers:
  print(layer, layer.trainable)
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/re
snet/resnet152v2_weights_tf_dim_ordering_tf_kernels_notop.h5
234545216/234545216 [==============================] - 4s 0us/step
<keras.engine.input_layer.InputLayer object at 0x000001CFC4D7B8E0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCB90B6D0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB90BA90> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCB90A560> Fa
lse
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x000001CFCB944C40> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB946080> False
<keras.layers.core.activation.Activation object at 0x000001CFCB9449A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB9465F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB9466B0> False
<keras.layers.core.activation.Activation object at 0x000001CFCB947E20> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCB94D840> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFC4D1DBA0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB94D030> False
<keras.layers.core.activation.Activation object at 0x000001CFF55A4E20> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB9471C0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFFA5BB9D0> False
<keras.layers.merging.add.Add object at 0x000001CFCB8E57B0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFF181ABC0> False
<keras.layers.core.activation.Activation object at 0x000001CFFA5B82B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB94DCC0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFFA5B8040> False
<keras.layers.core.activation.Activation object at 0x000001CFCB947640> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCB955ED0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFFA5B8250> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB957A90> False
<keras.layers.core.activation.Activation object at 0x000001CFCB8E5780> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB963580> False
<keras.layers.merging.add.Add object at 0x000001CFCB961390> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB961F00> False
<keras.layers.core.activation.Activation object at 0x000001CFCB9545B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB963610> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB963C10> False
<keras.layers.core.activation.Activation object at 0x000001CFCB9634F0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCB960F40> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB9561A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB977910> False
<keras.layers.core.activation.Activation object at 0x000001CFCB977160> False
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x000001CFCB963EB0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB996C80> False
<keras.layers.merging.add.Add object at 0x000001CFCB9958A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB997B80> False
<keras.layers.core.activation.Activation object at 0x000001CFCB996050> False
```

```
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB996620> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB994100> False
<keras.layers.core.activation.Activation object at 0x000001CFCB974E80> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCB963B80> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB976950> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB9942E0> False
<keras.layers.core.activation.Activation object at 0x000001CFCB9627A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB995C00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB954C10> False
<keras.layers.merging.add.Add object at 0x000001CFCB94DD20> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB995CF0> False
<keras.layers.core.activation.Activation object at 0x000001CFCB94F760> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB9B33D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB9B3520> False
<keras.layers.core.activation.Activation object at 0x000001CFCB9B3AF0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD571E40> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB995F90> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD573280> False
<keras.layers.core.activation.Activation object at 0x000001CFCB9B20B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD5732B0> False
<keras.layers.merging.add.Add object at 0x000001CFCD573EE0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD591750> False
<keras.layers.core.activation.Activation object at 0x000001CFCD590160> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD592680> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD592770> False
<keras.layers.core.activation.Activation object at 0x000001CFCD593E20> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCB90A6E0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD590C70> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD5722F0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD592440> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB9B1300> False
<keras.layers.merging.add.Add object at 0x000001CFCD570C10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB955BA0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD5729E0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB975B70> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB962F50> False
<keras.layers.core.activation.Activation object at 0x000001CFCB94F700> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCB9B12A0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD593DF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD5B50C0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD5B6EF0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD5D1240> False
<keras.layers.merging.add.Add object at 0x000001CFCD5D12A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD5D01C0> False
```

```
<keras.layers.core.activation.Activation object at 0x000001CFCD5B59C0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD5D3F10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD5D2FB0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD5D2B60> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD5E9750> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD5B6E60> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD5E8BE0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD5E8430> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD5B7820> False
<keras.layers.merging.add.Add object at 0x000001CFCD5B7D00> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD5E89D0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD5B6590> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB909F00> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB9B05B0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD5D0310> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCB9B33A0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD5E9600> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD60AD70> False
<keras.layers.core.activation.Activation object at 0x000001CFCB974B20> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD60A920> False
<keras.layers.merging.add.Add object at 0x000001CFCD608250> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD61D300> False
<keras.layers.core.activation.Activation object at 0x000001CFCD60BE80> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD61CA90> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD61EFB0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD61E7D0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD61ED70> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB974D30> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD631930> False
<keras.layers.core.activation.Activation object at 0x000001CFCD61CD90> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD6318A0> False
<keras.layers.merging.add.Add object at 0x000001CFCD6337F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD6589A0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD65BA60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD658DC0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD65BC40> False
<keras.layers.core.activation.Activation object at 0x000001CFCD65A680> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD658190> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD61DFF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD5B7CA0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD658100> False
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x000001CFCD65A5F0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD60BDC0> False
<keras.layers.merging.add.Add object at 0x000001CFCD572F80> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
```

```
CFCD630FA0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD608400> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD630F10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD673640> False
<keras.layers.core.activation.Activation object at 0x000001CFCD633430> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD6736D0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD6583A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD671990> False
<keras.layers.core.activation.Activation object at 0x000001CFCD673370> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD671870> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD691750> False
<keras.layers.merging.add.Add object at 0x000001CFCD6729E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD691FF0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD671660> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD6913C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD6B0EB0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD6B15A0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD6B2AD0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD6B0F40> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD6B2560> False
<keras.layers.core.activation.Activation object at 0x000001CFCD6B3670> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD6B0E20> False
<keras.layers.merging.add.Add object at 0x000001CFCD6B3AF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD6B2950> False
<keras.layers.core.activation.Activation object at 0x000001CFCD5EAC80> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD631180> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD671780> False
<keras.layers.core.activation.Activation object at 0x000001CFCD630DF0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD6926E0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD6B0D90> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD6CD180> False
<keras.layers.core.activation.Activation object at 0x000001CFCD6B0EE0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD6EBD00> False
<keras.layers.merging.add.Add object at 0x000001CFCD6CD270> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD6E8FD0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD630AC0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD70EC50> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD6EB5B0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD6EB910> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD70D840> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD6CEDA0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD70F070> False
<keras.layers.core.activation.Activation object at 0x000001CFCD72FFD0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD72E4D0> False
<keras.layers.merging.add.Add object at 0x000001CFCD72D0C0> False
```

```
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD72CF70> False
<keras.layers.core.activation.Activation object at 0x000001CFCD70CCA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD70C640> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD70C4F0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD658D30> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD670040> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD61FF10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD72DF00> False
<keras.layers.core.activation.Activation object at 0x000001CFCD72E830> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD6CF550> False
<keras.layers.merging.add.Add object at 0x000001CFCD70C940> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD690880> False
<keras.layers.core.activation.Activation object at 0x000001CFCD692A70> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD74FAF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD74EA10> False
<keras.layers.core.activation.Activation object at 0x000001CFCD693100> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD74F730> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD631A50> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD74DDE0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD74C0A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD74ECE0> False
<keras.layers.merging.add.Add object at 0x000001CFCD76DAB0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD74F670> False
<keras.layers.core.activation.Activation object at 0x000001CFCF759F00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF758310> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF75B2B0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD76C5B0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD76D780> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD76ED10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD61EE00> False
<keras.layers.core.activation.Activation object at 0x000001CFCB9B2DA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF75A260> False
<keras.layers.merging.add.Add object at 0x000001CFCF758880> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD692CB0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF75BD30> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD72DD50> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD6E8730> False
<keras.layers.core.activation.Activation object at 0x000001CFCD60BD30> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF77F160> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF75A590> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF77E590> False
<keras.layers.core.activation.Activation object at 0x000001CFCD74D0F0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF7959F0> False
<keras.layers.merging.add.Add object at 0x000001CFCF77F5B0> False
```

```
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF7968C0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF77C100> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF795A50> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF797B80> False
<keras.layers.core.activation.Activation object at 0x000001CFCF796440> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF7B3A30> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD74E650> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF7B3280> False
<keras.layers.core.activation.Activation object at 0x000001CFCF7B1810> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF795780> False
<keras.layers.merging.add.Add object at 0x000001CFCF7972E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF796D40> False
<keras.layers.core.activation.Activation object at 0x000001CFCD6E9390> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD6E9B10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF77EB90> False
<keras.layers.core.activation.Activation object at 0x000001CFCF7B1030> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD76C130> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF7B2020> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF7583D0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF77D330> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF7CDE10> False
<keras.layers.merging.add.Add object at 0x000001CFCF7CD210> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF77F2B0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF7CF8E0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF7EE500> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF7EE860> False
<keras.layers.core.activation.Activation object at 0x000001CFCF7EEC20> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF7EFBB0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD70DC90> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF808D90> False
<keras.layers.core.activation.Activation object at 0x000001CFCF7EEDD0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF80AAD0> False
<keras.layers.merging.add.Add object at 0x000001CFCF80AFE0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF809F00> False
<keras.layers.core.activation.Activation object at 0x000001CFCF80B790> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8094B0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF758220> False
<keras.layers.core.activation.Activation object at 0x000001CFCD60ADD0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCD76C040> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF7EE800> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF794430> False
<keras.layers.core.activation.Activation object at 0x000001CFCF808C10> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF7EF850> False
<keras.layers.merging.add.Add object at 0x000001CFCF7CF6A0> False
```

```
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF825B70> False
<keras.layers.core.activation.Activation object at 0x000001CFCD74DB40> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8266E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF827BB0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF826500> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF827EE0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF808AF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF84FD60> False
<keras.layers.core.activation.Activation object at 0x000001CFCF827D60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF861480> False
<keras.layers.merging.add.Add object at 0x000001CFCF8614B0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF860700> False
<keras.layers.core.activation.Activation object at 0x000001CFCF84CE20> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF861E70> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF863B20> False
<keras.layers.core.activation.Activation object at 0x000001CFCF87C250> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF87FC70> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF87D2A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFA08E0E20> False
<keras.layers.core.activation.Activation object at 0x000001CFC31A0340> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFC317A620> False
<keras.layers.merging.add.Add object at 0x000001CFA0A72230> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFA090B310> False
<keras.layers.core.activation.Activation object at 0x000001CFC317A200> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFC4DA1240> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFF5567DC0> False
<keras.layers.core.activation.Activation object at 0x000001CFA090A170> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFC30FCFA0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFA0AAACE0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFA0AA9840> False
<keras.layers.core.activation.Activation object at 0x000001CFA0AA97E0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCB9455D0> False
<keras.layers.merging.add.Add object at 0x000001CFA0AA9BD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB946D40> False
<keras.layers.core.activation.Activation object at 0x000001CFCB9458D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFF672D570> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFC493CE20> False
<keras.layers.core.activation.Activation object at 0x000001CFF69DF640> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFA0C05E40> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF861270> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD6B1090> False
<keras.layers.core.activation.Activation object at 0x000001CFCD6B1810> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD65B9D0> False
<keras.layers.merging.add.Add object at 0x000001CFCD65AB30> False
```

```
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCB9969B0> False
<keras.layers.core.activation.Activation object at 0x000001CFC8FFA3E0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCD5B45E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD72CEB0> False
<keras.layers.core.activation.Activation object at 0x000001CFCD5B6B90> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF87E290> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFA0AAA530> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFA0A725C0> False
<keras.layers.core.activation.Activation object at 0x000001CFF69DD750> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFFA45DDE0> False
<keras.layers.merging.add.Add object at 0x000001CFA0C05A20> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFC94CEAA0> False
<keras.layers.core.activation.Activation object at 0x000001CFCB8E5660> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFC94CE8F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFC3E7DE10> False
<keras.layers.core.activation.Activation object at 0x000001CFC94CE980> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFC3E7F2B0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFFA5B8B20> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFC31BC4C0> False
<keras.layers.core.activation.Activation object at 0x000001CFC31BFAF0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF75B490> False
<keras.layers.merging.add.Add object at 0x000001CFCF75B6D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCD74F370> False
<keras.layers.core.activation.Activation object at 0x000001CFFA5BB340> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF84D540> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF7EF1F0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF827520> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF7ED2D0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF808490> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFFA42BA30> False
<keras.layers.core.activation.Activation object at 0x000001CFCF7CFFD0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF808250> False
<keras.layers.merging.add.Add object at 0x000001CFFA45DAB0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFA0AAA920> False
<keras.layers.core.activation.Activation object at 0x000001CFC30FF100> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFA0C06110> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF89FA60> False
<keras.layers.core.activation.Activation object at 0x000001CFA0C077F0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF8B54B0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8B6CB0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF8B7B80> False
<keras.layers.core.activation.Activation object at 0x000001CFCF8B78E0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8BE080> False
<keras.layers.merging.add.Add object at 0x000001CFCF8BCD90> False
```

```
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF8BD630> False
<keras.layers.core.activation.Activation object at 0x000001CFCF8BF430> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8BE680> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF8D0A00> False
<keras.layers.core.activation.Activation object at 0x000001CFCF8BD7E0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF8D2740> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8D3E80> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF8D3400> False
<keras.layers.core.activation.Activation object at 0x000001CFCF8D33D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8E77C0> False
<keras.layers.merging.add.Add object at 0x000001CFCF8E68C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF8E4340> False
<keras.layers.core.activation.Activation object at 0x000001CFCF8D2680> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8D31C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF8E5F90> False
<keras.layers.core.activation.Activation object at 0x000001CFCF8E5AB0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF8D0F40> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF89F130> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFFA42BBB0> False
<keras.layers.core.activation.Activation object at 0x000001CFFA42B6D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8FF2B0> False
<keras.layers.merging.add.Add object at 0x000001CFCF8FE0E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF8FDAE0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF8FEC20> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8FF610> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF905DE0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF8FFDF0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF907970> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF925660> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF926290> False
<keras.layers.core.activation.Activation object at 0x000001CFCF9067D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF924670> False
<keras.layers.merging.add.Add object at 0x000001CFCF9273D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF938DF0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF939210> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF93B010> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF93BDF0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF93B3D0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF925C30> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF907310> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF8FC430> False
<keras.layers.core.activation.Activation object at 0x000001CFC30FDD50> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8B4580> False
<keras.layers.merging.add.Add object at 0x000001CFCF8BC250> False
```

```
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF950E50> False
<keras.layers.core.activation.Activation object at 0x000001CFCF8D1FC0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF952740> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF9503D0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF9515D0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF9685E0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF96A500> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF96B6D0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF96B130> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF96A800> False
<keras.layers.merging.add.Add object at 0x000001CFCF96BBE0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF97C430> False
<keras.layers.core.activation.Activation object at 0x000001CFCF97E0B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF97FE20> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF97F3A0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF97F7F0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF999AE0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF99BB20> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF99B190> False
<keras.layers.core.activation.Activation object at 0x000001CFCF99BCA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF97CA90> False
<keras.layers.merging.add.Add object at 0x000001CFCF97D750> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF96BA30> False
<keras.layers.core.activation.Activation object at 0x000001CFCF8BD300> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF953220> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF924640> False
<keras.layers.core.activation.Activation object at 0x000001CFC3E7EEF0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF9B19F0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF9B33A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF9B37F0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF9B3F10> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF9CA770> False
<keras.layers.merging.add.Add object at 0x000001CFCF9C9210> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF9C9000> False
<keras.layers.core.activation.Activation object at 0x000001CFCF9CBB20> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF9CAD70> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF9E13C0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF9C9ED0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF9E2E30> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF9F8B20> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF9E35B0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF9E3BB0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF9FBEB0> False
<keras.layers.merging.add.Add object at 0x000001CFCF9FA920> False
```

```
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA10640> False
<keras.layers.core.activation.Activation object at 0x000001CFCF9F90C0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF9FA8F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF9E0A00> False
<keras.layers.core.activation.Activation object at 0x000001CFCF938310> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF9E0A30> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF9990F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF9B0130> False
<keras.layers.core.activation.Activation object at 0x000001CFCF9C92A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF9B14E0> False
<keras.layers.merging.add.Add object at 0x000001CFCF93B790> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA13700> False
<keras.layers.core.activation.Activation object at 0x000001CFCF862770> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA134C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA2C9D0> False
<keras.layers.core.activation.Activation object at 0x000001CFCFA11E70> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCFA2D990> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF9B3E20> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA2DA80> False
<keras.layers.core.activation.Activation object at 0x000001CFCFA13AC0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA41930> False
<keras.layers.merging.add.Add object at 0x000001CFCFA41810> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA40A60> False
<keras.layers.core.activation.Activation object at 0x000001CFCFA2CAF0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA43130> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA43880> False
<keras.layers.core.activation.Activation object at 0x000001CFCFA64F70> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCFA64670> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA667A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA43910> False
<keras.layers.core.activation.Activation object at 0x000001CFCFA41570> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF8FF790> False
<keras.layers.merging.add.Add object at 0x000001CFCF8FE530> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA643D0> False
<keras.layers.core.activation.Activation object at 0x000001CFCFA426B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF97E230> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF953790> False
<keras.layers.core.activation.Activation object at 0x000001CFCFA2E080> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF9E0A90> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA654B0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA851E0> False
<keras.layers.core.activation.Activation object at 0x000001CFCF953C10> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA85810> False
<keras.layers.merging.add.Add object at 0x000001CFCFA878E0> False
```

```
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFAA12A0> False
<keras.layers.core.activation.Activation object at 0x000001CFCFA87EE0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFAA0CA0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFAA2F50> False
<keras.layers.core.activation.Activation object at 0x000001CFCFAA30A0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCFAA2CB0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA130D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFAC5900> False
<keras.layers.core.activation.Activation object at 0x000001CFCFAA0D00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFAC6C50> False
<keras.layers.merging.add.Add object at 0x000001CFCFAC77C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFAC4D90> False
<keras.layers.core.activation.Activation object at 0x000001CFCFAA3820> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFAA33D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF9B0D90> False
<keras.layers.core.activation.Activation object at 0x000001CFCFAC52D0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCF8FDB40> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA87CD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA87B20> False
<keras.layers.core.activation.Activation object at 0x000001CFCFAC4DC0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA85D50> False
<keras.layers.merging.add.Add object at 0x000001CFCFA2FCA0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFAE0160> False
<keras.layers.core.activation.Activation object at 0x000001CFCFAC4A00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFAE0820> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFAE04F0> False
<keras.layers.core.activation.Activation object at 0x000001CFCFAE0430> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCFAFD8D0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCF9B1D80> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFAFDC00> False
<keras.layers.core.activation.Activation object at 0x000001CFCFAE0640> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFB19A20> False
<keras.layers.merging.add.Add object at 0x000001CFCFB19900> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFB18AC0> False
<keras.layers.core.activation.Activation object at 0x000001CFCFAFC5E0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFB19660> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFB19ED0> False
<keras.layers.core.activation.Activation object at 0x000001CFCFA67DF0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCFA66CE0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFB19C60> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA10D60> False
<keras.layers.core.activation.Activation object at 0x000001CFCFA2FF70> False
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x000001CFCFB1B1F0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFAE2290> False
```

```
<keras.layers.merging.add.Add object at 0x000001CFCFAFEB60> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFAC4F70> False
<keras.layers.core.activation.Activation object at 0x000001CFCFAE0B50> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA845E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFA84730> False
<keras.layers.core.activation.Activation object at 0x000001CFCFAC67D0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCFB3F220> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFB1A9E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFB5EC80> False
<keras.layers.core.activation.Activation object at 0x000001CFCFB3FD00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA87700> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFB5ECB0> False
<keras.layers.merging.add.Add object at 0x000001CFCFB5D930> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFB75510> False
<keras.layers.core.activation.Activation object at 0x000001CFCFB3EAD0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFB77790> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFB77580> False
<keras.layers.core.activation.Activation object at 0x000001CFCFB77CA0> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCFB91BA0> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFB91E70> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFB93CD0> False
<keras.layers.core.activation.Activation object at 0x000001CFCFB91930> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFB92920> False
<keras.layers.merging.add.Add object at 0x000001CFCFAA1210> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFB3E5F0> False
<keras.layers.core.activation.Activation object at 0x000001CFCFB77FA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFA87AC0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCF9B3640> False
<keras.layers.core.activation.Activation object at 0x000001CFCFB75630> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001CFCFAFF640> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFB911E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFB5CAF0> False
<keras.layers.core.activation.Activation object at 0x000001CFCFB19CF0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001CFCFBBF0D0> True
<keras.layers.merging.add.Add object at 0x000001CFCFBBE380> True
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFCFBBCA00> True
<keras.layers.core.activation.Activation object at 0x000001CFCFB5CDF0> True
```

We have brought in a trained ResNet model and made the last few layers trainable for fine-tuning.

In [118…

```
def model_transfer_train(model_base):
  backend.clear_session()
  X = models.Sequential()
  X.add(model_base)
  X.add(layers.Flatten())
```

```
  X.add(layers.Dense(512, activation = 'relu'))
  X.add(layers.Dense(512, activation = 'relu'))
  X.add(layers.Dense(512, activation = 'relu'))
  X.add(layers.Dense(11, activation = 'softmax'))

  X.compile('adam',
    loss = 'categorical_crossentropy',
    metrics = ['accuracy'])

  return X
```

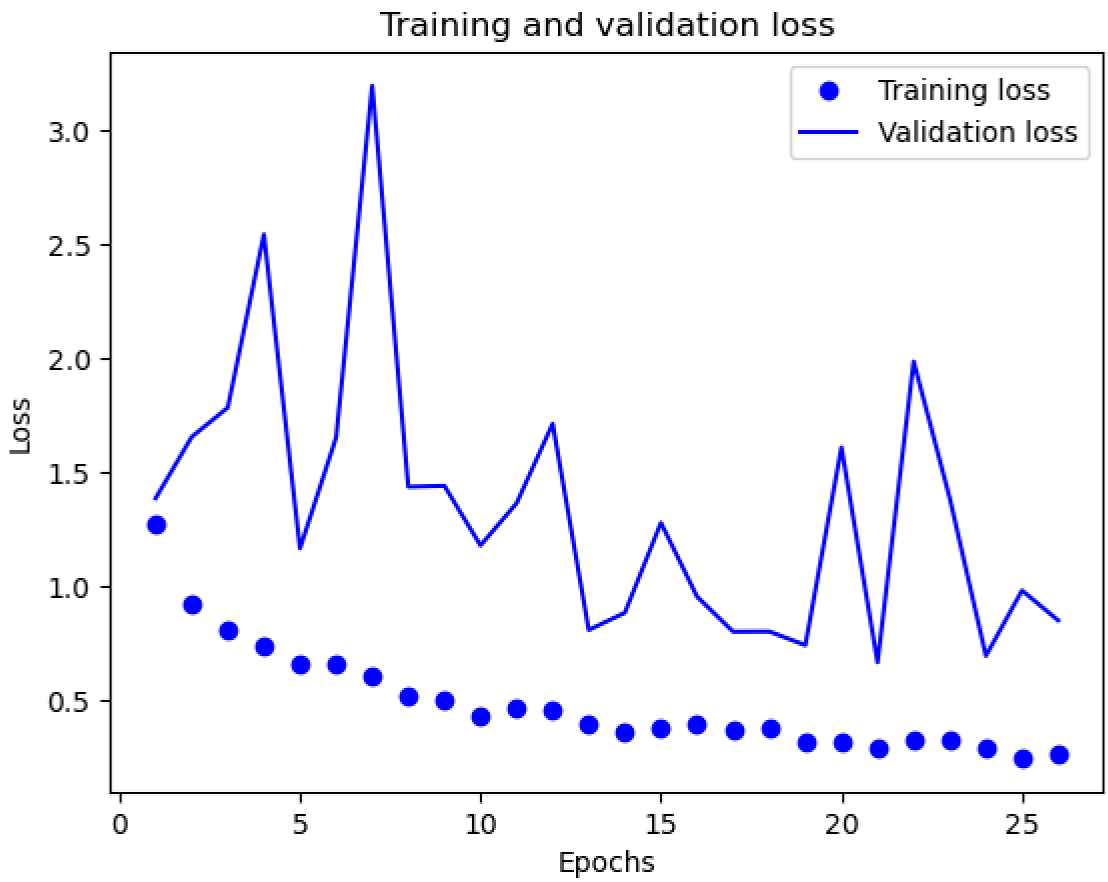
Finishing the model we add our own top to act as the classifier for our 11 leaf health classes.

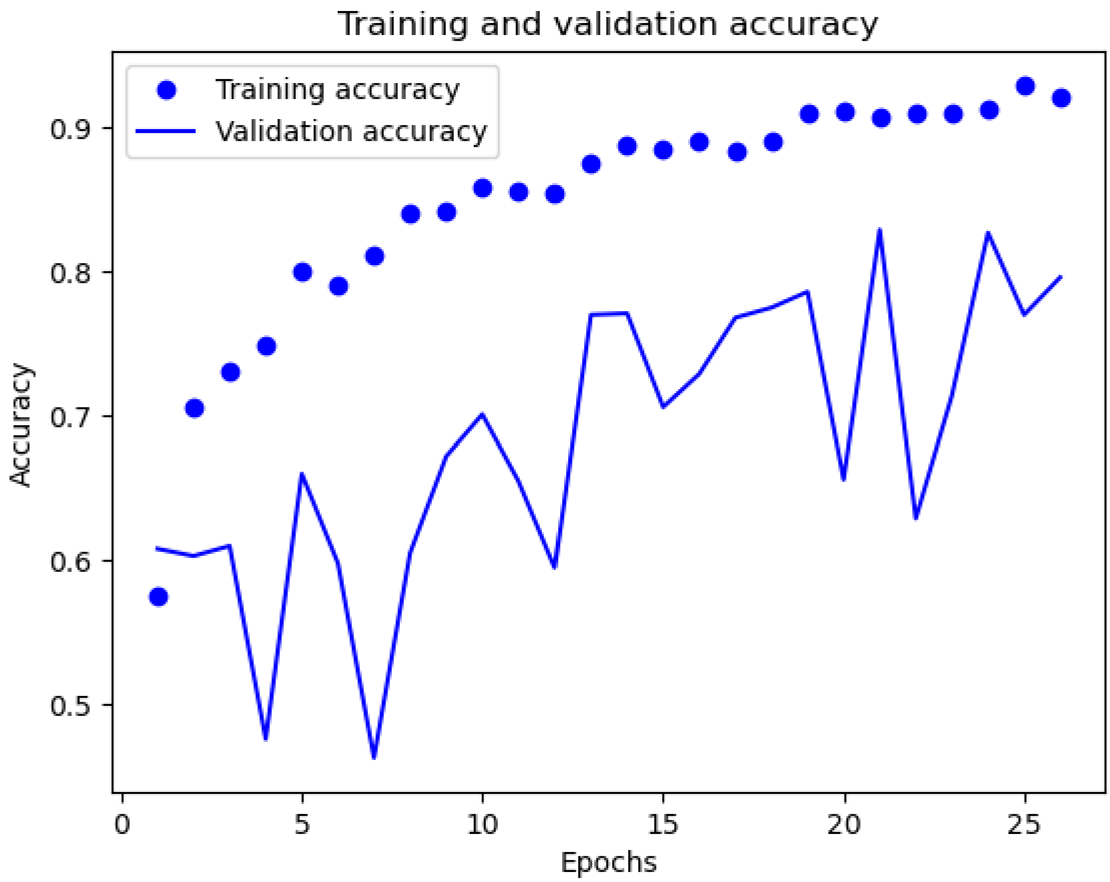
In [76]:
```
model = model_transfer_train(resnet_base)
```

In [78]:
```
history = model.fit(
    train_generator,
    steps_per_epoch=100,
    epochs=100,
    validation_data=val_generator,
    validation_steps=50,
    verbose = 1,
    callbacks=[EarlyStopping(monitor='val_accuracy', patience=5, restore_best_weights


plot_history(history)

test_loss, test_acc = model.evaluate(test_generator, steps = 100)
print('test_acc:', test_acc)
model_path = "tomato_leaf_classifier_transfer_resnet152v2.h5"
model.save(model_path)
```

```
Epoch 1/100
100/100 [==============================] - 190s 2s/step - loss: 1.2708 - accuracy: 0.
5755 - val_loss: 1.3864 - val_accuracy: 0.6080
Epoch 2/100
100/100 [==============================] - 186s 2s/step - loss: 0.9243 - accuracy: 0.
7055 - val_loss: 1.6565 - val_accuracy: 0.6030
Epoch 3/100
100/100 [==============================] - 186s 2s/step - loss: 0.8125 - accuracy: 0.
7315 - val_loss: 1.7855 - val_accuracy: 0.6100
Epoch 4/100
100/100 [==============================] - 184s 2s/step - loss: 0.7418 - accuracy: 0.
7494 - val_loss: 2.5459 - val_accuracy: 0.4760
Epoch 5/100
100/100 [==============================] - 190s 2s/step - loss: 0.6594 - accuracy: 0.
8008 - val_loss: 1.1666 - val_accuracy: 0.6600
Epoch 6/100
100/100 [==============================] - 187s 2s/step - loss: 0.6575 - accuracy: 0.
7900 - val_loss: 1.6536 - val_accuracy: 0.5980
Epoch 7/100
100/100 [==============================] - 188s 2s/step - loss: 0.6052 - accuracy: 0.
8110 - val_loss: 3.1982 - val_accuracy: 0.4630
Epoch 8/100
100/100 [==============================] - 187s 2s/step - loss: 0.5165 - accuracy: 0.
8410 - val_loss: 1.4373 - val_accuracy: 0.6050
Epoch 9/100
100/100 [==============================] - 200s 2s/step - loss: 0.5052 - accuracy: 0.
8415 - val_loss: 1.4416 - val_accuracy: 0.6720
Epoch 10/100
100/100 [==============================] - 199s 2s/step - loss: 0.4331 - accuracy: 0.
8580 - val_loss: 1.1798 - val_accuracy: 0.7010
Epoch 11/100
100/100 [==============================] - 193s 2s/step - loss: 0.4640 - accuracy: 0.
8565 - val_loss: 1.3652 - val_accuracy: 0.6550
Epoch 12/100
100/100 [==============================] - 187s 2s/step - loss: 0.4571 - accuracy: 0.
8550 - val_loss: 1.7144 - val_accuracy: 0.5950
Epoch 13/100
100/100 [==============================] - 188s 2s/step - loss: 0.3972 - accuracy: 0.
8750 - val_loss: 0.8088 - val_accuracy: 0.7700
Epoch 14/100
100/100 [==============================] - 190s 2s/step - loss: 0.3648 - accuracy: 0.
8880 - val_loss: 0.8842 - val_accuracy: 0.7710
Epoch 15/100
100/100 [==============================] - 187s 2s/step - loss: 0.3807 - accuracy: 0.
8845 - val_loss: 1.2795 - val_accuracy: 0.7060
Epoch 16/100
100/100 [==============================] - 187s 2s/step - loss: 0.3978 - accuracy: 0.
8905 - val_loss: 0.9551 - val_accuracy: 0.7290
Epoch 17/100
100/100 [==============================] - 190s 2s/step - loss: 0.3719 - accuracy: 0.
8840 - val_loss: 0.8012 - val_accuracy: 0.7680
Epoch 18/100
100/100 [==============================] - 188s 2s/step - loss: 0.3806 - accuracy: 0.
8905 - val_loss: 0.8020 - val_accuracy: 0.7750
Epoch 19/100
100/100 [==============================] - 217s 2s/step - loss: 0.3206 - accuracy: 0.
9095 - val_loss: 0.7418 - val_accuracy: 0.7860
Epoch 20/100
100/100 [==============================] - 238s 2s/step - loss: 0.3133 - accuracy: 0.
9120 - val_loss: 1.6108 - val_accuracy: 0.6560
```

```
Epoch 21/100
100/100 [==============================] - 240s 2s/step - loss: 0.2914 - accuracy: 0.
9070 - val_loss: 0.6660 - val_accuracy: 0.8290
Epoch 22/100
100/100 [==============================] - 236s 2s/step - loss: 0.3252 - accuracy: 0.
9097 - val_loss: 1.9882 - val_accuracy: 0.6290
Epoch 23/100
100/100 [==============================] - 240s 2s/step - loss: 0.3224 - accuracy: 0.
9105 - val_loss: 1.3868 - val_accuracy: 0.7140
Epoch 24/100
100/100 [==============================] - 246s 2s/step - loss: 0.2930 - accuracy: 0.
9125 - val_loss: 0.6948 - val_accuracy: 0.8270
Epoch 25/100
100/100 [==============================] - 233s 2s/step - loss: 0.2473 - accuracy: 0.
9295 - val_loss: 0.9821 - val_accuracy: 0.7700
Epoch 26/100
100/100 [==============================] - 238s 2s/step - loss: 0.2600 - accuracy: 0.
9205 - val_loss: 0.8507 - val_accuracy: 0.7960
```






```
100/100 [==============================] - 144s 1s/step - loss: 0.6345 - accuracy: 0.
8435
test_acc: 0.843500018119812
```

In [79]:
```
model_size(model_path)
```

```
Model is approx 1054.418624 MB
```

Our first transfer learning model is 84% accurate but also over 1 GB in size. Let's try a few more transfer learning models before we move on to other alternatives.

In [106…
```
backend.clear_session()
effnet_base = EfficientNetB0(weights = 'imagenet', include_top = False, input_shape =

effnet_base.trainable = False # I tried doing fine tuning with this model as well but
```

In [107…
```
def model_efficient_net_train(base):
  backend.clear_session()
  X = models.Sequential()
  X.add(base)
  X.add(layers.Flatten())
  X.add(layers.Dense(512, activation = 'relu'))
  X.add(layers.Dense(512, activation = 'relu'))
  X.add(layers.Dense(512, activation = 'relu'))
  X.add(layers.Dense(11, activation = 'softmax'))

  X.compile('adam',
    loss = 'categorical_crossentropy',
    metrics = ['accuracy'])

  return X
```

In [108…

```
model = model_efficient_net_train(effnet_base)
```

In [109…

```
history = model.fit(
    train_generator,
    steps_per_epoch=100,
    epochs=100,
    validation_data=val_generator,
    validation_steps=50,
    verbose = 1,
    callbacks=[EarlyStopping(monitor='val_accuracy', patience=4, restore_best_weights


plot_history(history)

test_loss, test_acc = model.evaluate(test_generator, steps = 100)
print('test_acc:', test_acc)
model_path = "tomato_leaf_classifier_transfer_effnetB0.h5"
model.save(model_path)
```

```
Epoch 1/100
100/100 [==============================] - 292s 3s/step - loss: 15.2926 - accuracy: 0.1120 - val_loss: 2.3949 - val_accuracy: 0.1020
Epoch 2/100
100/100 [==============================] - 265s 3s/step - loss: 2.5077 - accuracy: 0.1070 - val_loss: 2.3673 - val_accuracy: 0.1000
Epoch 3/100
100/100 [==============================] - 260s 3s/step - loss: 2.4231 - accuracy: 0.1180 - val_loss: 2.3676 - val_accuracy: 0.1060
Epoch 4/100
 86/100 [========================>.....] - ETA: 26s - loss: 2.4793 - accuracy: 0.1122
```

```
---------------------------------------------------------------------------
KeyboardInterrupt                         Traceback (most recent call last)
Cell In [109], line 1
----> 1 history = model.fit(
      2     train_generator3,
      3     steps_per_epoch=100,
      4     epochs=100,
      5     validation_data=validation_generator3,
      6     validation_steps=50,
      7     verbose = 1,
      8     callbacks=[EarlyStopping(monitor='val_accuracy', patience=4, restore_best
_weights = True)])
     11 plot_history(history)
     13 test_loss, test_acc = model.evaluate(test_generator3, steps = 100)

File ~\anaconda3\envs\deep-learning-cv\lib\site-packages\keras\utils\traceback_utils.
py:64, in filter_traceback.<locals>.error_handler(*args, **kwargs)
     62 filtered_tb = None
     63 try:
---> 64   return fn(*args, **kwargs)
     65 except Exception as e:  # pylint: disable=broad-except
     66   filtered_tb = _process_traceback_frames(e.__traceback__)

File ~\anaconda3\envs\deep-learning-cv\lib\site-packages\keras\engine\training.py:140
9, in Model.fit(self, x, y, batch_size, epochs, verbose, callbacks, validation_split,
validation_data, shuffle, class_weight, sample_weight, initial_epoch, steps_per_epoc
h, validation_steps, validation_batch_size, validation_freq, max_queue_size, workers,
use_multiprocessing)
   1402 with tf.profiler.experimental.Trace(
   1403     'train',
   1404     epoch_num=epoch,
   1405     step_num=step,
   1406     batch_size=batch_size,
   1407     _r=1):
   1408   callbacks.on_train_batch_begin(step)
-> 1409   tmp_logs = self.train_function(iterator)
   1410   if data_handler.should_sync:
   1411     context.async_wait()

File ~\anaconda3\envs\deep-learning-cv\lib\site-packages\tensorflow\python\util\trace
back_utils.py:150, in filter_traceback.<locals>.error_handler(*args, **kwargs)
    148 filtered_tb = None
    149 try:
--> 150   return fn(*args, **kwargs)
    151 except Exception as e:
    152   filtered_tb = _process_traceback_frames(e.__traceback__)

File ~\anaconda3\envs\deep-learning-cv\lib\site-packages\tensorflow\python\eager\def_
function.py:915, in Function.__call__(self, *args, **kwds)
    912 compiler = "xla" if self._jit_compile else "nonXla"
    914 with OptionalXlaContext(self._jit_compile):
--> 915   result = self._call(*args, **kwds)
    917 new_tracing_count = self.experimental_get_tracing_count()
    918 without_tracing = (tracing_count == new_tracing_count)

File ~\anaconda3\envs\deep-learning-cv\lib\site-packages\tensorflow\python\eager\def_
function.py:947, in Function._call(self, *args, **kwds)
    944   self._lock.release()
    945   # In this case we have created variables on the first call, so we run the
    946   # defunned version which is guaranteed to never create variables.
```

```
--> 947   return self._stateless_fn(*args, **kwds)  # pylint: disable=not-callable
    948 elif self._stateful_fn is not None:
    949   # Release the lock early so that multiple threads can perform the call
    950   # in parallel.
    951   self._lock.release()

File ~\anaconda3\envs\deep-learning-cv\lib\site-packages\tensorflow\python\eager\function.py:2453, in Function.__call__(self, *args, **kwargs)
   2450 with self._lock:
   2451   (graph_function,
   2452    filtered_flat_args) = self._maybe_define_function(args, kwargs)
-> 2453 return graph_function._call_flat(
   2454     filtered_flat_args, captured_inputs=graph_function.captured_inputs)

File ~\anaconda3\envs\deep-learning-cv\lib\site-packages\tensorflow\python\eager\function.py:1860, in ConcreteFunction._call_flat(self, args, captured_inputs, cancellation_manager)
   1856 possible_gradient_type = gradients_util.PossibleTapeGradientTypes(args)
   1857 if (possible_gradient_type == gradients_util.POSSIBLE_GRADIENT_TYPES_NONE
   1858     and executing_eagerly):
   1859   # No tape is watching; skip to running the function.
-> 1860   return self._build_call_outputs(self._inference_function.call(
   1861       ctx, args, cancellation_manager=cancellation_manager))
   1862 forward_backward = self._select_forward_and_backward_functions(
   1863     args,
   1864     possible_gradient_type,
   1865     executing_eagerly)
   1866 forward_function, args_with_tangents = forward_backward.forward()

File ~\anaconda3\envs\deep-learning-cv\lib\site-packages\tensorflow\python\eager\function.py:497, in _EagerDefinedFunction.call(self, ctx, args, cancellation_manager)
    495 with _InterpolateFunctionError(self):
    496   if cancellation_manager is None:
--> 497     outputs = execute.execute(
    498         str(self.signature.name),
    499         num_outputs=self._num_outputs,
    500         inputs=args,
    501         attrs=attrs,
    502         ctx=ctx)
    503   else:
    504     outputs = execute.execute_with_cancellation(
    505         str(self.signature.name),
    506         num_outputs=self._num_outputs,
   (...)
    509         ctx=ctx,
    510         cancellation_manager=cancellation_manager)

File ~\anaconda3\envs\deep-learning-cv\lib\site-packages\tensorflow\python\eager\execute.py:54, in quick_execute(op_name, num_outputs, inputs, attrs, ctx, name)
     52 try:
     53   ctx.ensure_initialized()
---> 54   tensors = pywrap_tfe.TFE_Py_Execute(ctx._handle, device_name, op_name,
     55                                       inputs, attrs, num_outputs)
     56 except core._NotOkStatusException as e:
     57   if name is not None:

KeyboardInterrupt:
```

In [99]:

```
model_size(model_path)
```

```
Model is approx 529.615288 MB
```

Unfortunately we didn't get a good result from this one with an accuracy never making it out of the 40%'s, and a model size of 530 MB. This is a little concerning because my plan was to train a smaller model using a good transfer learning model as a teacher, and with this being half the size of our previous transfer learner but unable to compete on accuracy I am worried.

In [121…

```
# Now we can freeze all the resnet weights except the last few, and train those before
backend.clear_session()
base = DenseNet201(weights = 'imagenet', include_top = False, input_shape = (256, 256,

# Here we freeze all the layers except the last 4.
for layer in base.layers[:-4]:
    layer.trainable = False
for layer in base.layers:
    print(layer, layer.trainable)
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/de
nsenet/densenet201_weights_tf_dim_ordering_tf_kernels_notop.h5
74836368/74836368 [==============================] - 2s 0us/step
<keras.engine.input_layer.InputLayer object at 0x000001D169596A40> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001D169A18490> Fa
lse
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A181F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169A19450> False
<keras.layers.core.activation.Activation object at 0x000001D169A18E50> False
<keras.layers.reshaping.zero_padding2d.ZeroPadding2D object at 0x000001D1694FFEE0> Fa
lse
<keras.layers.pooling.max_pooling2d.MaxPooling2D object at 0x000001CFA0123460> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16954F2B0> False
<keras.layers.core.activation.Activation object at 0x000001D16954EF80> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A1A560> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169A1AF20> False
<keras.layers.core.activation.Activation object at 0x000001D16A4BBBB0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A4D8A0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169A1B100> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169A4F7C0> False
<keras.layers.core.activation.Activation object at 0x000001CFF1838040> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A4F280> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169A5F5E0> False
<keras.layers.core.activation.Activation object at 0x000001D169A5DB40> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A5C250> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169A5DF30> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169A5E230> False
<keras.layers.core.activation.Activation object at 0x000001D169A1A980> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A1B1F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16C4251B0> False
<keras.layers.core.activation.Activation object at 0x000001D16A4B82B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A196F0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169A5D1B0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169A7C370> False
<keras.layers.core.activation.Activation object at 0x000001D169A4EA10> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A4D9CC0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A4DA6B0> False
<keras.layers.core.activation.Activation object at 0x000001D169A5D630> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A7C550> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169A7DB10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169A8A8C0> False
<keras.layers.core.activation.Activation object at 0x000001D169A7DC30> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A89090> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169A89C00> False
<keras.layers.core.activation.Activation object at 0x000001D169A7F2B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A88BE0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169A8AE00> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169AA3430> False
```

```
<keras.layers.core.activation.Activation object at 0x000001D169A8A8F0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169AA1C00> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169AA3A30> False
<keras.layers.core.activation.Activation object at 0x000001D169A8B670> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A8B730> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169AA3970> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169A18340> False
<keras.layers.core.activation.Activation object at 0x000001D169AA3460> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A88FA0> False
<keras.layers.pooling.average_pooling2d.AveragePooling2D object at 0x000001D169A19180
> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
CFF17BB760> False
<keras.layers.core.activation.Activation object at 0x000001D169AA3F10> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169AC1930> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169A7E2C0> False
<keras.layers.core.activation.Activation object at 0x000001D169A88DF0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169AC0E20> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169AC2FB0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169ADAA10> False
<keras.layers.core.activation.Activation object at 0x000001D169AC3220> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169AD92D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169AD9660> False
<keras.layers.core.activation.Activation object at 0x000001D169AC0880> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169AF3BE0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169ADA5F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169AF3EE0> False
<keras.layers.core.activation.Activation object at 0x000001D169ADA590> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169AF2B60> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169AF0850> False
<keras.layers.core.activation.Activation object at 0x000001D169B150C0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169B172E0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169B15240> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169AF2620> False
<keras.layers.core.activation.Activation object at 0x000001D169B149A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169B16BC0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169B176D0> False
<keras.layers.core.activation.Activation object at 0x000001D169AF31F0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169A7D510> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169B170A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169A7E5C0> False
<keras.layers.core.activation.Activation object at 0x000001D169ADB6A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169AD9FC0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169AA1210> False
<keras.layers.core.activation.Activation object at 0x000001D169ADA440> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169B330D0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169597160> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169B332E0> False
```

```
<keras.layers.core.activation.Activation object at 0x000001D169AD8520> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169B335B0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169B32230> False
<keras.layers.core.activation.Activation object at 0x000001D169A89F90> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169B49840> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169B328C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169B4BAF0> False
<keras.layers.core.activation.Activation object at 0x000001D169B4A9E0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169B65CF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169B66410> False
<keras.layers.core.activation.Activation object at 0x000001D169B669E0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169B67010> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169B64280> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169B4A950> False
<keras.layers.core.activation.Activation object at 0x000001D169B65210> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169AC3190> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169B65DE0> False
<keras.layers.core.activation.Activation object at 0x000001D169B4BA30> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169B31DE0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169B33040> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169AD9F60> False
<keras.layers.core.activation.Activation object at 0x000001D169AD8FA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169AF3760> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169B17250> False
<keras.layers.core.activation.Activation object at 0x000001D169AC12D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169B87AF0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169AF1B10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169BA29B0> False
<keras.layers.core.activation.Activation object at 0x000001D169AD9E10> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169BA1180> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169BA1CF0> False
<keras.layers.core.activation.Activation object at 0x000001D169B87940> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169BA15A0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169BA2EF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169BC3520> False
<keras.layers.core.activation.Activation object at 0x000001D169BA29E0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169BC1CF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169BC2860> False
<keras.layers.core.activation.Activation object at 0x000001D169BD8C10> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169BD9B40> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169BDABF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169BC1240> False
<keras.layers.core.activation.Activation object at 0x000001D169BD9BA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169BD80D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169BDB3D0> False
<keras.layers.core.activation.Activation object at 0x000001D169BC2C50> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169B87220> False
```

```
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169BD9930> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169B845E0> False
<keras.layers.core.activation.Activation object at 0x000001D169BD8AF0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169B87280> False
<keras.layers.pooling.average_pooling2d.AveragePooling2D object at 0x000001D169BA3610
> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169AC2200> False
<keras.layers.core.activation.Activation object at 0x000001D169B150F0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169BF72B0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169BF5900> False
<keras.layers.core.activation.Activation object at 0x000001D169B4B3D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169BF5690> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169BF6BC0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C13700> False
<keras.layers.core.activation.Activation object at 0x000001D169B31000> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C11ED0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C12200> False
<keras.layers.core.activation.Activation object at 0x000001D169C11060> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C318A0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169C132E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C30EB0> False
<keras.layers.core.activation.Activation object at 0x000001D169C30B20> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C13130> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C301F0> False
<keras.layers.core.activation.Activation object at 0x000001D169AC3730> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C11CF0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169C12650> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169BA0F10> False
<keras.layers.core.activation.Activation object at 0x000001D169B84880> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C30FA0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169BF4A60> False
<keras.layers.core.activation.Activation object at 0x000001D169C11780> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C32740> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169BF4700> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C48790> False
<keras.layers.core.activation.Activation object at 0x000001D169BF5210> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C48D60> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C4BC40> False
<keras.layers.core.activation.Activation object at 0x000001D169C33FD0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C699C0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169C4BC10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C6BC40> False
<keras.layers.core.activation.Activation object at 0x000001D169C6AB60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C7E4A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C7CF70> False
<keras.layers.core.activation.Activation object at 0x000001D169C6AAA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C7EE60> False
```

```
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169C7C220> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169CA7A00> False
<keras.layers.core.activation.Activation object at 0x000001D169CA5C60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C7FBE0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169CA50F0> False
<keras.layers.core.activation.Activation object at 0x000001D169CA76D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169CA4220> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169CA4CD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169CA4730> False
<keras.layers.core.activation.Activation object at 0x000001D169CA5630> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C7E6E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C4AE90> False
<keras.layers.core.activation.Activation object at 0x000001D169BA3F40> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C7FE80> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169C6B190> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169CBE620> False
<keras.layers.core.activation.Activation object at 0x000001D169C7E650> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169CBD600> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169CBD960> False
<keras.layers.core.activation.Activation object at 0x000001D169C4BB20> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169CBD360> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169CBEB60> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169CD3190> False
<keras.layers.core.activation.Activation object at 0x000001D169CBE650> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169CD1960> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169CD36D0> False
<keras.layers.core.activation.Activation object at 0x000001D169CBF160> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169CFBE80> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A4D8820> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169CF96F0> False
<keras.layers.core.activation.Activation object at 0x000001D169CF9A20> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169CD3DF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169CD2200> False
<keras.layers.core.activation.Activation object at 0x000001D169C4B460> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169BC0850> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169CFA3B0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C4ABC0> False
<keras.layers.core.activation.Activation object at 0x000001D169CFBB50> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C4BFD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169CD1CC0> False
<keras.layers.core.activation.Activation object at 0x000001D169C7E230> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169CD09A0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169CF96C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169D13B80> False
<keras.layers.core.activation.Activation object at 0x000001D169CBC070> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D13400> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
```

```
D169D13670> False
<keras.layers.core.activation.Activation object at 0x000001D169CD2950> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D31330> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169D10A60> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169D320B0> False
<keras.layers.core.activation.Activation object at 0x000001D169D11DB0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D33EE0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169D33C40> False
<keras.layers.core.activation.Activation object at 0x000001D169D32410> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D533D0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169D50A30> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169D6F370> False
<keras.layers.core.activation.Activation object at 0x000001D169D6D600> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D6F6D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169D6F8E0> False
<keras.layers.core.activation.Activation object at 0x000001D169D52140> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D503D0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169D51390> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169CA47C0> False
<keras.layers.core.activation.Activation object at 0x000001D169D6EB60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D30130> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C11C60> False
<keras.layers.core.activation.Activation object at 0x000001D169D6E230> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D11FF0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169D6EC50> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169C7FA00> False
<keras.layers.core.activation.Activation object at 0x000001D169CF95D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D7CF70> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169D7DC00> False
<keras.layers.core.activation.Activation object at 0x000001D169C12B90> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D7D7B0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169D7CF10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169DAAB00> False
<keras.layers.core.activation.Activation object at 0x000001D169D7DF90> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169DA92D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169DA9E40> False
<keras.layers.core.activation.Activation object at 0x000001D169D7FA90> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169DA9FF0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169DAB040> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169DC29B0> False
<keras.layers.core.activation.Activation object at 0x000001D169DC12D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169DABC10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169DC3D30> False
<keras.layers.core.activation.Activation object at 0x000001D169CBDAB0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169C7D540> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169C68EB0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169DC2DD0> False
```

```
<keras.layers.core.activation.Activation object at 0x000001D169DC36A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169DC13C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169DA9030> False
<keras.layers.core.activation.Activation object at 0x000001D169DA9420> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D504C0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169D13910> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169DE3BB0> False
<keras.layers.core.activation.Activation object at 0x000001D169D7E230> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169DE2440> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169DE2FB0> False
<keras.layers.core.activation.Activation object at 0x000001D169D13A30> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E00BB0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169DE3FD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E01960> False
<keras.layers.core.activation.Activation object at 0x000001D169DE3CA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E02B90> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E01FC0> False
<keras.layers.core.activation.Activation object at 0x000001D169E01ED0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E158A0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169E036D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E17B50> False
<keras.layers.core.activation.Activation object at 0x000001D169E16A40> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E02F50> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E022C0> False
<keras.layers.core.activation.Activation object at 0x000001D169E024A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E15D80> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169E009D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169D7E5F0> False
<keras.layers.core.activation.Activation object at 0x000001D169E02DA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169DE2C50> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169D7F400> False
<keras.layers.core.activation.Activation object at 0x000001D169E03BB0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169D51960> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169D7E800> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E38A60> False
<keras.layers.core.activation.Activation object at 0x000001D169E385E0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E5C7C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E5E980> False
<keras.layers.core.activation.Activation object at 0x000001D169E3AFE0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E5FC70> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169E5C970> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E76500> False
<keras.layers.core.activation.Activation object at 0x000001D169E5D990> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E754E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E75840> False
<keras.layers.core.activation.Activation object at 0x000001D169E5F7C0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E76D10> False
```

```
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169E76A40> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E923B0> False
<keras.layers.core.activation.Activation object at 0x000001D169E90F70> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E77610> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E751B0> False
<keras.layers.core.activation.Activation object at 0x000001D169DC0490> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169DC1150> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169E5F070> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E15030> False
<keras.layers.core.activation.Activation object at 0x000001D169E75EA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169DAA2F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E930A0> False
<keras.layers.core.activation.Activation object at 0x000001D169DE0490> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E5D690> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169E5F7F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169EB7670> False
<keras.layers.core.activation.Activation object at 0x000001D169E5DFC0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169EB5E40> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169EB69B0> False
<keras.layers.core.activation.Activation object at 0x000001D169DE0460> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169ECBA30> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169EB7BB0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169ECADD0> False
<keras.layers.core.activation.Activation object at 0x000001D169EB76A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169ECAAD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169EC9C30> False
<keras.layers.core.activation.Activation object at 0x000001D169EB6260> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169EE92A0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169ECA320> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169EEB790> False
<keras.layers.core.activation.Activation object at 0x000001D169EEA440> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169EC9090> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169DE2590> False
<keras.layers.core.activation.Activation object at 0x000001D169DE0700> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E3BC10> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169EE9A80> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169EEB9A0> False
<keras.layers.core.activation.Activation object at 0x000001D169ECBBE0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169E01B40> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169E91960> False
<keras.layers.core.activation.Activation object at 0x000001D169ECBA00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169F13FA0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169E75540> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169F11BA0> False
<keras.layers.core.activation.Activation object at 0x000001D169F11840> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169F02380> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
```

```
D169F007C0> False
<keras.layers.core.activation.Activation object at 0x000001D169EB5F90> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169F02D40> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169F007F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169F00C10> False
<keras.layers.core.activation.Activation object at 0x000001D169F01C90> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169F48EE0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169F49B70> False
<keras.layers.core.activation.Activation object at 0x000001D169F02E60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169F49720> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169F48E80> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169F65DB0> False
<keras.layers.core.activation.Activation object at 0x000001D169F648B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169F4AA70> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169F48D90> False
<keras.layers.core.activation.Activation object at 0x000001D169F006A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169F03790> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169F020B0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169F65930> False
<keras.layers.core.activation.Activation object at 0x000001D169F4B310> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169F02620> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169EB6D10> False
<keras.layers.core.activation.Activation object at 0x000001D169F4A560> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169F002E0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169F66A40> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169F87070> False
<keras.layers.core.activation.Activation object at 0x000001D169F48160> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169F85840> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169F863B0> False
<keras.layers.core.activation.Activation object at 0x000001D169E93F70> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FA7DF0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169F875B0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169FA7FA0> False
<keras.layers.core.activation.Activation object at 0x000001D169F870A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FA62F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169FA5660> False
<keras.layers.core.activation.Activation object at 0x000001D169F85C60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FC0B20> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169FA7B20> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169FC1F30> False
<keras.layers.core.activation.Activation object at 0x000001D169FC14B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FA6830> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169FA6200> False
<keras.layers.core.activation.Activation object at 0x000001D169E5F400> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169EB5390> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169FC2D40> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169EB5540> False
```

```
<keras.layers.core.activation.Activation object at 0x000001D169FA65F0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FA5ED0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169F65660> False
<keras.layers.core.activation.Activation object at 0x000001D169FA73D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FE7FD0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169F86F20> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169FE4CD0> False
<keras.layers.core.activation.Activation object at 0x000001D169F12260> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FE7400> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169FE7430> False
<keras.layers.core.activation.Activation object at 0x000001D169F87850> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FFF340> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169FE62C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169FFFB50> False
<keras.layers.core.activation.Activation object at 0x000001D169FFED70> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A014730> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A0168F0> False
<keras.layers.core.activation.Activation object at 0x000001D169FFEF50> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A0172B0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A014040> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A03B850> False
<keras.layers.core.activation.Activation object at 0x000001D16A0380D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A016B90> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A0174F0> False
<keras.layers.core.activation.Activation object at 0x000001D169FFCA90> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FFF7C0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D169FFDB10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169DC07C0> False
<keras.layers.core.activation.Activation object at 0x000001D16A0141C0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FFE740> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A017880> False
<keras.layers.core.activation.Activation object at 0x000001D169E5EF50> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FE6500> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A03A9E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A056A70> False
<keras.layers.core.activation.Activation object at 0x000001D16A03B430> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A055240> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A055DB0> False
<keras.layers.core.activation.Activation object at 0x000001D169FFD5A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A057B50> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A056FB0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A06F5E0> False
<keras.layers.core.activation.Activation object at 0x000001D16A056AA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A06DDB0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A06E920> False
<keras.layers.core.activation.Activation object at 0x000001D16A0574F0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A07FF40> False
```

```
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A06FB20> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A07DC00> False
<keras.layers.core.activation.Activation object at 0x000001D16A07DC60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169FC0EE0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169FC1690> False
<keras.layers.core.activation.Activation object at 0x000001D169FC3A00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D169F67520> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A07EE90> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A06CDC0> False
<keras.layers.core.activation.Activation object at 0x000001D16A07E3E0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A06D900> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169FFF070> False
<keras.layers.core.activation.Activation object at 0x000001D169FC0A60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A0B3580> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A0555D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A0B0C10> False
<keras.layers.core.activation.Activation object at 0x000001D16A057C40> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A0B38E0> False
<keras.layers.pooling.average_pooling2d.AveragePooling2D object at 0x000001D16A0B1AE0
> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D169BF7A60> False
<keras.layers.core.activation.Activation object at 0x000001D16A055090> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A0D3F40> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A0D29B0> False
<keras.layers.core.activation.Activation object at 0x000001D16A0B3070> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A0D1300> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A0D3FD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A0F3E20> False
<keras.layers.core.activation.Activation object at 0x000001D16A0B3340> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A10D780> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A10ECB0> False
<keras.layers.core.activation.Activation object at 0x000001D16A10D960> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A10F5B0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A10FA00> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A0D0640> False
<keras.layers.core.activation.Activation object at 0x000001D16A10E260> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A0F0190> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A0F2BC0> False
<keras.layers.core.activation.Activation object at 0x000001D169FE4820> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A0F1210> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A10F5E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A10E890> False
<keras.layers.core.activation.Activation object at 0x000001D16A0B3D00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A0D2CE0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A07CFA0> False
<keras.layers.core.activation.Activation object at 0x000001D16A0F0130> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A131570> False
```

```
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A0561D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1322F0> False
<keras.layers.core.activation.Activation object at 0x000001D16A0B34C0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A1419F0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A132680> False
<keras.layers.core.activation.Activation object at 0x000001D16A132650> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A143670> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A140CD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1413C0> False
<keras.layers.core.activation.Activation object at 0x000001D16A143160> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A160FD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A163AC0> False
<keras.layers.core.activation.Activation object at 0x000001D16A06ECB0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A07D2A0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A07F010> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A0B3220> False
<keras.layers.core.activation.Activation object at 0x000001D16A0B3130> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A03B6A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A143DF0> False
<keras.layers.core.activation.Activation object at 0x000001D16A163F70> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A10D840> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A142DD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A0F2740> False
<keras.layers.core.activation.Activation object at 0x000001D16A0F2170> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A189B70> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A189990> False
<keras.layers.core.activation.Activation object at 0x000001D16A130DF0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A189360> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A188D00> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A19AAD0> False
<keras.layers.core.activation.Activation object at 0x000001D16A189F60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A1992A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A199E10> False
<keras.layers.core.activation.Activation object at 0x000001D16A18BA60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A19A6E0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A19B010> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1BF640> False
<keras.layers.core.activation.Activation object at 0x000001D16A19AB00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A1BDE10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1BF1F0> False
<keras.layers.core.activation.Activation object at 0x000001D16A19BDF0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A1997B0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A19AB60> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1BEDA0> False
<keras.layers.core.activation.Activation object at 0x000001D16A1BF670> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A189CF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
```

```
D16A18B9D0> False
<keras.layers.core.activation.Activation object at 0x000001D16A1BD390> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A133520> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A10F850> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1DF9D0> False
<keras.layers.core.activation.Activation object at 0x000001D16A1BEAD0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A1DED70> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1DD4E0> False
<keras.layers.core.activation.Activation object at 0x000001D16A132380> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A1F5720> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A1DFF10> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1F6DD0> False
<keras.layers.core.activation.Activation object at 0x000001D16A1DFA00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A1F6BC0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1F57B0> False
<keras.layers.core.activation.Activation object at 0x000001D16A1F4B80> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A211600> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A1F5900> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A212380> False
<keras.layers.core.activation.Activation object at 0x000001D16A1F7550> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A22C190> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A22F790> False
<keras.layers.core.activation.Activation object at 0x000001D16A211F00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A211420> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A211390> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A016440> False
<keras.layers.core.activation.Activation object at 0x000001D16A22F1F0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A19A230> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A213820> False
<keras.layers.core.activation.Activation object at 0x000001D16A22D9F0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A19B970> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A213C70> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A18B190> False
<keras.layers.core.activation.Activation object at 0x000001D16A1F4A60> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A256470> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A254310> False
<keras.layers.core.activation.Activation object at 0x000001D16A19A3B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A256E30> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A2550C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A255A80> False
<keras.layers.core.activation.Activation object at 0x000001D16A255D80> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A268FD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A269C60> False
<keras.layers.core.activation.Activation object at 0x000001D16A256F50> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A268C40> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A268F70> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A28AB60> False
```

```
<keras.layers.core.activation.Activation object at 0x000001D16A269FF0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A289330> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A28A710> False
<keras.layers.core.activation.Activation object at 0x000001D16A26A650> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A2683A0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A269EA0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1BCE50> False
<keras.layers.core.activation.Activation object at 0x000001D16A0F09A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A255060> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1F5390> False
<keras.layers.core.activation.Activation object at 0x000001D16A28AE00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A255D50> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A26B970> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A1F6020> False
<keras.layers.core.activation.Activation object at 0x000001D16A26A9B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A2AA620> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A2AA920> False
<keras.layers.core.activation.Activation object at 0x000001D16A255210> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A2C7C70> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A2AA3E0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A2C7F70> False
<keras.layers.core.activation.Activation object at 0x000001D16A2AAEF0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A2C6BF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A2C42B0> False
<keras.layers.core.activation.Activation object at 0x000001D16A2AB910> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A2E49A0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A2C79A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A2E6B90> False
<keras.layers.core.activation.Activation object at 0x000001D16A2C7940> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A2F9AE0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A2FB010> False
<keras.layers.core.activation.Activation object at 0x000001D16A2E7CA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A2E5C30> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A2E4BB0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A2F8160> False
<keras.layers.core.activation.Activation object at 0x000001D16A2E43D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A2FA830> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A2F9150> False
<keras.layers.core.activation.Activation object at 0x000001D16A2FA230> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A2545B0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A2E7A60> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A2AAAD0> False
<keras.layers.core.activation.Activation object at 0x000001D16A2A8820> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A289FF0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A289A20> False
<keras.layers.core.activation.Activation object at 0x000001D16A1DD030> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A325A20> False
```

```
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A28A200> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A327CA0> False
<keras.layers.core.activation.Activation object at 0x000001D16A326BC0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A33A500> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A338FD0> False
<keras.layers.core.activation.Activation object at 0x000001D16A326B00> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A33AEC0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A338760> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A3391E0> False
<keras.layers.core.activation.Activation object at 0x000001D16A339450> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A35D060> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A35EE60> False
<keras.layers.core.activation.Activation object at 0x000001D16A338E50> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A33B9A0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A1F58D0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A2AAF80> False
<keras.layers.core.activation.Activation object at 0x000001D16A2AA560> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A2570A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A254D90> False
<keras.layers.core.activation.Activation object at 0x000001D16A35F6A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A2F93C0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A339D80> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A2C54B0> False
<keras.layers.core.activation.Activation object at 0x000001D16A2F83A0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A37CFD0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A37DE40> False
<keras.layers.core.activation.Activation object at 0x000001D16A33BCA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A37F3A0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A37E020> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A39AF80> False
<keras.layers.core.activation.Activation object at 0x000001D16A37E410> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A399750> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A39A2C0> False
<keras.layers.core.activation.Activation object at 0x000001D16A37F910> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A3B7D00> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A39B4C0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A3B7EE0> False
<keras.layers.core.activation.Activation object at 0x000001D16A39AFB0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A3B6200> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A3B5D80> False
<keras.layers.core.activation.Activation object at 0x000001D16A39BD90> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A39A530> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A39A650> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A3B62F0> False
<keras.layers.core.activation.Activation object at 0x000001D16A3B79D0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A39BB20> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
```

```
D16A338C70> False
<keras.layers.core.activation.Activation object at 0x000001D16A324FA0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A37E2F0> False
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A3B58A0> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A3D32E0> False
<keras.layers.core.activation.Activation object at 0x000001D16A37F2B0> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16A3D3130> False
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16A3D2020> False
<keras.layers.core.activation.Activation object at 0x000001D16A37D030> False
<keras.layers.convolutional.conv2d.Conv2D object at 0x000001D16B2A1390> True
<keras.layers.merging.concatenate.Concatenate object at 0x000001D16A3D3E80> True
<keras.layers.normalization.batch_normalization.BatchNormalization object at 0x000001
D16B2A3250> True
<keras.layers.core.activation.Activation object at 0x000001D16A3D2CE0> True
```

In [122…

```
model = model_transfer_train(base)
```

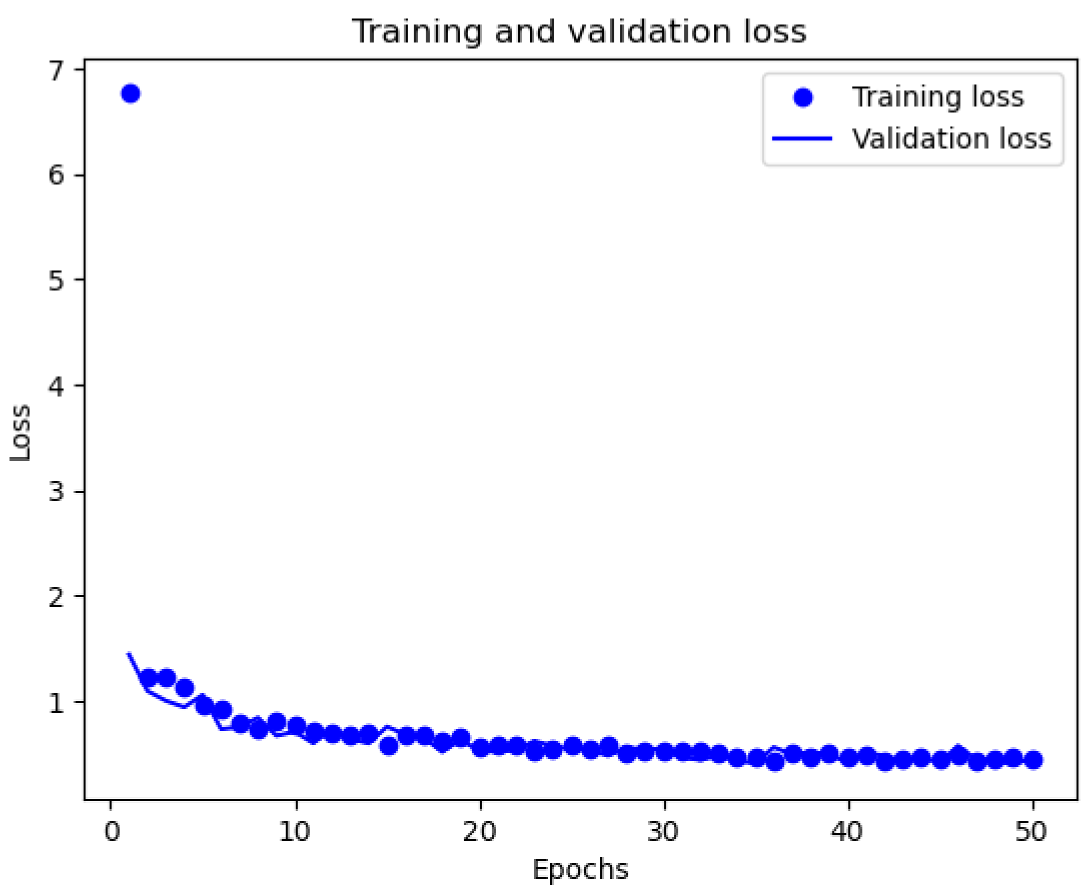
In [123…

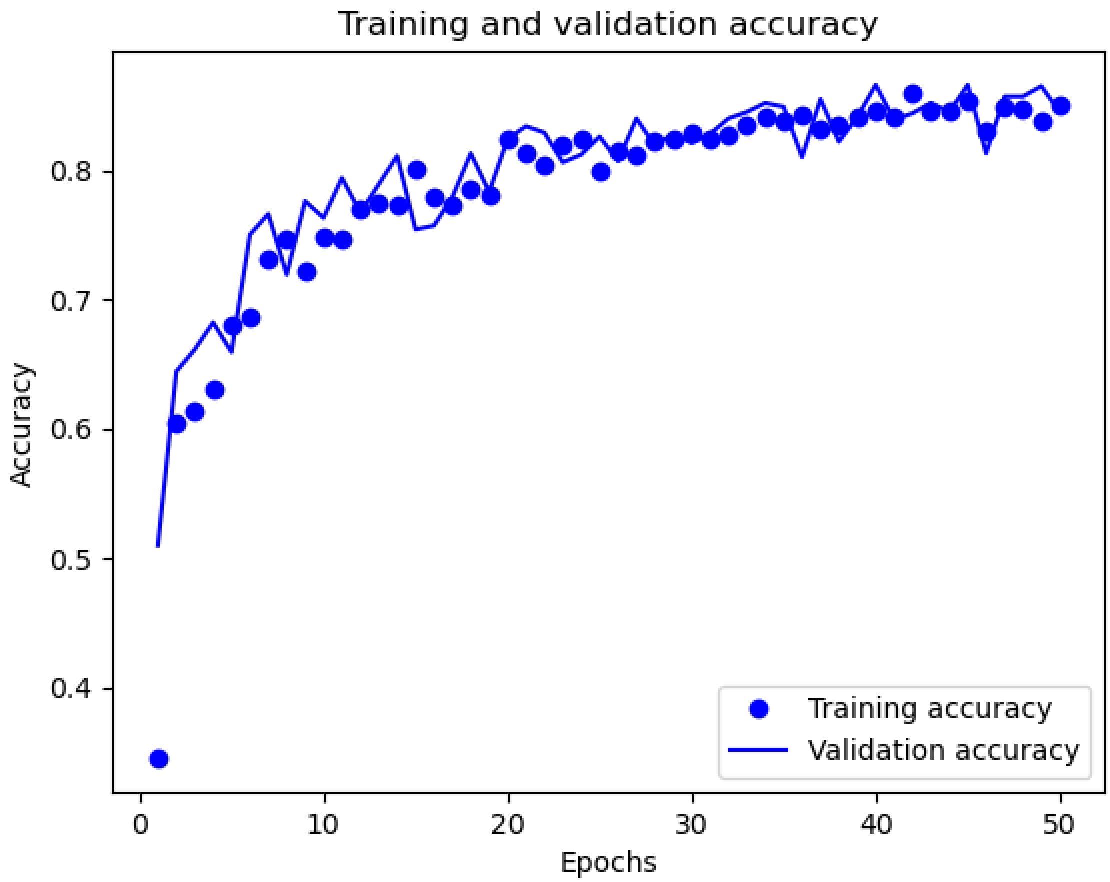
```
history = model.fit(
    train_generator3,
    steps_per_epoch=100,
    epochs=100,
    validation_data=validation_generator3,
    validation_steps=50,
    verbose = 1,
    callbacks=[EarlyStopping(monitor='val_accuracy', patience=10, restore_best_weights


plot_history(history)

test_loss, test_acc = model.evaluate(test_generator3, steps = 100)
print('test_acc:', test_acc)
model_path = "tomato_leaf_classifier_densenet.h5"
model.save(model_path)
```

```
Epoch 1/100
100/100 [==============================] - 174s 2s/step - loss: 6.7753 - accuracy: 0.
3455 - val_loss: 1.4400 - val_accuracy: 0.5100
Epoch 2/100
100/100 [==============================] - 155s 2s/step - loss: 1.2224 - accuracy: 0.
6040 - val_loss: 1.0930 - val_accuracy: 0.6440
Epoch 3/100
100/100 [==============================] - 154s 2s/step - loss: 1.2217 - accuracy: 0.
6132 - val_loss: 0.9998 - val_accuracy: 0.6610
Epoch 4/100
100/100 [==============================] - 155s 2s/step - loss: 1.1315 - accuracy: 0.
6305 - val_loss: 0.9390 - val_accuracy: 0.6820
Epoch 5/100
100/100 [==============================] - 154s 2s/step - loss: 0.9586 - accuracy: 0.
6800 - val_loss: 1.0561 - val_accuracy: 0.6590
Epoch 6/100
100/100 [==============================] - 155s 2s/step - loss: 0.9241 - accuracy: 0.
6855 - val_loss: 0.7305 - val_accuracy: 0.7500
Epoch 7/100
100/100 [==============================] - 155s 2s/step - loss: 0.7963 - accuracy: 0.
7310 - val_loss: 0.7508 - val_accuracy: 0.7660
Epoch 8/100
100/100 [==============================] - 155s 2s/step - loss: 0.7379 - accuracy: 0.
7470 - val_loss: 0.8399 - val_accuracy: 0.7190
Epoch 9/100
100/100 [==============================] - 152s 2s/step - loss: 0.8057 - accuracy: 0.
7211 - val_loss: 0.6672 - val_accuracy: 0.7760
Epoch 10/100
100/100 [==============================] - 154s 2s/step - loss: 0.7778 - accuracy: 0.
7475 - val_loss: 0.7065 - val_accuracy: 0.7630
Epoch 11/100
100/100 [==============================] - 155s 2s/step - loss: 0.7156 - accuracy: 0.
7465 - val_loss: 0.5963 - val_accuracy: 0.7940
Epoch 12/100
100/100 [==============================] - 154s 2s/step - loss: 0.6917 - accuracy: 0.
7700 - val_loss: 0.7306 - val_accuracy: 0.7670
Epoch 13/100
100/100 [==============================] - 155s 2s/step - loss: 0.6718 - accuracy: 0.
7750 - val_loss: 0.6385 - val_accuracy: 0.7890
Epoch 14/100
100/100 [==============================] - 155s 2s/step - loss: 0.6869 - accuracy: 0.
7735 - val_loss: 0.5952 - val_accuracy: 0.8110
Epoch 15/100
100/100 [==============================] - 156s 2s/step - loss: 0.5788 - accuracy: 0.
8005 - val_loss: 0.7539 - val_accuracy: 0.7540
Epoch 16/100
100/100 [==============================] - 156s 2s/step - loss: 0.6711 - accuracy: 0.
7795 - val_loss: 0.6793 - val_accuracy: 0.7570
Epoch 17/100
100/100 [==============================] - 156s 2s/step - loss: 0.6700 - accuracy: 0.
7730 - val_loss: 0.6496 - val_accuracy: 0.7790
Epoch 18/100
100/100 [==============================] - 156s 2s/step - loss: 0.6264 - accuracy: 0.
7855 - val_loss: 0.5133 - val_accuracy: 0.8130
Epoch 19/100
100/100 [==============================] - 156s 2s/step - loss: 0.6513 - accuracy: 0.
7800 - val_loss: 0.6316 - val_accuracy: 0.7830
Epoch 20/100
100/100 [==============================] - 155s 2s/step - loss: 0.5560 - accuracy: 0.
8245 - val_loss: 0.5229 - val_accuracy: 0.8240
```

```
Epoch 21/100
100/100 [==============================] - 156s 2s/step - loss: 0.5707 - accuracy: 0.
8125 - val_loss: 0.5162 - val_accuracy: 0.8340
Epoch 22/100
100/100 [==============================] - 156s 2s/step - loss: 0.5713 - accuracy: 0.
8035 - val_loss: 0.5311 - val_accuracy: 0.8290
Epoch 23/100
100/100 [==============================] - 155s 2s/step - loss: 0.5307 - accuracy: 0.
8200 - val_loss: 0.6176 - val_accuracy: 0.8060
Epoch 24/100
100/100 [==============================] - 155s 2s/step - loss: 0.5495 - accuracy: 0.
8245 - val_loss: 0.5702 - val_accuracy: 0.8120
Epoch 25/100
100/100 [==============================] - 156s 2s/step - loss: 0.5855 - accuracy: 0.
8000 - val_loss: 0.5380 - val_accuracy: 0.8260
Epoch 26/100
100/100 [==============================] - 155s 2s/step - loss: 0.5444 - accuracy: 0.
8155 - val_loss: 0.5355 - val_accuracy: 0.8070
Epoch 27/100
100/100 [==============================] - 156s 2s/step - loss: 0.5751 - accuracy: 0.
8120 - val_loss: 0.4770 - val_accuracy: 0.8400
Epoch 28/100
100/100 [==============================] - 156s 2s/step - loss: 0.5027 - accuracy: 0.
8220 - val_loss: 0.5517 - val_accuracy: 0.8190
Epoch 29/100
100/100 [==============================] - 156s 2s/step - loss: 0.5214 - accuracy: 0.
8240 - val_loss: 0.5392 - val_accuracy: 0.8280
Epoch 30/100
100/100 [==============================] - 156s 2s/step - loss: 0.5201 - accuracy: 0.
8290 - val_loss: 0.5441 - val_accuracy: 0.8210
Epoch 31/100
100/100 [==============================] - 156s 2s/step - loss: 0.5236 - accuracy: 0.
8240 - val_loss: 0.4606 - val_accuracy: 0.8280
Epoch 32/100
100/100 [==============================] - 156s 2s/step - loss: 0.5208 - accuracy: 0.
8275 - val_loss: 0.4377 - val_accuracy: 0.8400
Epoch 33/100
100/100 [==============================] - 156s 2s/step - loss: 0.5030 - accuracy: 0.
8350 - val_loss: 0.4767 - val_accuracy: 0.8450
Epoch 34/100
100/100 [==============================] - 156s 2s/step - loss: 0.4724 - accuracy: 0.
8405 - val_loss: 0.4291 - val_accuracy: 0.8520
Epoch 35/100
100/100 [==============================] - 155s 2s/step - loss: 0.4576 - accuracy: 0.
8380 - val_loss: 0.4069 - val_accuracy: 0.8490
Epoch 36/100
100/100 [==============================] - 156s 2s/step - loss: 0.4315 - accuracy: 0.
8430 - val_loss: 0.5582 - val_accuracy: 0.8100
Epoch 37/100
100/100 [==============================] - 156s 2s/step - loss: 0.4959 - accuracy: 0.
8320 - val_loss: 0.4903 - val_accuracy: 0.8550
Epoch 38/100
100/100 [==============================] - 154s 2s/step - loss: 0.4712 - accuracy: 0.
8345 - val_loss: 0.5126 - val_accuracy: 0.8220
Epoch 39/100
100/100 [==============================] - 155s 2s/step - loss: 0.5037 - accuracy: 0.
8415 - val_loss: 0.4934 - val_accuracy: 0.8410
Epoch 40/100
100/100 [==============================] - 156s 2s/step - loss: 0.4670 - accuracy: 0.
8460 - val_loss: 0.4108 - val_accuracy: 0.8660
```

```
Epoch 41/100
100/100 [==============================] - 155s 2s/step - loss: 0.4773 - accuracy: 0.
8410 - val_loss: 0.5005 - val_accuracy: 0.8390
Epoch 42/100
100/100 [==============================] - 155s 2s/step - loss: 0.4315 - accuracy: 0.
8590 - val_loss: 0.4816 - val_accuracy: 0.8440
Epoch 43/100
100/100 [==============================] - 156s 2s/step - loss: 0.4441 - accuracy: 0.
8450 - val_loss: 0.4184 - val_accuracy: 0.8520
Epoch 44/100
100/100 [==============================] - 155s 2s/step - loss: 0.4739 - accuracy: 0.
8465 - val_loss: 0.4714 - val_accuracy: 0.8450
Epoch 45/100
100/100 [==============================] - 155s 2s/step - loss: 0.4384 - accuracy: 0.
8530 - val_loss: 0.4059 - val_accuracy: 0.8660
Epoch 46/100
100/100 [==============================] - 155s 2s/step - loss: 0.4911 - accuracy: 0.
8310 - val_loss: 0.5731 - val_accuracy: 0.8130
Epoch 47/100
100/100 [==============================] - 155s 2s/step - loss: 0.4226 - accuracy: 0.
8485 - val_loss: 0.4403 - val_accuracy: 0.8570
Epoch 48/100
100/100 [==============================] - 155s 2s/step - loss: 0.4422 - accuracy: 0.
8472 - val_loss: 0.4379 - val_accuracy: 0.8570
Epoch 49/100
100/100 [==============================] - 156s 2s/step - loss: 0.4629 - accuracy: 0.
8385 - val_loss: 0.3908 - val_accuracy: 0.8650
Epoch 50/100
100/100 [==============================] - 155s 2s/step - loss: 0.4522 - accuracy: 0.
8510 - val_loss: 0.4558 - val_accuracy: 0.8450
```






```
100/100 [==============================] - 92s 916ms/step - loss: 0.3740 - accuracy: 0.8710
test_acc: 0.8709999918937683
```

In [124…

```
model_size(model_path)
```

```
Model is approx 836.169824 MB
```

The best model that we have so far is the DenseNet 201 transfer model, with a trained model size of 836 MB, which is likely too big for our offline edge devices. We have an accuracy of 87% for our best model. My hope is that by doing some knowledge distillation we can get something under 32 MB that gets close to 80% accuracy. I don't know the specific constraints for the problem, but I think that this is both achievable and realistic for this kind of problem.

Knowledge Distillation

The process of knowledge distillation is a setup where we have a trained teacher model that acts as second avenue of feedback to a much smaller student model. The student model is given both the input and the teacher information for its training backpropagation. For our purposes we will be using our DenseNet trained model as our teacher model, and we will start with the Base CNN as our first student. We will then step up in size until we get close to 80% accuracy (+/-2%) which would be a 15% increase in accuracy for our small models.

In [128…

```
from utils import Distiller # KD class adapted from the keras library examples
import keras
```

In [133…

```
student = Base_CNN()
distiller = Distiller(student=student, teacher=model) # Takes both models for KD proce
distiller.compile(
    optimizer = keras.optimizers.Adam(),
    student_loss_fn = keras.losses.CategoricalCrossentropy(),
    distillation_loss_fn = keras.losses.KLDivergence(),
    metrics = [keras.metrics.CategoricalAccuracy()],
    alpha = 0.1,
    temperature = 10
)
```

In [135…

```
history = distiller.fit(
    train_generator3,
    steps_per_epoch=100,
    epochs=100,
    validation_data=validation_generator3,
    validation_steps=50,
    verbose = 1,
    callbacks=[EarlyStopping(monitor='val_categorical_accuracy', patience=10, restore_
```

```
Epoch 1/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.292
5 - student_loss: 2.0479 - distillation_loss: 0.0307 - val_categorical_accuracy: 0.08
60 - val_student_loss: 2.4433
Epoch 2/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.336
5 - student_loss: 1.8855 - distillation_loss: 0.0290 - val_categorical_accuracy: 0.16
20 - val_student_loss: 3.3626
Epoch 3/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.407
0 - student_loss: 1.7353 - distillation_loss: 0.0269 - val_categorical_accuracy: 0.22
10 - val_student_loss: 2.5427
Epoch 4/100
100/100 [==============================] - 132s 1s/step - categorical_accuracy: 0.419
5 - student_loss: 1.6798 - distillation_loss: 0.0257 - val_categorical_accuracy: 0.23
60 - val_student_loss: 2.8907
Epoch 5/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.421
5 - student_loss: 1.7337 - distillation_loss: 0.0262 - val_categorical_accuracy: 0.21
20 - val_student_loss: 2.6056
Epoch 6/100
100/100 [==============================] - 134s 1s/step - categorical_accuracy: 0.461
0 - student_loss: 1.5556 - distillation_loss: 0.0239 - val_categorical_accuracy: 0.26
90 - val_student_loss: 2.0648
Epoch 7/100
100/100 [==============================] - 131s 1s/step - categorical_accuracy: 0.487
5 - student_loss: 1.5113 - distillation_loss: 0.0233 - val_categorical_accuracy: 0.48
60 - val_student_loss: 1.5689
Epoch 8/100
100/100 [==============================] - 131s 1s/step - categorical_accuracy: 0.485
0 - student_loss: 1.4685 - distillation_loss: 0.0227 - val_categorical_accuracy: 0.42
70 - val_student_loss: 2.0517
Epoch 9/100
100/100 [==============================] - 134s 1s/step - categorical_accuracy: 0.523
0 - student_loss: 1.3843 - distillation_loss: 0.0216 - val_categorical_accuracy: 0.41
80 - val_student_loss: 2.5555
Epoch 10/100
100/100 [==============================] - 133s 1s/step - categorical_accuracy: 0.548
5 - student_loss: 1.3164 - distillation_loss: 0.0212 - val_categorical_accuracy: 0.31
90 - val_student_loss: 3.5153
Epoch 11/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.551
5 - student_loss: 1.3258 - distillation_loss: 0.0208 - val_categorical_accuracy: 0.44
40 - val_student_loss: 1.7940
Epoch 12/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.551
5 - student_loss: 1.2858 - distillation_loss: 0.0208 - val_categorical_accuracy: 0.50
60 - val_student_loss: 1.2782
Epoch 13/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.530
5 - student_loss: 1.3889 - distillation_loss: 0.0217 - val_categorical_accuracy: 0.27
00 - val_student_loss: 4.2867
Epoch 14/100
100/100 [==============================] - 131s 1s/step - categorical_accuracy: 0.559
5 - student_loss: 1.2750 - distillation_loss: 0.0204 - val_categorical_accuracy: 0.45
90 - val_student_loss: 2.4101
Epoch 15/100
100/100 [==============================] - 131s 1s/step - categorical_accuracy: 0.592
0 - student_loss: 1.2123 - distillation_loss: 0.0196 - val_categorical_accuracy: 0.55
70 - val_student_loss: 0.8901
```

```
Epoch 16/100
100/100 [==============================] - 129s 1s/step - categorical_accuracy: 0.599
5 - student_loss: 1.1913 - distillation_loss: 0.0191 - val_categorical_accuracy: 0.59
80 - val_student_loss: 1.0555
Epoch 17/100
100/100 [==============================] - 134s 1s/step - categorical_accuracy: 0.574
5 - student_loss: 1.2758 - distillation_loss: 0.0202 - val_categorical_accuracy: 0.61
30 - val_student_loss: 2.0561
Epoch 18/100
100/100 [==============================] - 133s 1s/step - categorical_accuracy: 0.571
5 - student_loss: 1.2376 - distillation_loss: 0.0194 - val_categorical_accuracy: 0.59
70 - val_student_loss: 0.8585
Epoch 19/100
100/100 [==============================] - 132s 1s/step - categorical_accuracy: 0.586
5 - student_loss: 1.2207 - distillation_loss: 0.0199 - val_categorical_accuracy: 0.26
90 - val_student_loss: 2.6003
Epoch 20/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.599
5 - student_loss: 1.1387 - distillation_loss: 0.0181 - val_categorical_accuracy: 0.58
40 - val_student_loss: 1.4591
Epoch 21/100
100/100 [==============================] - 131s 1s/step - categorical_accuracy: 0.585
0 - student_loss: 1.2081 - distillation_loss: 0.0189 - val_categorical_accuracy: 0.61
60 - val_student_loss: 0.7015
Epoch 22/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.589
5 - student_loss: 1.2028 - distillation_loss: 0.0190 - val_categorical_accuracy: 0.59
00 - val_student_loss: 1.9274
Epoch 23/100
100/100 [==============================] - 129s 1s/step - categorical_accuracy: 0.589
5 - student_loss: 1.1733 - distillation_loss: 0.0185 - val_categorical_accuracy: 0.57
30 - val_student_loss: 1.3579
Epoch 24/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.602
0 - student_loss: 1.1517 - distillation_loss: 0.0179 - val_categorical_accuracy: 0.37
20 - val_student_loss: 1.4832
Epoch 25/100
100/100 [==============================] - 129s 1s/step - categorical_accuracy: 0.615
2 - student_loss: 1.1451 - distillation_loss: 0.0186 - val_categorical_accuracy: 0.66
80 - val_student_loss: 1.2212
Epoch 26/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.639
0 - student_loss: 1.0726 - distillation_loss: 0.0171 - val_categorical_accuracy: 0.58
50 - val_student_loss: 1.9295
Epoch 27/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.636
0 - student_loss: 1.0916 - distillation_loss: 0.0171 - val_categorical_accuracy: 0.44
60 - val_student_loss: 2.0436
Epoch 28/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.614
5 - student_loss: 1.1554 - distillation_loss: 0.0181 - val_categorical_accuracy: 0.51
50 - val_student_loss: 1.4140
Epoch 29/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.634
0 - student_loss: 1.0612 - distillation_loss: 0.0169 - val_categorical_accuracy: 0.37
20 - val_student_loss: 2.7720
Epoch 30/100
100/100 [==============================] - 129s 1s/step - categorical_accuracy: 0.638
4 - student_loss: 1.0655 - distillation_loss: 0.0174 - val_categorical_accuracy: 0.47
20 - val_student_loss: 1.0632
```

```
Epoch 31/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.610
0 - student_loss: 1.1651 - distillation_loss: 0.0183 - val_categorical_accuracy: 0.37
40 - val_student_loss: 2.0020
Epoch 32/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.625
5 - student_loss: 1.0857 - distillation_loss: 0.0177 - val_categorical_accuracy: 0.34
40 - val_student_loss: 2.8150
Epoch 33/100
100/100 [==============================] - 128s 1s/step - categorical_accuracy: 0.651
5 - student_loss: 1.0193 - distillation_loss: 0.0169 - val_categorical_accuracy: 0.51
50 - val_student_loss: 1.4228
Epoch 34/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.646
0 - student_loss: 1.0587 - distillation_loss: 0.0166 - val_categorical_accuracy: 0.67
80 - val_student_loss: 1.2373
Epoch 35/100
100/100 [==============================] - 131s 1s/step - categorical_accuracy: 0.652
5 - student_loss: 1.0264 - distillation_loss: 0.0161 - val_categorical_accuracy: 0.55
30 - val_student_loss: 1.5131
Epoch 36/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.620
0 - student_loss: 1.1236 - distillation_loss: 0.0176 - val_categorical_accuracy: 0.70
10 - val_student_loss: 0.8184
Epoch 37/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.665
0 - student_loss: 1.0083 - distillation_loss: 0.0164 - val_categorical_accuracy: 0.53
20 - val_student_loss: 1.4865
Epoch 38/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.658
5 - student_loss: 0.9936 - distillation_loss: 0.0157 - val_categorical_accuracy: 0.47
40 - val_student_loss: 2.6707
Epoch 39/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.661
1 - student_loss: 1.0048 - distillation_loss: 0.0161 - val_categorical_accuracy: 0.65
60 - val_student_loss: 1.2406
Epoch 40/100
100/100 [==============================] - 132s 1s/step - categorical_accuracy: 0.676
0 - student_loss: 0.9581 - distillation_loss: 0.0152 - val_categorical_accuracy: 0.42
40 - val_student_loss: 1.9443
Epoch 41/100
100/100 [==============================] - 131s 1s/step - categorical_accuracy: 0.640
0 - student_loss: 1.0966 - distillation_loss: 0.0168 - val_categorical_accuracy: 0.50
50 - val_student_loss: 2.3205
Epoch 42/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.676
5 - student_loss: 0.9639 - distillation_loss: 0.0155 - val_categorical_accuracy: 0.66
40 - val_student_loss: 0.8544
Epoch 43/100
100/100 [==============================] - 130s 1s/step - categorical_accuracy: 0.673
0 - student_loss: 1.0147 - distillation_loss: 0.0163 - val_categorical_accuracy: 0.68
40 - val_student_loss: 1.4043
Epoch 44/100
100/100 [==============================] - 131s 1s/step - categorical_accuracy: 0.677
0 - student_loss: 0.9464 - distillation_loss: 0.0152 - val_categorical_accuracy: 0.60
40 - val_student_loss: 0.9938
Epoch 45/100
100/100 [==============================] - 131s 1s/step - categorical_accuracy: 0.689
0 - student_loss: 0.9440 - distillation_loss: 0.0150 - val_categorical_accuracy: 0.47
50 - val_student_loss: 1.7083
```

```
Epoch 46/100
100/100 [==============================] - 131s 1s/step - categorical_accuracy: 0.679
5 - student_loss: 0.9480 - distillation_loss: 0.0152 - val_categorical_accuracy: 0.66
00 - val_student_loss: 0.9039
```

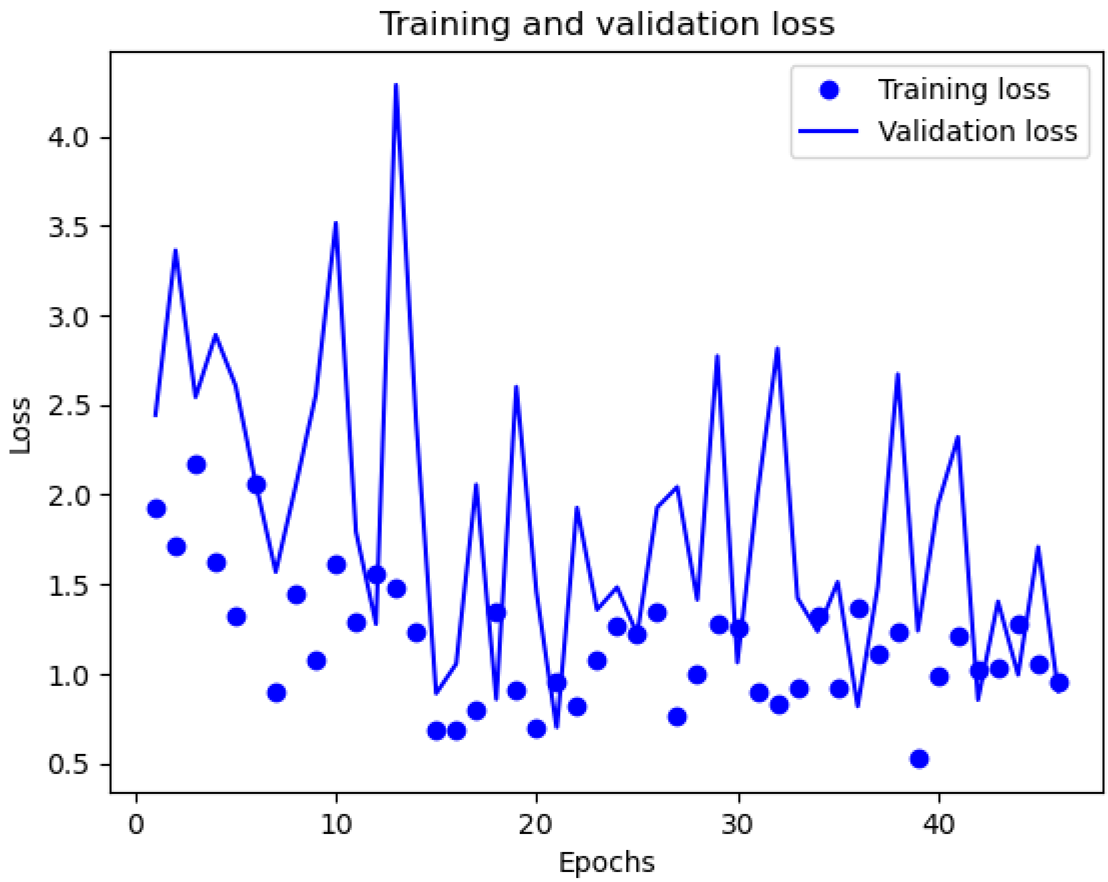
```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In [135], line 11
      1 history = distiller.fit(
      2     train_generator3,
      3     steps_per_epoch=100,
   (...)
      7     verbose = 1,
      8     callbacks=[EarlyStopping(monitor='val_categorical_accuracy', patience=10,
restore_best_weights = True)])
---> 11 plot_history(history)
     13 test_loss, test_acc = model.evaluate(test_generator3, steps = 100)
     14 print('test_acc:', test_acc)

Cell In [40], line 4, in plot_history(history)
      2 def plot_history(history):
      3   history_dict = history.history
----> 4   loss_values = history_dict['loss']
      5   val_loss_values = history_dict['val_loss']
      6   acc_values = history_dict['accuracy']

KeyError: 'loss'
```

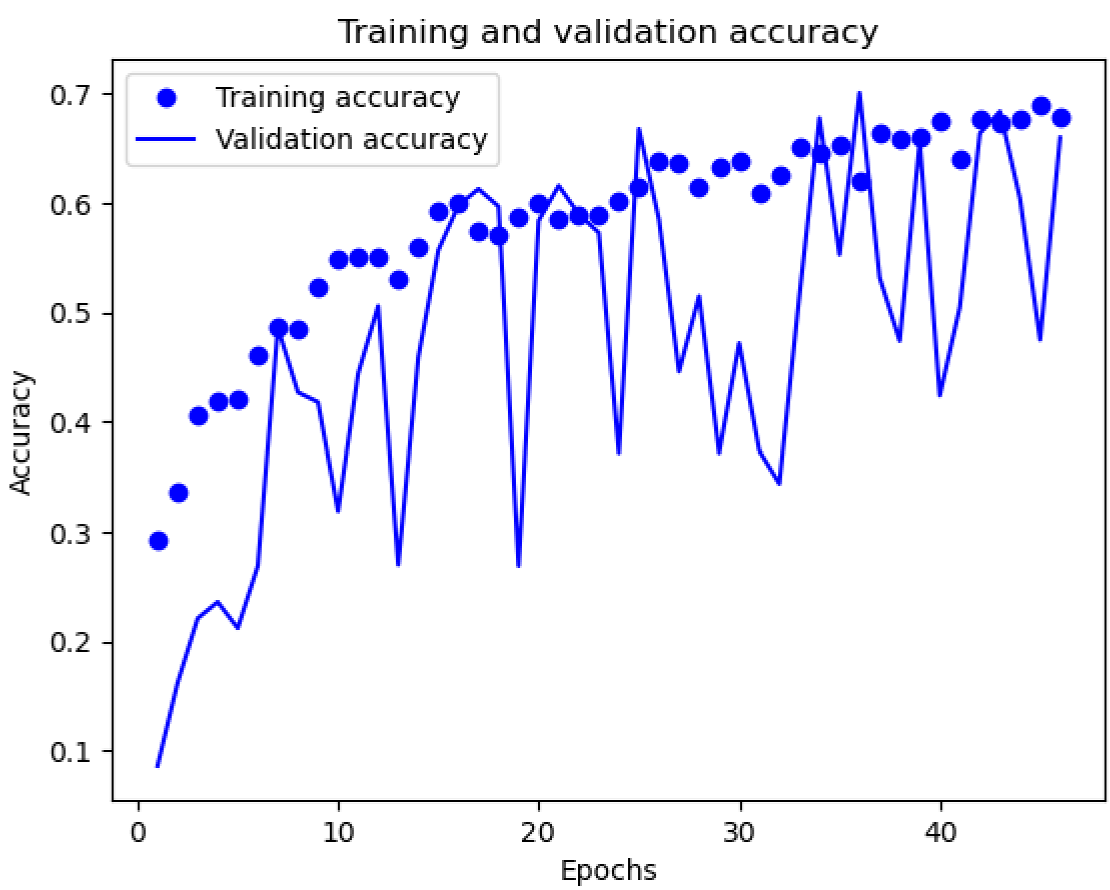
In [148…

```python
# Build a plotting function
def distiller_plot_history(history):
  history_dict = history.history
  loss_values = history_dict['student_loss']
  val_loss_values = history_dict['val_student_loss']
  acc_values =  history_dict['categorical_accuracy']
  val_acc_values = history_dict['val_categorical_accuracy']
  epochs = range(1, len(history_dict['categorical_accuracy']) + 1)

  plt.plot(epochs, loss_values, 'bo', label = 'Training loss')
  plt.plot(epochs, val_loss_values, 'b', label = 'Validation loss')
  plt.title('Training and validation loss')
  plt.xlabel('Epochs')
  plt.ylabel('Loss')
  plt.legend()
  plt.show()

  plt.plot(epochs, acc_values, 'bo', label = 'Training accuracy')
  plt.plot(epochs, val_acc_values, 'b', label = 'Validation accuracy')
  plt.title('Training and validation accuracy')
  plt.xlabel('Epochs')
  plt.ylabel('Accuracy')
  plt.legend()
  return plt.show()
```

In [149…

```python
distiller_plot_history(history)
```


Training and validation loss
Training loss
Validation loss
Loss
Epochs
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0
10
20
30
40


Training and validation accuracy
Training accuracy
Validation accuracy
Accuracy
Epochs
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
10
20
30
40

These graphs clearly illustrate the problem that could arise with small patience windows, where our accuracy could spike and dip for several epochs in a row.

In [143…
```
student.evaluate(test_generator3, steps = 100)
```

```
100/100 [==============================] - 11s 100ms/step - loss: 0.9002 - accuracy: 0.7115
```

Out[143]:
```
[0.9001874327659607, 0.7114999890327454]
```

In [144…
```
model_path = "tomato_leaf_classifier_kd.h5"
student.save(model_path)
```

In [145…
```
model_size(model_path)
```

```
Model is approx 1.816936 MB
```

Improved Base_CNN accuracy from 64.75% to 71.15% and at a size of just 1.82 MB. With our best model giving us near 87% accuracy lets see if we can get a smaller model closer to that value.

In [154…
```
student = Next_CNN()
distiller = Distiller(student=student, teacher=model)
distiller.compile(
    optimizer = keras.optimizers.Adam(),
    student_loss_fn = keras.losses.CategoricalCrossentropy(),
    distillation_loss_fn = keras.losses.KLDivergence(),
    metrics = [keras.metrics.CategoricalAccuracy()],
    alpha = 0.1,
    temperature = 10
)
```

In [155…
```
history = distiller.fit(
    train_generator3,
    steps_per_epoch=100,
    epochs=100,
    validation_data=validation_generator3,
    validation_steps=50,
    verbose = 1,
    callbacks=[EarlyStopping(monitor='val_categorical_accuracy', patience=10, restore_
```

```
Epoch 1/100
100/100 [==============================] - 162s 2s/step - categorical_accuracy: 0.305
0 - student_loss: 2.1138 - distillation_loss: 0.0328 - val_categorical_accuracy: 0.12
40 - val_student_loss: 3.0044
Epoch 2/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.417
5 - student_loss: 1.7190 - distillation_loss: 0.0271 - val_categorical_accuracy: 0.12
60 - val_student_loss: 5.5132
Epoch 3/100
100/100 [==============================] - 153s 2s/step - categorical_accuracy: 0.481
1 - student_loss: 1.5569 - distillation_loss: 0.0247 - val_categorical_accuracy: 0.12
40 - val_student_loss: 5.0516
Epoch 4/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.503
5 - student_loss: 1.3985 - distillation_loss: 0.0233 - val_categorical_accuracy: 0.12
00 - val_student_loss: 5.2131
Epoch 5/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.569
5 - student_loss: 1.2544 - distillation_loss: 0.0212 - val_categorical_accuracy: 0.23
30 - val_student_loss: 2.6008
Epoch 6/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.576
0 - student_loss: 1.2267 - distillation_loss: 0.0198 - val_categorical_accuracy: 0.33
30 - val_student_loss: 2.1153
Epoch 7/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.577
5 - student_loss: 1.2149 - distillation_loss: 0.0197 - val_categorical_accuracy: 0.40
40 - val_student_loss: 2.8616
Epoch 8/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.606
5 - student_loss: 1.1304 - distillation_loss: 0.0187 - val_categorical_accuracy: 0.36
10 - val_student_loss: 2.2302
Epoch 9/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.610
5 - student_loss: 1.1182 - distillation_loss: 0.0184 - val_categorical_accuracy: 0.50
20 - val_student_loss: 1.9881
Epoch 10/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.640
4 - student_loss: 1.0506 - distillation_loss: 0.0178 - val_categorical_accuracy: 0.53
50 - val_student_loss: 1.2556
Epoch 11/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.635
5 - student_loss: 1.0424 - distillation_loss: 0.0174 - val_categorical_accuracy: 0.22
40 - val_student_loss: 4.2424
Epoch 12/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.643
0 - student_loss: 1.0228 - distillation_loss: 0.0169 - val_categorical_accuracy: 0.54
80 - val_student_loss: 1.5999
Epoch 13/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.658
5 - student_loss: 0.9725 - distillation_loss: 0.0163 - val_categorical_accuracy: 0.60
60 - val_student_loss: 0.9686
Epoch 14/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.678
0 - student_loss: 0.9144 - distillation_loss: 0.0158 - val_categorical_accuracy: 0.47
00 - val_student_loss: 1.6537
Epoch 15/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.669
0 - student_loss: 0.9235 - distillation_loss: 0.0157 - val_categorical_accuracy: 0.43
50 - val_student_loss: 1.8233
```

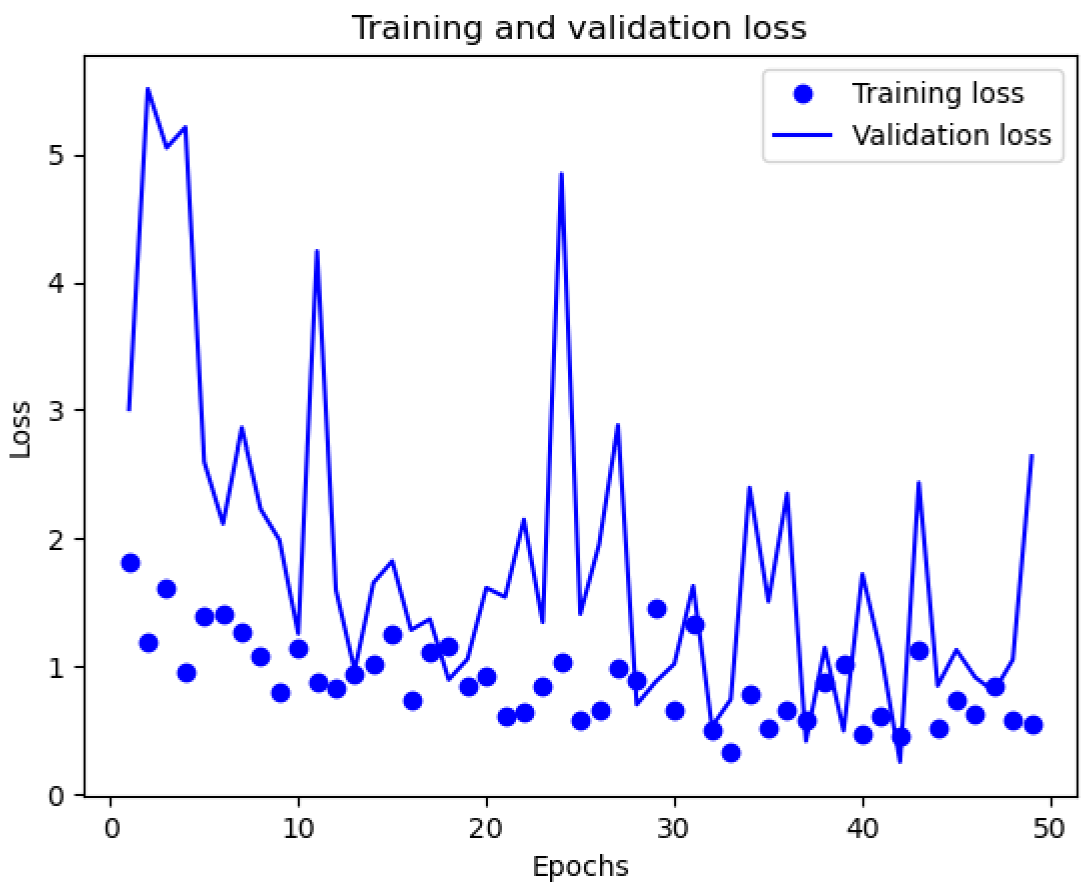
```
Epoch 16/100
100/100 [==============================] - 157s 2s/step - categorical_accuracy: 0.669
5 - student_loss: 0.9191 - distillation_loss: 0.0156 - val_categorical_accuracy: 0.60
30 - val_student_loss: 1.2839
Epoch 17/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.690
0 - student_loss: 0.8691 - distillation_loss: 0.0147 - val_categorical_accuracy: 0.53
50 - val_student_loss: 1.3689
Epoch 18/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.671
5 - student_loss: 0.9317 - distillation_loss: 0.0155 - val_categorical_accuracy: 0.61
80 - val_student_loss: 0.8948
Epoch 19/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.708
5 - student_loss: 0.8288 - distillation_loss: 0.0148 - val_categorical_accuracy: 0.60
20 - val_student_loss: 1.0560
Epoch 20/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.704
5 - student_loss: 0.8263 - distillation_loss: 0.0135 - val_categorical_accuracy: 0.59
60 - val_student_loss: 1.6140
Epoch 21/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.711
0 - student_loss: 0.8324 - distillation_loss: 0.0150 - val_categorical_accuracy: 0.47
90 - val_student_loss: 1.5424
Epoch 22/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.713
5 - student_loss: 0.8117 - distillation_loss: 0.0138 - val_categorical_accuracy: 0.46
00 - val_student_loss: 2.1488
Epoch 23/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.726
5 - student_loss: 0.7792 - distillation_loss: 0.0137 - val_categorical_accuracy: 0.49
60 - val_student_loss: 1.3453
Epoch 24/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.721
0 - student_loss: 0.8004 - distillation_loss: 0.0135 - val_categorical_accuracy: 0.28
80 - val_student_loss: 4.8470
Epoch 25/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.733
0 - student_loss: 0.7978 - distillation_loss: 0.0137 - val_categorical_accuracy: 0.63
20 - val_student_loss: 1.4095
Epoch 26/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.720
5 - student_loss: 0.7721 - distillation_loss: 0.0136 - val_categorical_accuracy: 0.50
70 - val_student_loss: 1.9551
Epoch 27/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.709
5 - student_loss: 0.8407 - distillation_loss: 0.0140 - val_categorical_accuracy: 0.35
30 - val_student_loss: 2.8799
Epoch 28/100
100/100 [==============================] - 153s 2s/step - categorical_accuracy: 0.709
5 - student_loss: 0.7994 - distillation_loss: 0.0139 - val_categorical_accuracy: 0.62
50 - val_student_loss: 0.7002
Epoch 29/100
100/100 [==============================] - 153s 2s/step - categorical_accuracy: 0.752
4 - student_loss: 0.7292 - distillation_loss: 0.0130 - val_categorical_accuracy: 0.67
40 - val_student_loss: 0.8773
Epoch 30/100
100/100 [==============================] - 156s 2s/step - categorical_accuracy: 0.739
5 - student_loss: 0.7576 - distillation_loss: 0.0128 - val_categorical_accuracy: 0.72
50 - val_student_loss: 1.0186
```

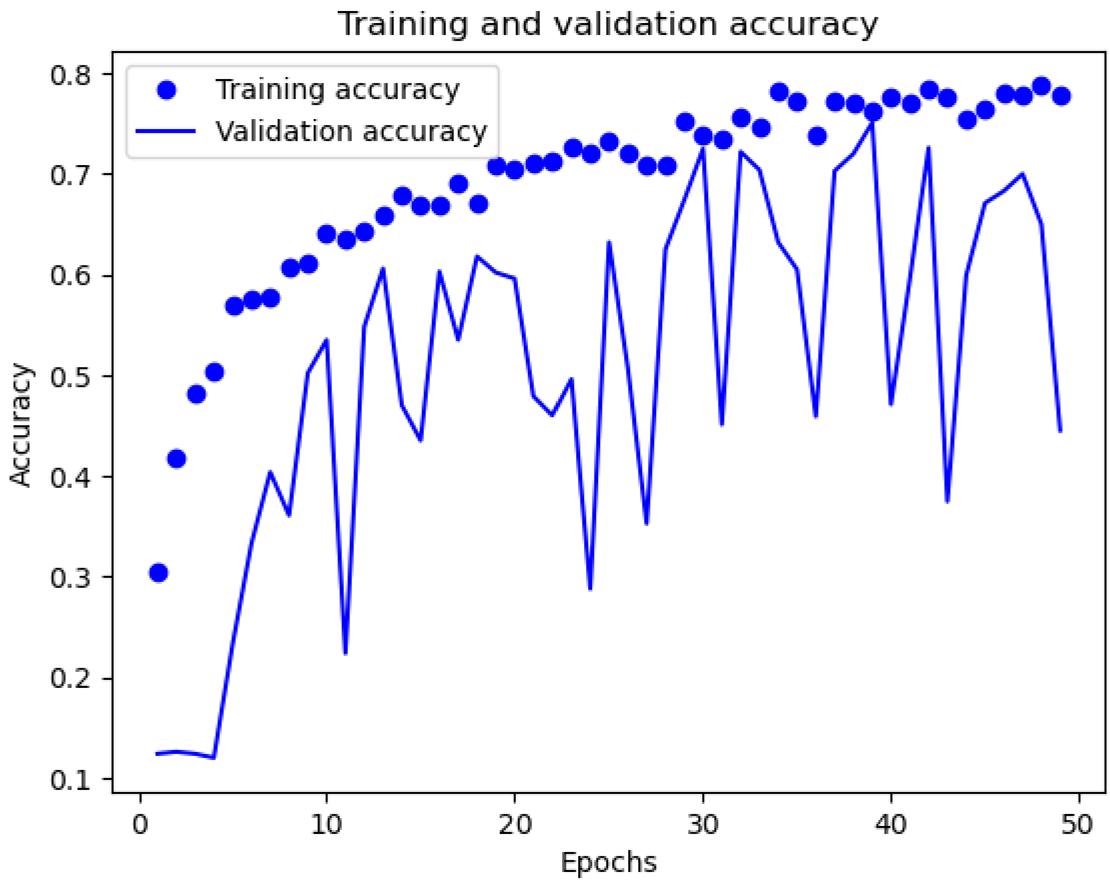
```
Epoch 31/100
100/100 [==============================] - 156s 2s/step - categorical_accuracy: 0.735
5 - student_loss: 0.7803 - distillation_loss: 0.0134 - val_categorical_accuracy: 0.45
10 - val_student_loss: 1.6294
Epoch 32/100
100/100 [==============================] - 157s 2s/step - categorical_accuracy: 0.756
5 - student_loss: 0.6868 - distillation_loss: 0.0120 - val_categorical_accuracy: 0.72
20 - val_student_loss: 0.5273
Epoch 33/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.746
0 - student_loss: 0.7159 - distillation_loss: 0.0128 - val_categorical_accuracy: 0.70
40 - val_student_loss: 0.7404
Epoch 34/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.782
0 - student_loss: 0.6611 - distillation_loss: 0.0115 - val_categorical_accuracy: 0.63
20 - val_student_loss: 2.3984
Epoch 35/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.772
0 - student_loss: 0.6479 - distillation_loss: 0.0117 - val_categorical_accuracy: 0.60
50 - val_student_loss: 1.5078
Epoch 36/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.738
0 - student_loss: 0.7037 - distillation_loss: 0.0131 - val_categorical_accuracy: 0.45
90 - val_student_loss: 2.3524
Epoch 37/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.773
0 - student_loss: 0.6373 - distillation_loss: 0.0122 - val_categorical_accuracy: 0.70
30 - val_student_loss: 0.4144
Epoch 38/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.769
5 - student_loss: 0.6720 - distillation_loss: 0.0124 - val_categorical_accuracy: 0.72
00 - val_student_loss: 1.1460
Epoch 39/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.762
0 - student_loss: 0.6715 - distillation_loss: 0.0120 - val_categorical_accuracy: 0.75
10 - val_student_loss: 0.4947
Epoch 40/100
100/100 [==============================] - 153s 2s/step - categorical_accuracy: 0.776
0 - student_loss: 0.6558 - distillation_loss: 0.0118 - val_categorical_accuracy: 0.47
10 - val_student_loss: 1.7218
Epoch 41/100
100/100 [==============================] - 156s 2s/step - categorical_accuracy: 0.769
5 - student_loss: 0.6387 - distillation_loss: 0.0119 - val_categorical_accuracy: 0.59
50 - val_student_loss: 1.0955
Epoch 42/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.784
0 - student_loss: 0.6123 - distillation_loss: 0.0112 - val_categorical_accuracy: 0.72
60 - val_student_loss: 0.2513
Epoch 43/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.777
0 - student_loss: 0.6633 - distillation_loss: 0.0119 - val_categorical_accuracy: 0.37
50 - val_student_loss: 2.4370
Epoch 44/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.754
5 - student_loss: 0.6864 - distillation_loss: 0.0125 - val_categorical_accuracy: 0.59
90 - val_student_loss: 0.8470
Epoch 45/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.763
5 - student_loss: 0.6771 - distillation_loss: 0.0117 - val_categorical_accuracy: 0.67
10 - val_student_loss: 1.1282
```

```
Epoch 46/100
100/100 [==============================] - 154s 2s/step - categorical_accuracy: 0.780
5 - student_loss: 0.6148 - distillation_loss: 0.0111 - val_categorical_accuracy: 0.68
30 - val_student_loss: 0.9128
Epoch 47/100
100/100 [==============================] - 155s 2s/step - categorical_accuracy: 0.778
0 - student_loss: 0.6671 - distillation_loss: 0.0116 - val_categorical_accuracy: 0.70
00 - val_student_loss: 0.8023
Epoch 48/100
100/100 [==============================] - 156s 2s/step - categorical_accuracy: 0.788
5 - student_loss: 0.6042 - distillation_loss: 0.0106 - val_categorical_accuracy: 0.65
00 - val_student_loss: 1.0522
Epoch 49/100
100/100 [==============================] - 162s 2s/step - categorical_accuracy: 0.778
5 - student_loss: 0.6217 - distillation_loss: 0.0112 - val_categorical_accuracy: 0.44
50 - val_student_loss: 2.6440
```

In [156…

```
distiller_plot_history(history)
```





In [157…

```
student.evaluate(test_generator3, steps = 100)
```

```
100/100 [==============================] - 21s 204ms/step - loss: 0.6120 - accuracy:
0.7865
```

Out[157]:

```
[0.6120110154151917, 0.7864999771118164]
```

In [158…

```
model_path = "tomato_leaf_classifier_kd2.h5"
student.save(model_path)
```

In [159…

```
model_size(model_path)
```

```
Model is approx 7.073504 MB
```

We now have a knowledge distillation model that is 79% accurate at just 7.07 MB. This fits within my original goal of under 32 MB, albeit slightly lower accuracy than I had hoped for. I bet with experimentation we should be able to find something above 80% while staying below my memory target, but I am happy with where this is at for a proof of concept.

Conclusion

The process of generating a CNN, finding a superior transfer learning model, and using knowledge distillation to create an improved small CNN for edge device use appears to be possible for this problem. Starting with a CNN with 64.75% accuracy, we were able to improve its accuracy to 71.15% while having a memory footprint of just 1.82 MB. Increasing the CNN's size and continuing with the same knowledge distillation technique we were able to create a

7.07 MB model with 78.65% accuracy. I was not able to find a combination that satisfied my goal of 80% accuracy and under 32 MB, but I do believe that continued experimentation can get us there. Additionally, we should that data augmentation was not needed with the number of examples available, but that it also did appear to make the models more resistant to over-fitting. This process for creating a small and accurate model appears to be viable for the problem, and additional compression techniques could be used as well. I believe that future experimentation could use the smallest available DenseNet for transfer learning, use it as the student model, and then apply pruning, quantization, and half precision data types for an inference version with nearly the same accuracy as our best large model but at just 64 MB.