

PLANT DISEASE DIAGNOSTIC

Deep Learning

TEAM

- Anna Dahlhaus
 - github user: a-dahl
- Christian Grundman
 - github user: cgrundman
- Pedro Andres Alba Diaz
 - github user: pedroandresAD

CONTENT

- Motivation
- Objectives
- Data Analysis
- Preprocessing
- Models
- Further Improvement
- Conclusion

MOTIVATION



Early diagnosis of plant diseases allows preventive measures to reduce economic and production damage.

Traditional methods are limited by available expertise.

Plant surveillance by farmers is not always practical or cost-effective.

OBJECTIVES

Evaluate different databases and implement various neural network architectures for classifying plant diseases.



AVAILABLE DATASETS



- Background of the photos
- Number of classes and images (diseases and species)
- Image properties (dimensions, resolution, distribution, etc.)

NEW PLANT DISEASE DATASET

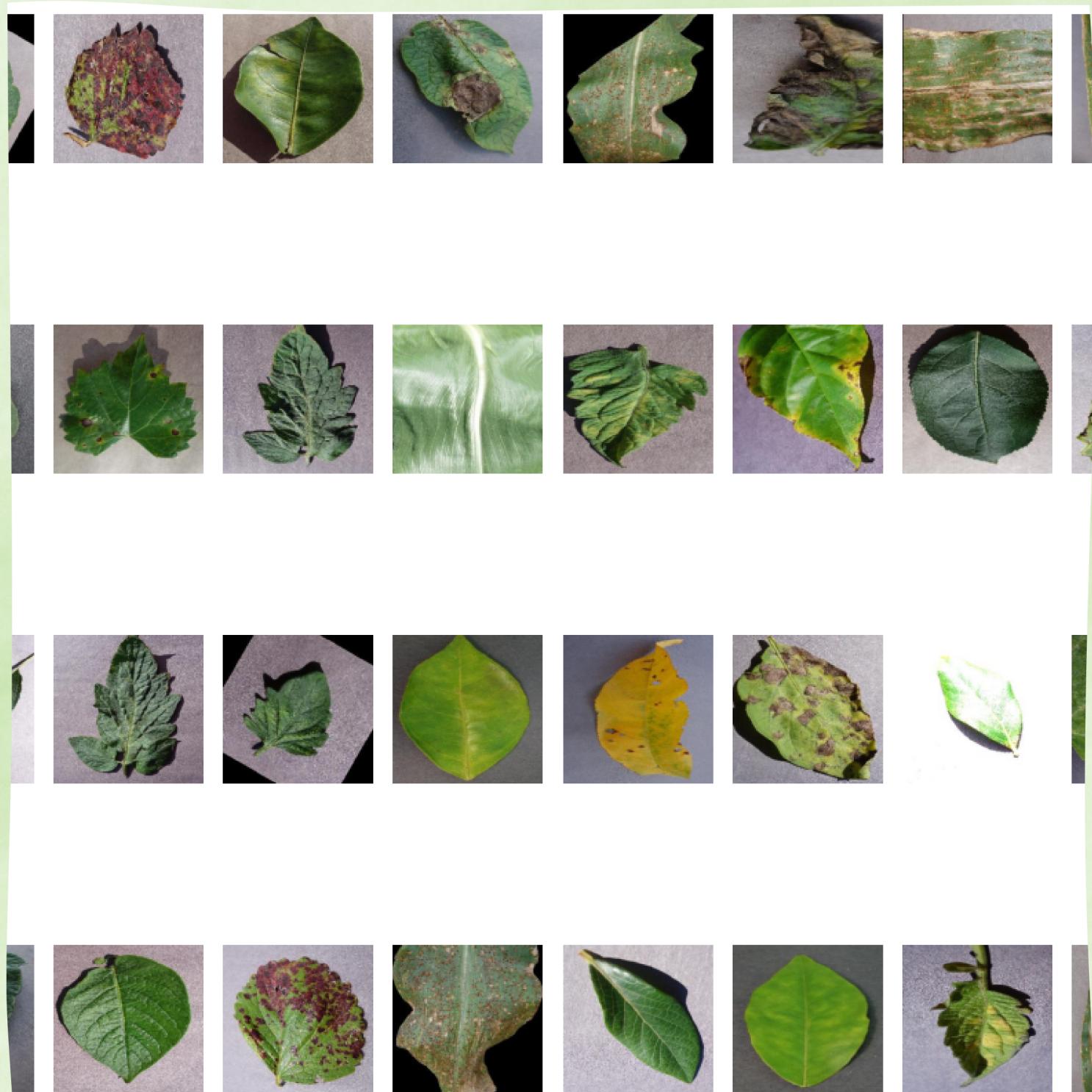
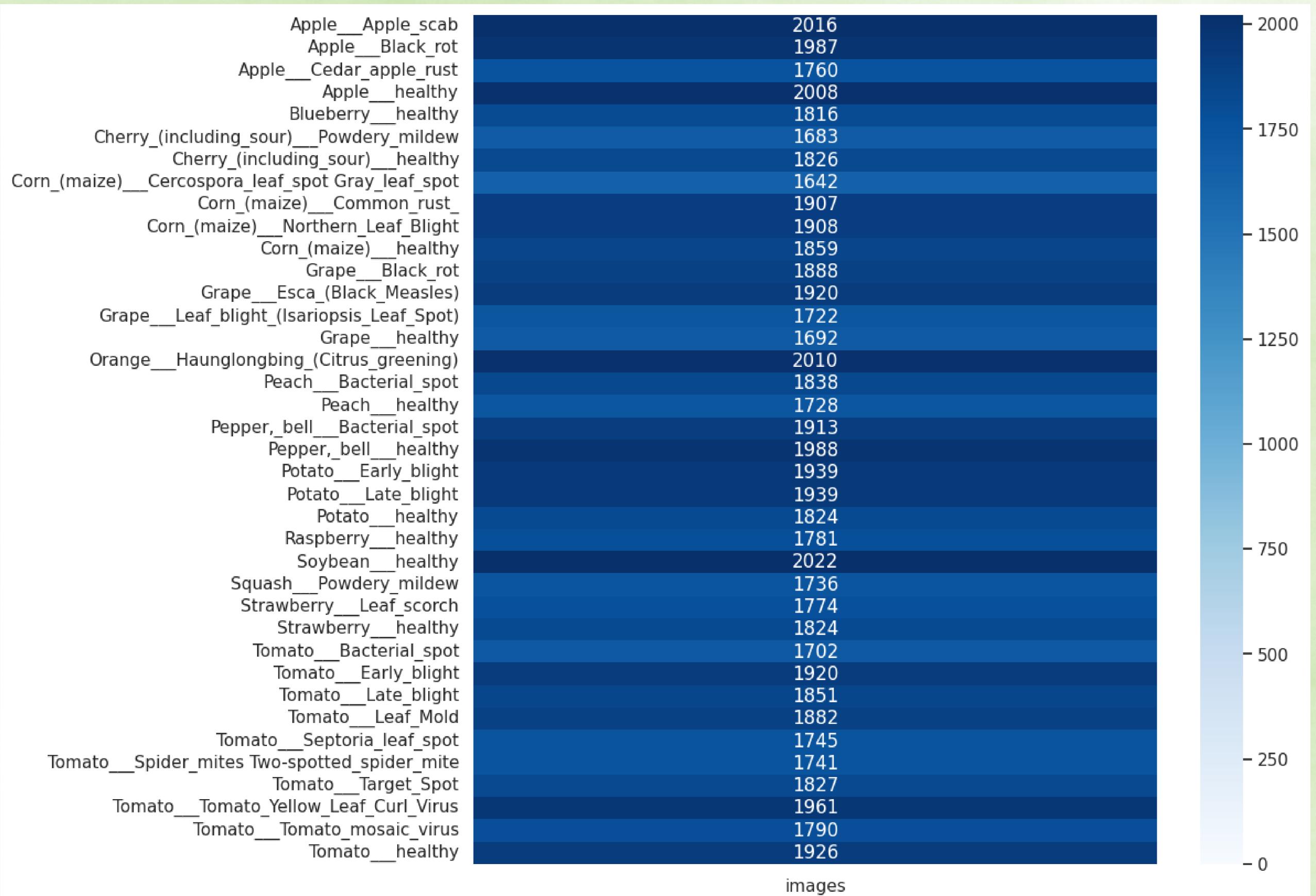


Image Data set for Plant
Disease detection.

- Images of 256x256 pixels
- Image processing algorithms were pre-implemented to detect diseases through the colour characteristics.
- 38 classes

DATA DISTRIBUTION



PREPROCESSING

Ensure the same dimension between images 256*256*3 and normalise.

Create dataset from directory by:

- a. Flow from directory_(ImageDataGenerator)
- b. keras.utils.image_dataset_from_directory

```
def image_generator():

    train_datagen = ImageDataGenerator(rescale=1./255)

    # - class_mode: How the labels are represented. Should be one of "binary", "categorical" or "sparse".
    #           Pick the one that better suits here given that the labels are going to be 1D binary labels.
    train_generator = train_datagen.flow_from_directory(directory= folder_plant_diseases_new,
                                                          target_size=(256, 256),
                                                          batch_size=64,
                                                          shuffle=True,
                                                          keep_aspect_ratio=True,
                                                          class_mode="categorical")
    #class_mode="categorical")

    return train_generator

data = image_generator()
```

MODELS

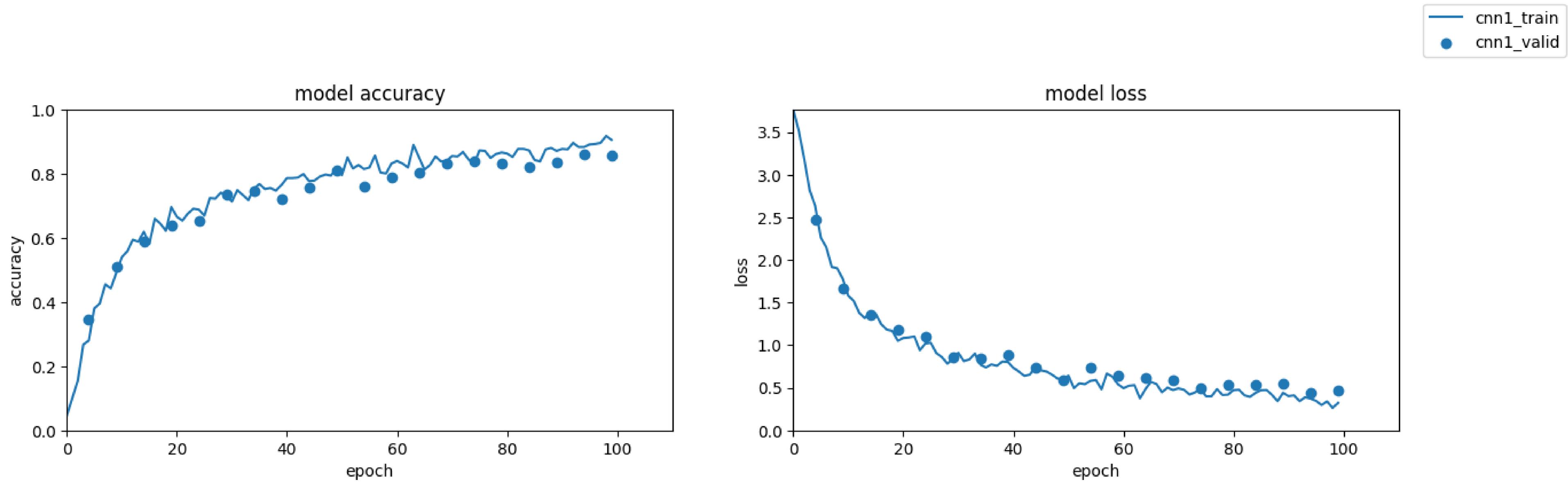
1. CNN: 3 Conv Layers and 3 FC layers
2. CNN: 3 Conv Layers and 4 FC layers
3. CNN: 4 Conv Layers and 3 FC layers
4. CNN: 3 Conv Layers and 3 FC layers and dropout in dense layers
5. CNN: 3 Conv Layers and 3 FC layers and dropout in conv layers
6. Neural Network to classify by species
7. Neural Network for tomato diseases

MODEL 1

Model: "sequential_2"

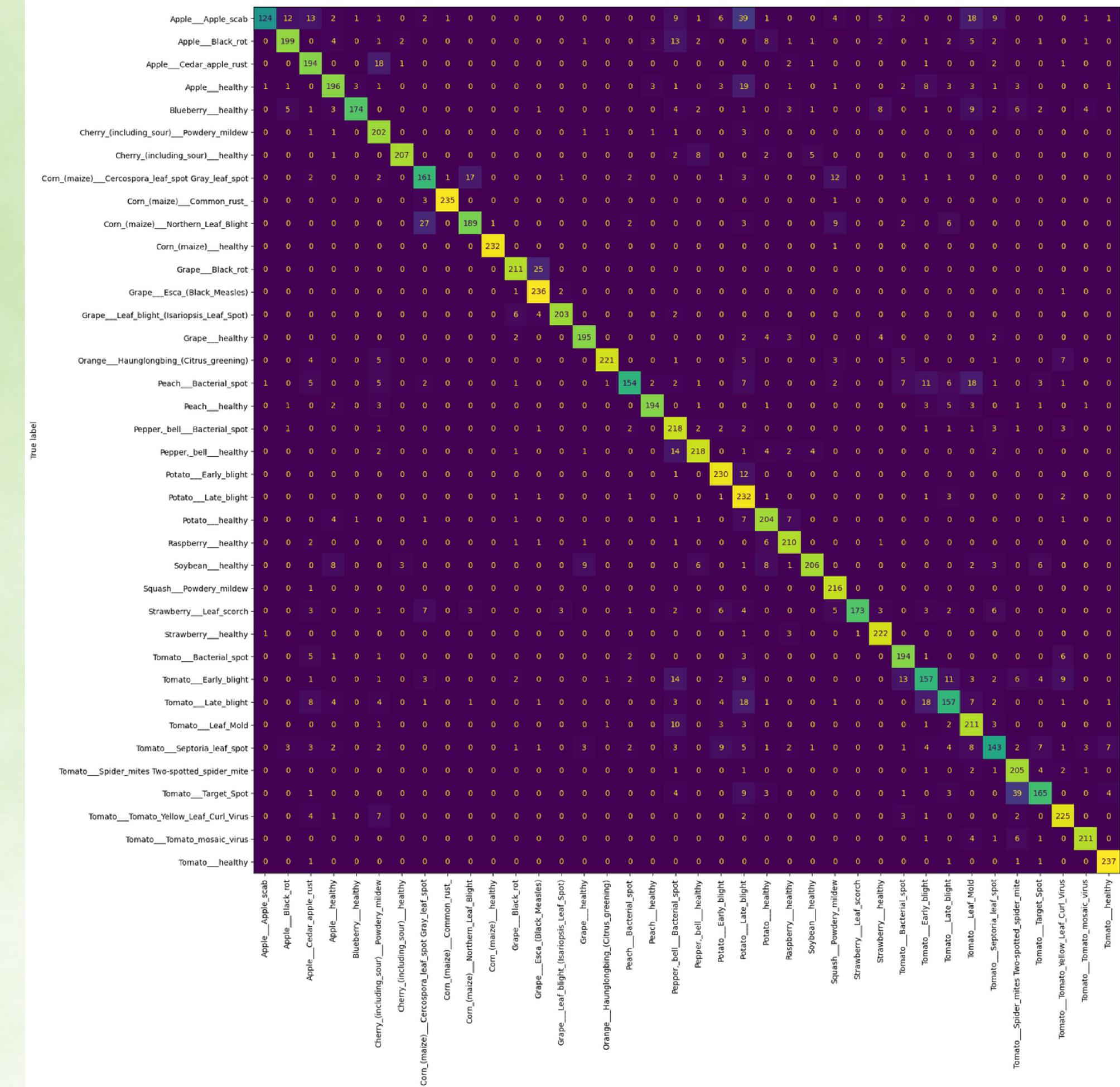
Layer (type)	Output Shape	Param #
<hr/>		
conv2d_6 (Conv2D)	(None, 254, 254, 16)	448
max_pooling2d_6 (MaxPooling 2D)	(None, 127, 127, 16)	0
conv2d_7 (Conv2D)	(None, 125, 125, 32)	4640
max_pooling2d_7 (MaxPooling 2D)	(None, 62, 62, 32)	0
conv2d_8 (Conv2D)	(None, 60, 60, 64)	18496
max_pooling2d_8 (MaxPooling 2D)	(None, 30, 30, 64)	0
flatten_2 (Flatten)	(None, 57600)	0
dense_6 (Dense)	(None, 512)	29491712
dense_7 (Dense)	(None, 256)	131328
dense_8 (Dense)	(None, 38)	9766
<hr/>		
Total params: 29,656,390		
Trainable params: 29,656,390		
Non-trainable params: 0		

EVOLUTION OF ACCURACY AND LOSS OVER THE EPOCHS

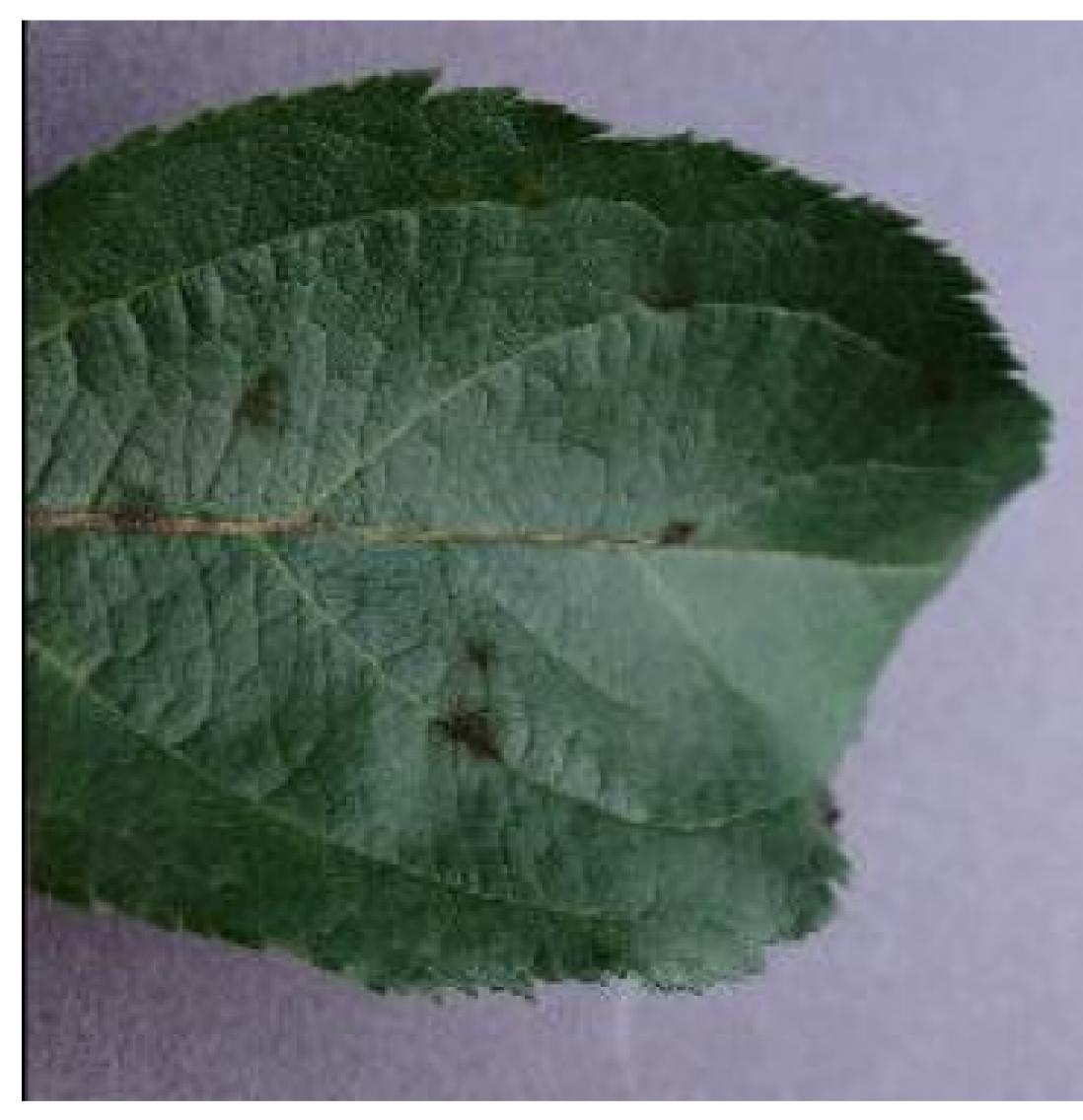


CONFUSION MATRIX

		precision	recall	f1-score	su
	Apple__Apple_scab	0.98	0.49	0.65	
	Apple__Black_rot	0.99	0.89	0.89	
	Apple__Cedar_apple_rust	0.78	0.88	0.83	
	Apple__healthy	0.86	0.78	0.82	
	Blueberry__healthy	0.97	0.77	0.86	
	Cherry_(including_sour)__Powdery_mildew	0.78	0.96	0.86	
	Cherry_(including_sour)__healthy	0.97	0.91	0.94	
Corn_(maize)	Cercospora_leaf_spot_Gray_leaf_spot	0.78	0.79	0.78	
	Corn_(maize)__Common_rust_	0.99	0.98	0.99	
	Corn_(maize)__Northern_Leaf_Blight	0.99	0.79	0.84	
	Corn_(maize)__healthy	1.00	1.00	1.00	
	Grape__Black_rot	0.93	0.89	0.91	
	Grape__Esca_(Black_Measles)	0.87	0.98	0.92	
Grape	Leaf_blight_(Isariopsis_Leaf_Spot)	0.97	0.94	0.96	
	Grape__healthy	0.92	0.92	0.92	
	Orange__Haunglongbing_(Citrus_greening)	0.98	0.88	0.93	
	Peach__Bacterial_spot	0.92	0.67	0.77	
	Peach__healthy	0.96	0.99	0.93	
	Pepper,_bell__Bacterial_spot	0.71	0.91	0.80	
	Pepper,_bell__healthy	0.99	0.88	0.89	
	Potato__Early_blight	0.86	0.95	0.90	
	Potato__Late_blight	0.59	0.95	0.73	
	Potato__healthy	0.84	0.89	0.86	
	Raspberry__healthy	0.89	0.94	0.92	
	Soybean__healthy	0.94	0.81	0.87	
	Squash__Powdery_mildew	0.85	1.00	0.92	
	Strawberry__Leaf_scorch	0.99	0.78	0.87	
	Strawberry__healthy	0.91	0.97	0.94	
	Tomato__Bacterial_spot	0.84	0.91	0.87	
	Tomato__Early_blight	0.73	0.65	0.69	
	Tomato__Late_blight	0.76	0.68	0.72	
	Tomato__Leaf_Mold	0.71	0.90	0.79	
	Tomato__Septoria_leaf_spot	0.77	0.66	0.71	
Tomato	Spider_mites_Two-spotted_spider_mite	0.75	0.94	0.84	
	Tomato__Target_Spot	0.85	0.72	0.78	
	Tomato__Tomato_Yellow_Leaf_Curl_Virus	0.87	0.92	0.89	
	Tomato__Tomato_mosaic_virus	0.95	0.94	0.95	
	Tomato__healthy	0.94	0.98	0.96	
	accuracy				0.86
	macro avg	0.87	0.86	0.86	
	weighted avg	0.87	0.86	0.86	



MISSMATCHED IMAGES



Apple scab misslabeled
as potato Late blight

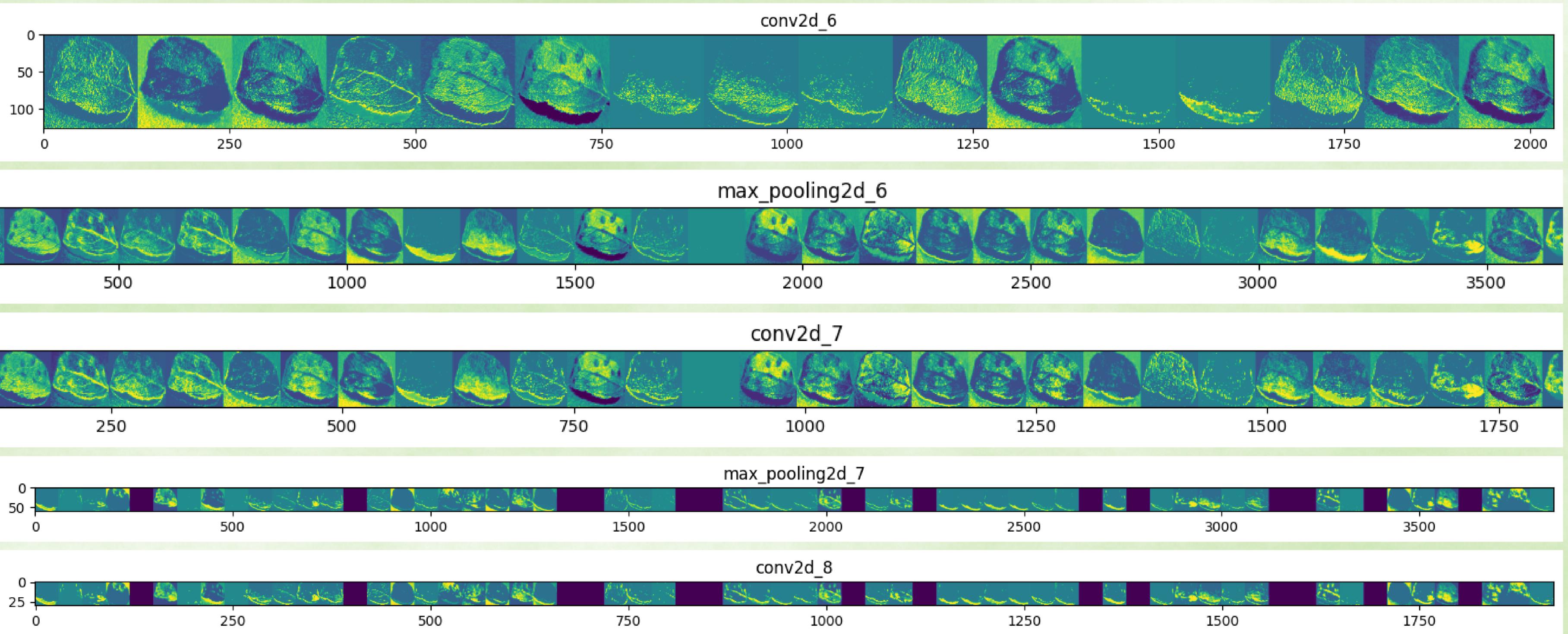


Potato Late blight



Potato Late blight

FEATURE EXTRACTION

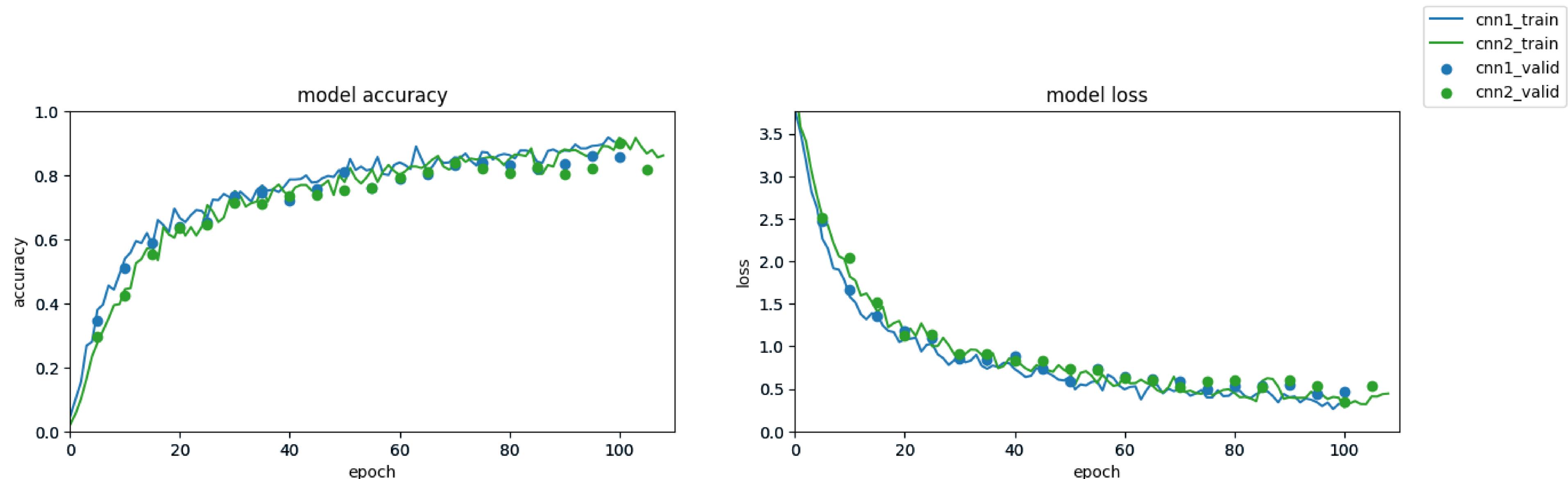


MODEL 2

Model: "sequential_52"

Layer (type)	Output Shape	Param #
<hr/>		
conv2d_134 (Conv2D)	(None, 254, 254, 16)	448
max_pooling2d_134 (MaxPooling2D)	(None, 127, 127, 16)	0
conv2d_135 (Conv2D)	(None, 125, 125, 32)	4640
max_pooling2d_135 (MaxPooling2D)	(None, 62, 62, 32)	0
conv2d_136 (Conv2D)	(None, 60, 60, 64)	18496
max_pooling2d_136 (MaxPooling2D)	(None, 30, 30, 64)	0
flatten_52 (Flatten)	(None, 57600)	0
dense_154 (Dense)	(None, 1024)	58983424
dense_155 (Dense)	(None, 512)	524800
dense_156 (Dense)	(None, 256)	131328
dense_157 (Dense)	(None, 38)	9766
<hr/>		
Total params: 59,672,902		
Trainable params: 59,672,902		
Non-trainable params: 0		

EVOLUTION OF ACCURACY AND LOSS OVER THE EPOCHS

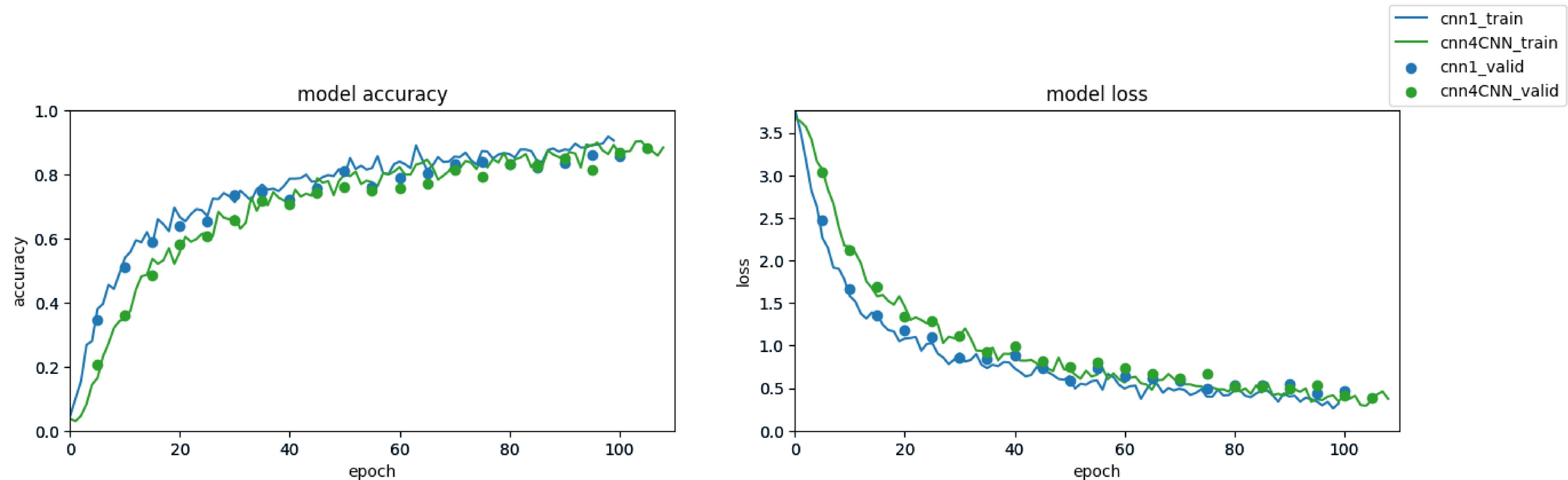


MODEL 3

Model: "sequential_60"

Layer (type)	Output Shape	Param #
<hr/>		
conv2d_160 (Conv2D)	(None, 254, 254, 16)	448
max_pooling2d_160 (MaxPooling2D)	(None, 127, 127, 16)	0
conv2d_161 (Conv2D)	(None, 125, 125, 32)	4640
max_pooling2d_161 (MaxPooling2D)	(None, 62, 62, 32)	0
conv2d_162 (Conv2D)	(None, 60, 60, 64)	18496
max_pooling2d_162 (MaxPooling2D)	(None, 30, 30, 64)	0
conv2d_163 (Conv2D)	(None, 28, 28, 128)	73856
max_pooling2d_163 (MaxPooling2D)	(None, 14, 14, 128)	0
flatten_60 (Flatten)	(None, 25088)	0
dense_179 (Dense)	(None, 512)	12845568
dense_180 (Dense)	(None, 256)	131328
dense_181 (Dense)	(None, 38)	9766
<hr/>		
Total params: 13,084,102		
Trainable params: 13,084,102		
Non-trainable params: 0		

EVOLUTION OF ACCURACY AND LOSS OVER THE EPOCHS



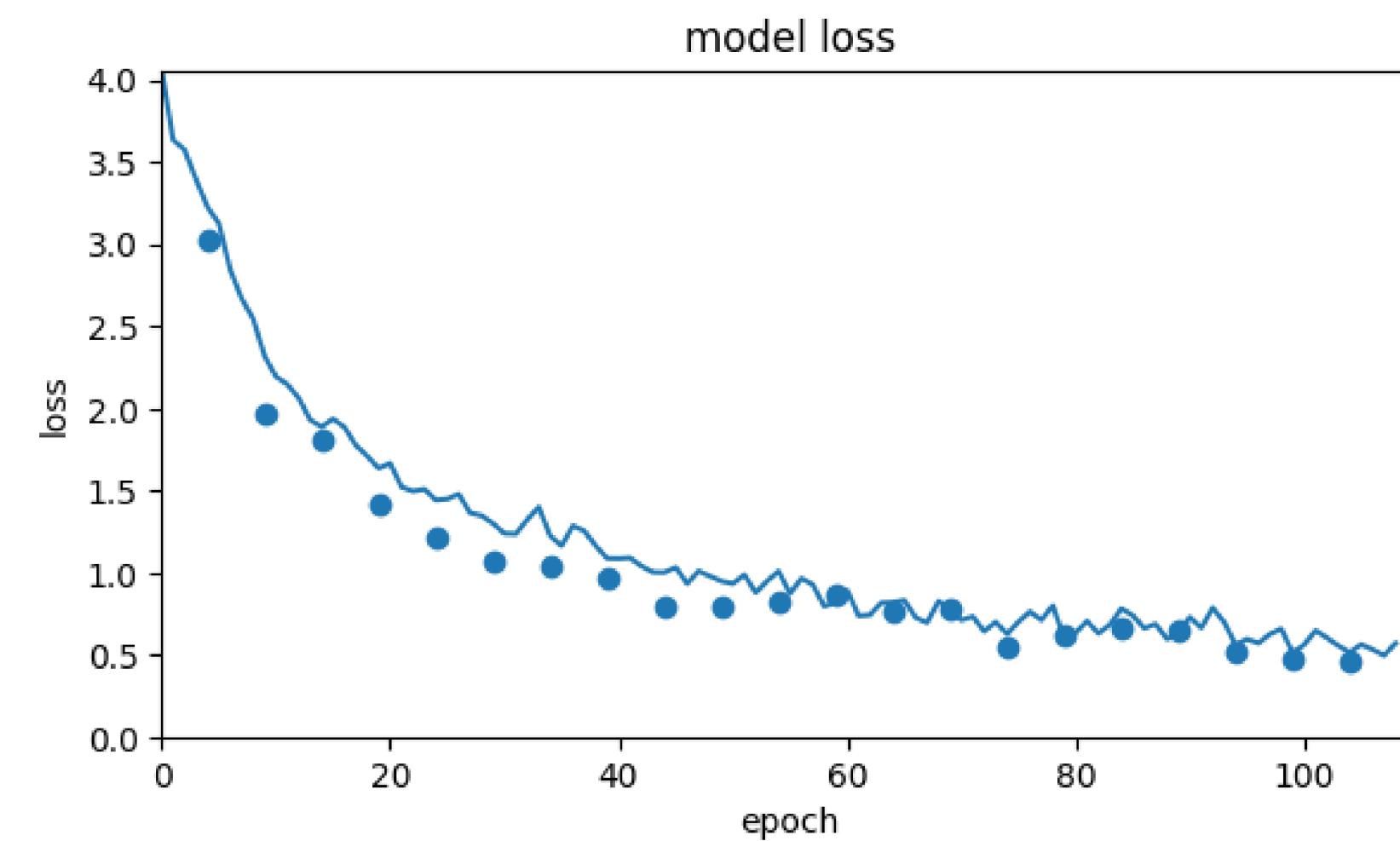
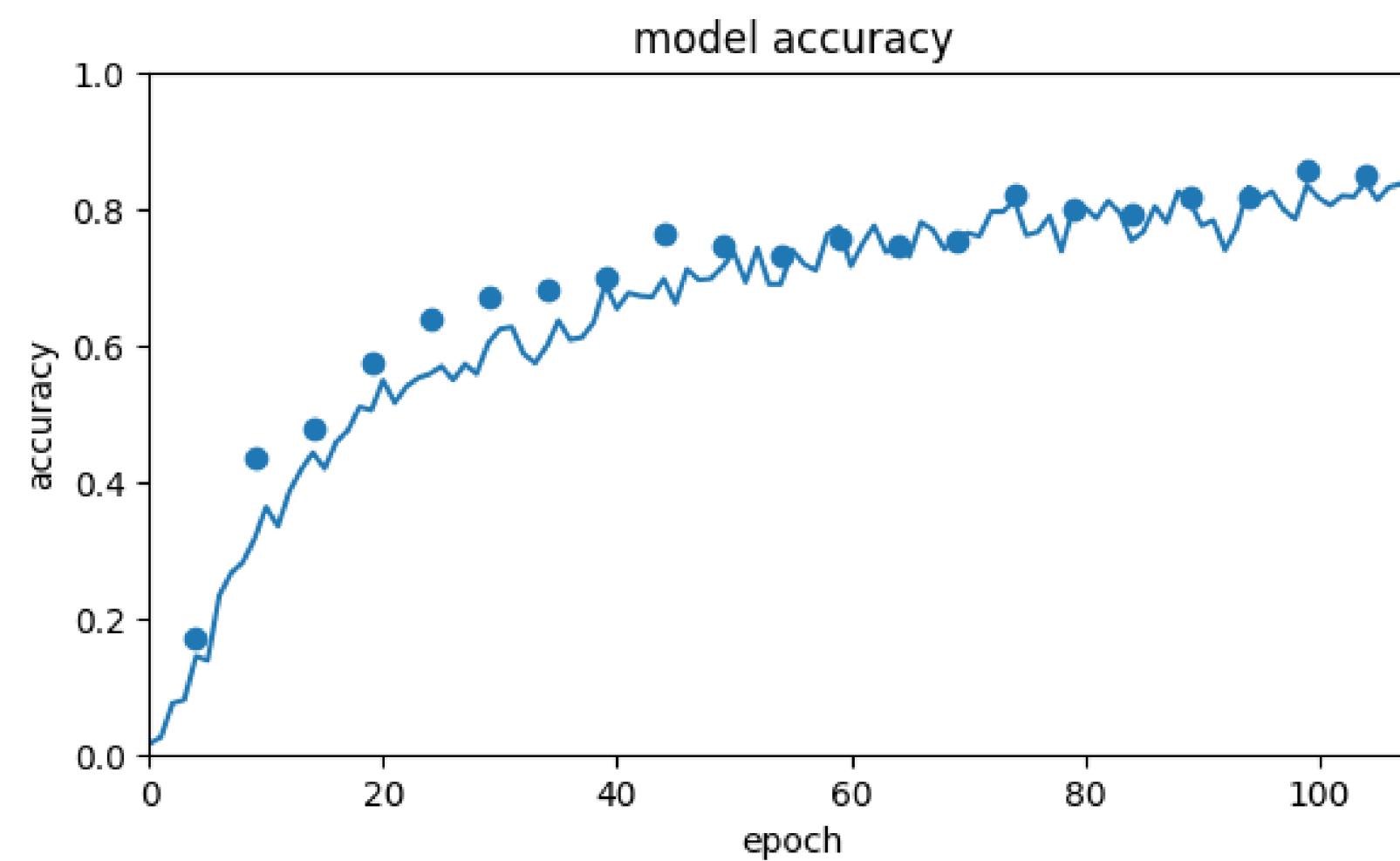
MODEL 4

Model: "sequential_1"

Layer (type)	Output Shape	Param #
<hr/>		
conv2d_3 (Conv2D)	(None, 254, 254, 16)	448
max_pooling2d_3 (MaxPooling 2D)	(None, 127, 127, 16)	0
conv2d_4 (Conv2D)	(None, 125, 125, 32)	4640
max_pooling2d_4 (MaxPooling 2D)	(None, 62, 62, 32)	0
conv2d_5 (Conv2D)	(None, 60, 60, 64)	18496
max_pooling2d_5 (MaxPooling 2D)	(None, 30, 30, 64)	0
flatten_1 (Flatten)	(None, 57600)	0
dense_3 (Dense)	(None, 512)	29491712
dropout_2 (Dropout)	(None, 512)	0
dense_4 (Dense)	(None, 256)	131328
dropout_3 (Dropout)	(None, 256)	0
dense_5 (Dense)	(None, 38)	9766
<hr/>		
Total params: 29,656,390		
Trainable params: 29,656,390		
Non-trainable params: 0		

EVOLUTION OF ACCURACY AND LOSS OVER THE EPOCHS

cnn_dropout_DL_train
cnn_dropout_DL_valid

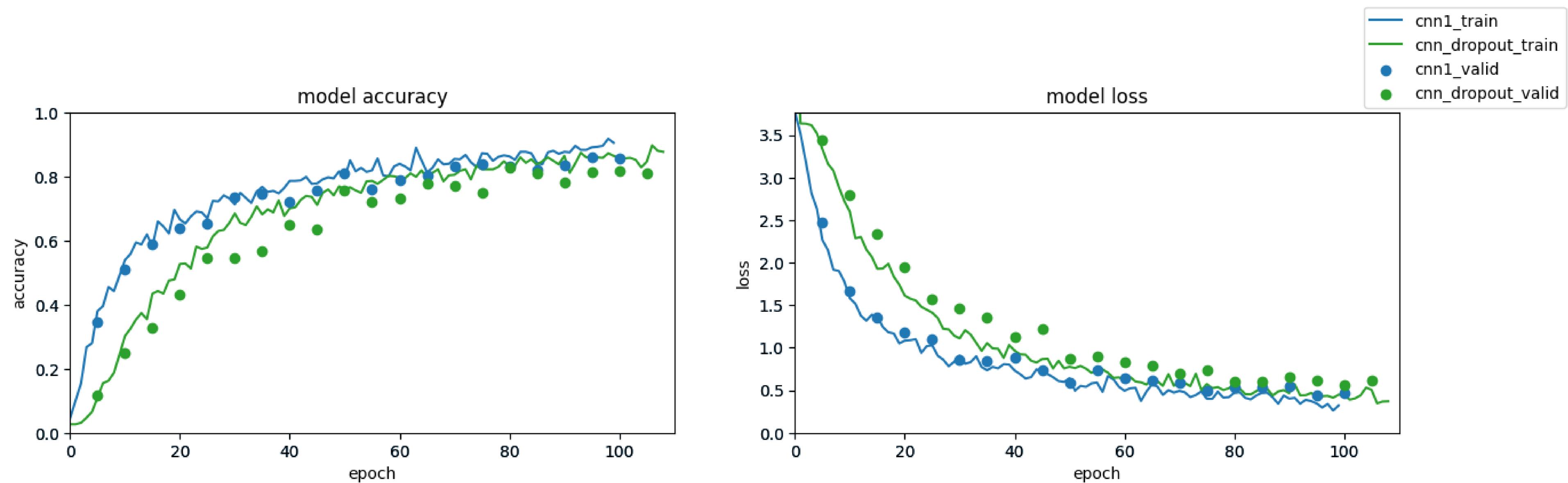


MODEL 5

Model: "sequential_61"

Layer (type)	Output Shape	Param #
=====		
conv2d_164 (Conv2D)	(None, 254, 254, 16)	448
max_pooling2d_164 (MaxPooling2D)	(None, 127, 127, 16)	0
dropout_7 (Dropout)	(None, 127, 127, 16)	0
conv2d_165 (Conv2D)	(None, 125, 125, 32)	4640
max_pooling2d_165 (MaxPooling2D)	(None, 62, 62, 32)	0
dropout_8 (Dropout)	(None, 62, 62, 32)	0
conv2d_166 (Conv2D)	(None, 60, 60, 64)	18496
max_pooling2d_166 (MaxPooling2D)	(None, 30, 30, 64)	0
dropout_9 (Dropout)	(None, 30, 30, 64)	0
flatten_61 (Flatten)	(None, 57600)	0
dense_182 (Dense)	(None, 512)	29491712
dense_183 (Dense)	(None, 256)	131328
dense_184 (Dense)	(None, 38)	9766
=====		
Total params: 29,656,390		
Trainable params: 29,656,390		
Non-trainable params: 0		

EVOLUTION OF ACCURACY AND LOSS OVER THE EPOCHS

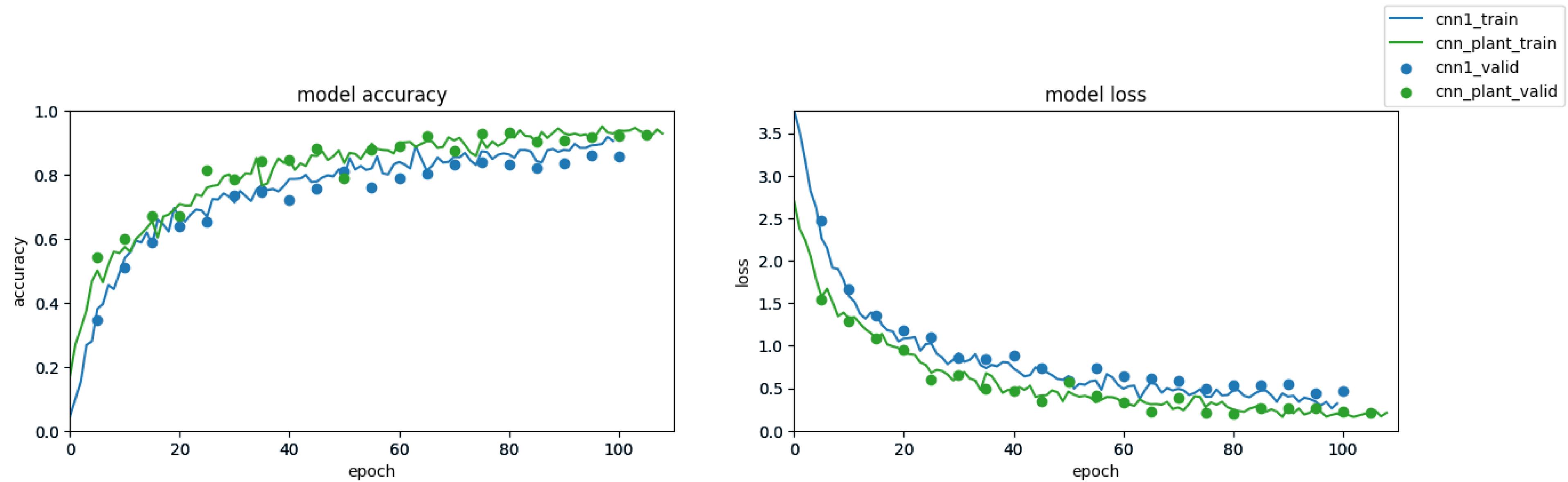


MODEL 6 - PLANTS

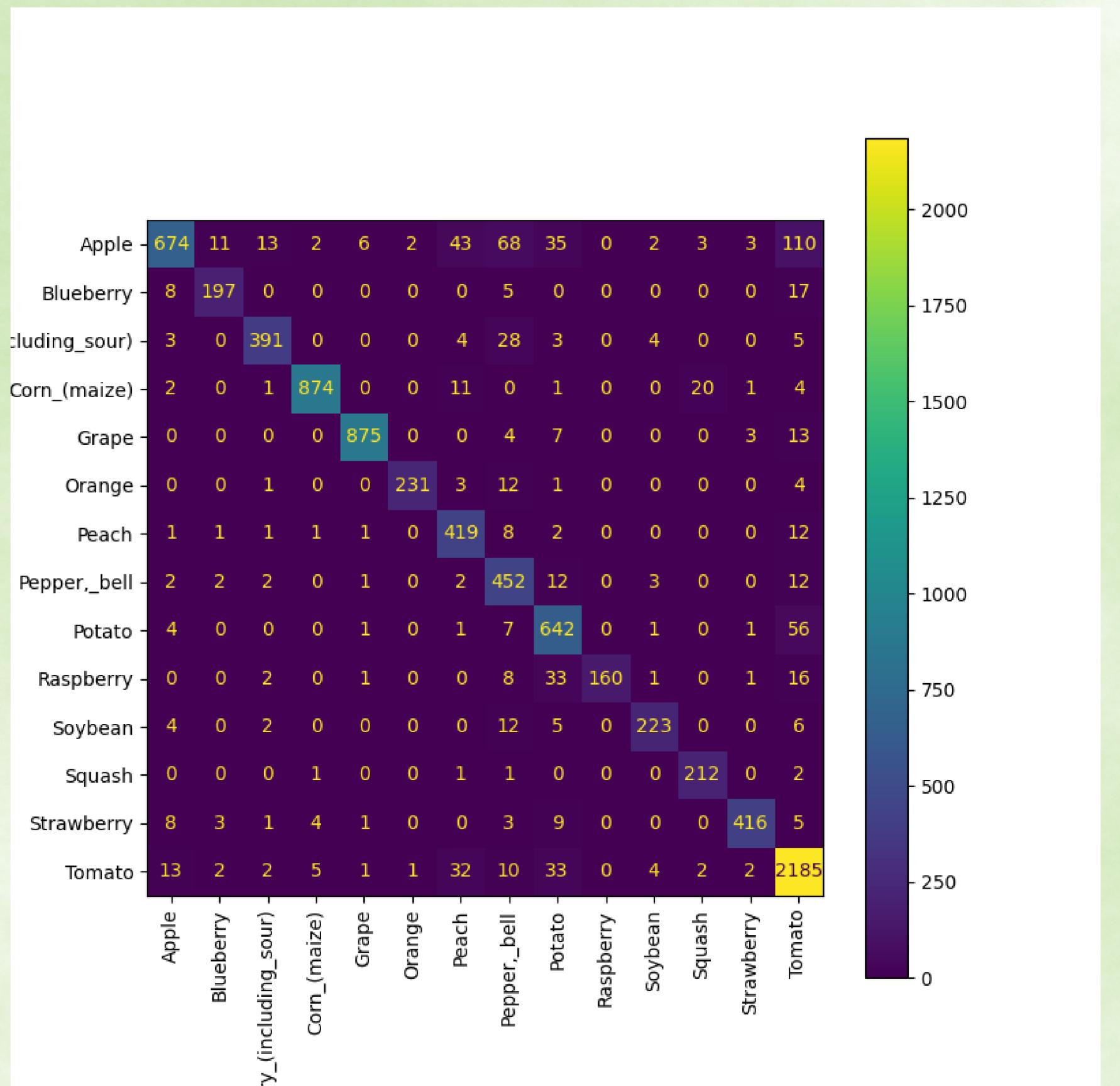
Model: "sequential_3"

Layer (type)	Output Shape	Param #
<hr/>		
conv2d_9 (Conv2D)	(None, 254, 254, 16)	448
max_pooling2d_9 (MaxPooling2D)	(None, 127, 127, 16)	0
conv2d_10 (Conv2D)	(None, 125, 125, 32)	4640
max_pooling2d_10 (MaxPooling2D)	(None, 62, 62, 32)	0
conv2d_11 (Conv2D)	(None, 60, 60, 64)	18496
max_pooling2d_11 (MaxPooling2D)	(None, 30, 30, 64)	0
flatten_3 (Flatten)	(None, 57600)	0
dense_9 (Dense)	(None, 512)	29491712
dense_10 (Dense)	(None, 256)	131328
dense_11 (Dense)	(None, 14)	3598
<hr/>		
Total params: 29,650,222		
Trainable params: 29,650,222		
Non-trainable params: 0		

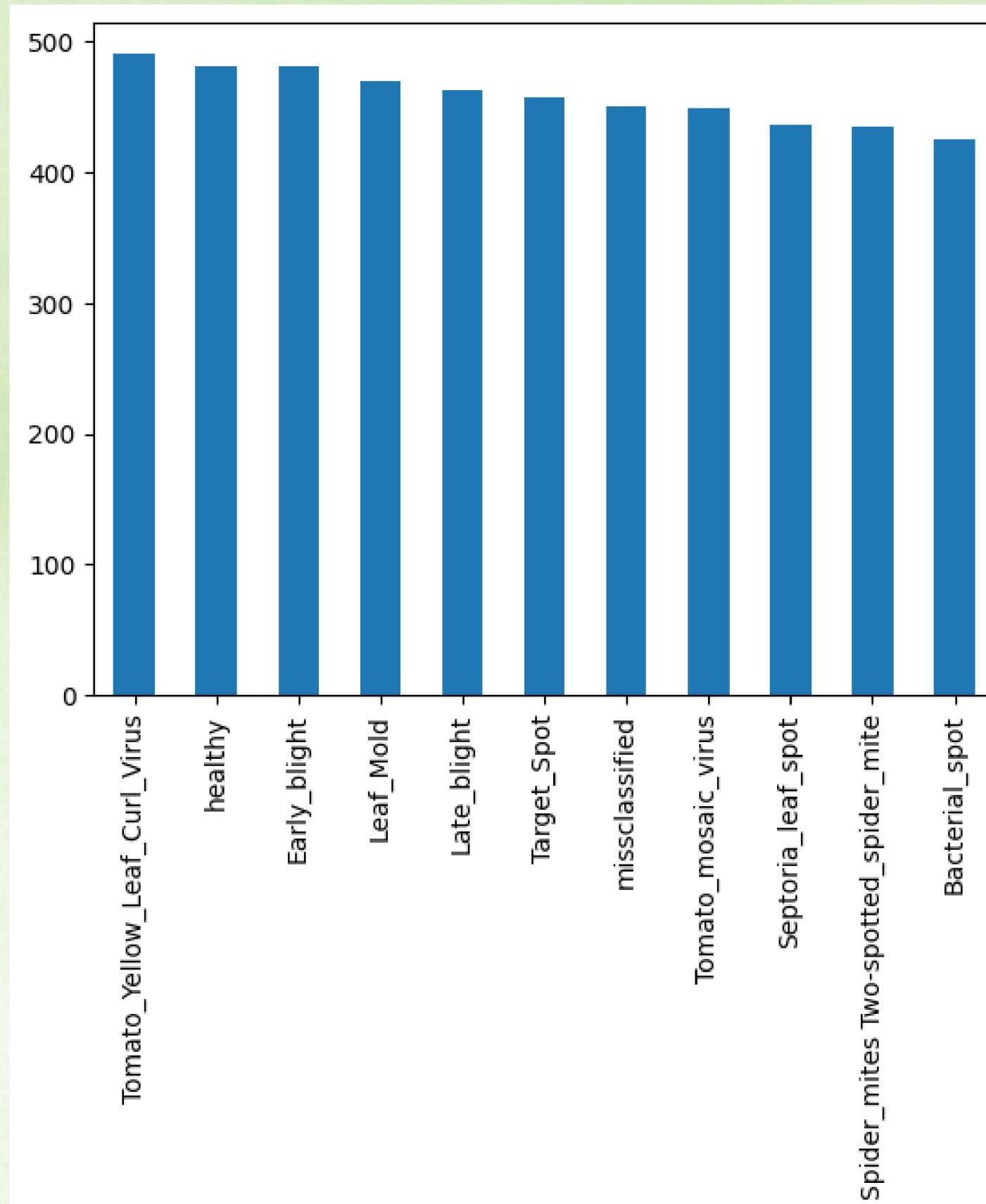
EVOLUTION OF ACCURACY AND LOSS OVER THE EPOCHS



CLASSIFICATION REPORT



MODEL 7 - TOMATOES

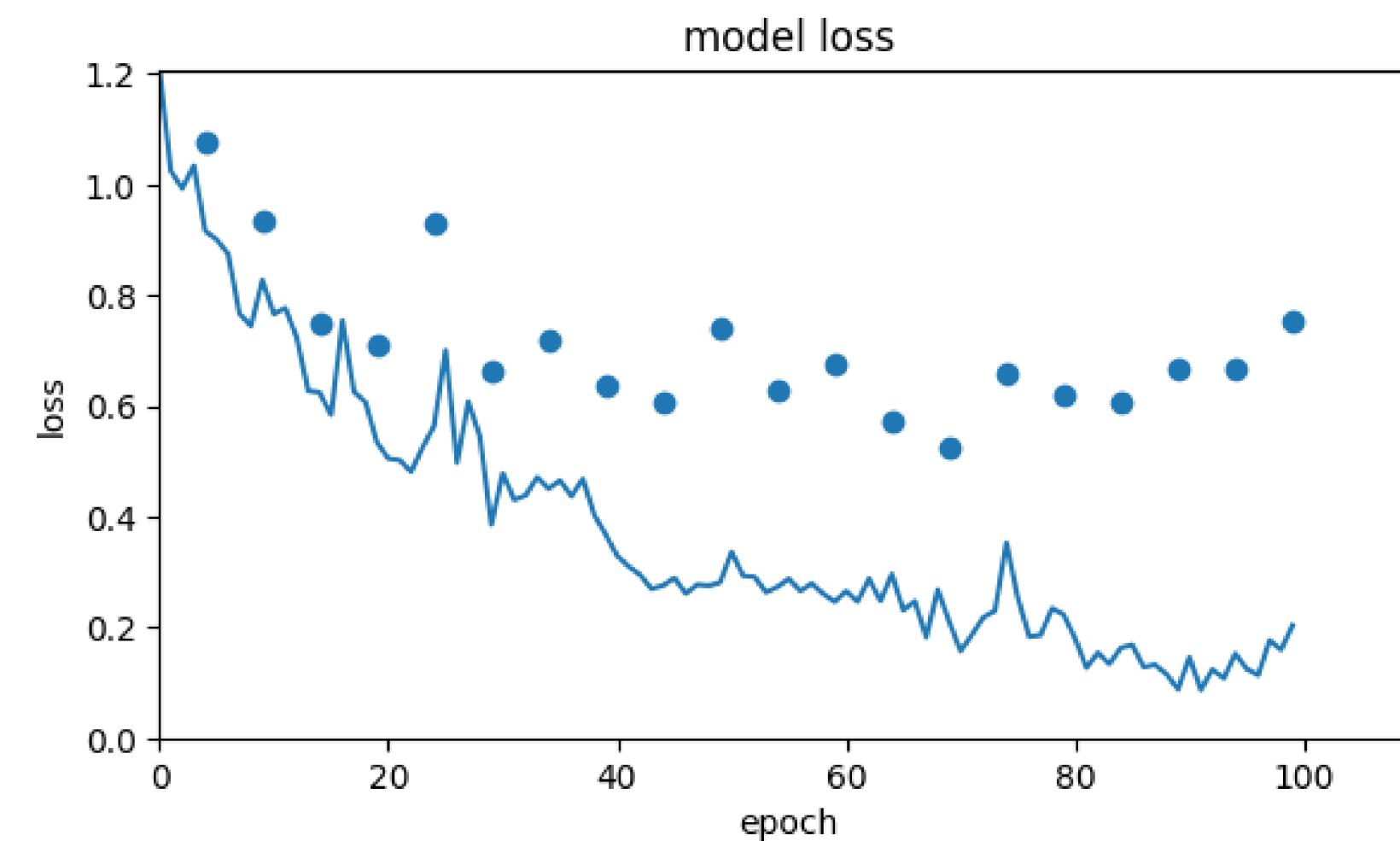
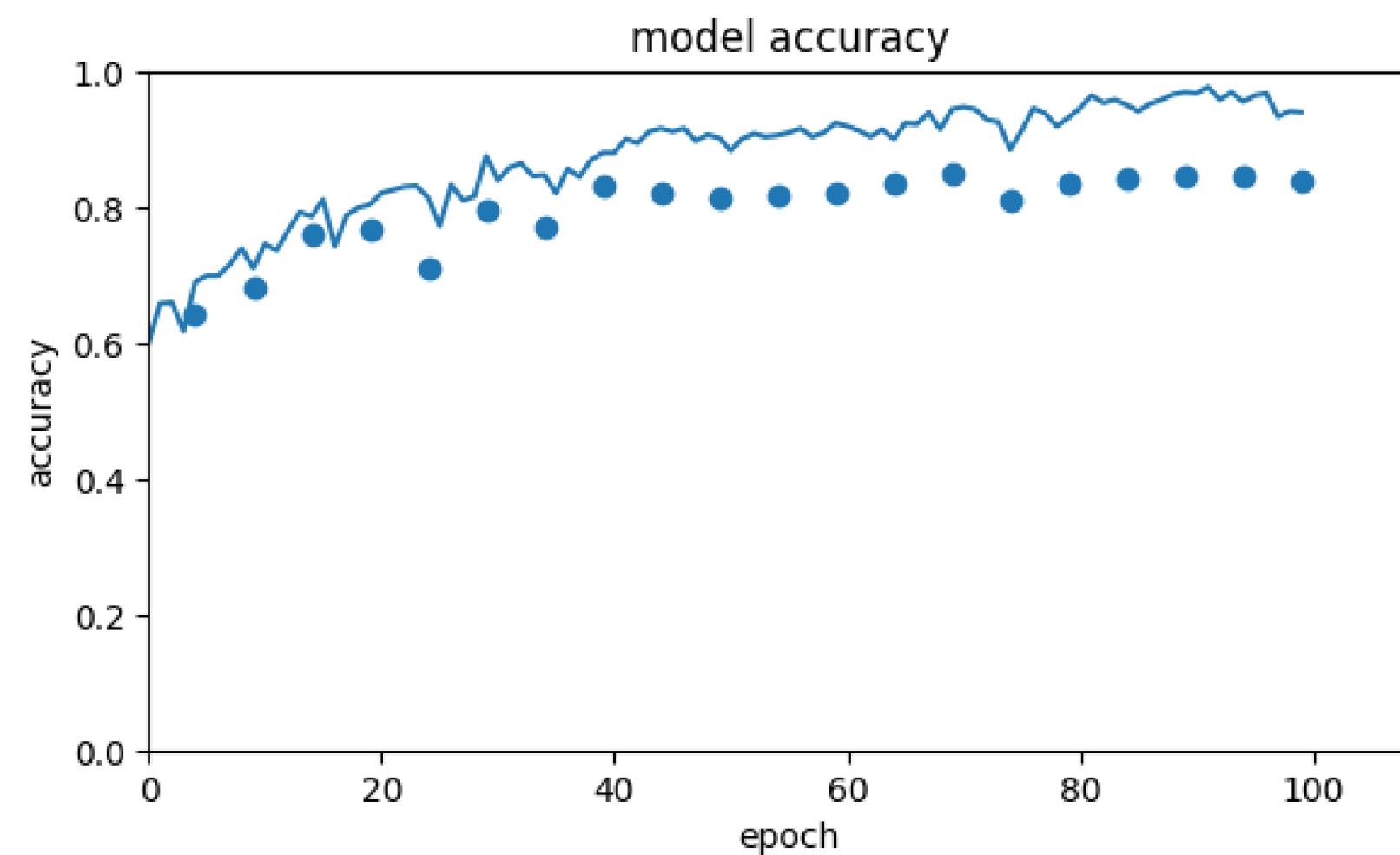


Model: "sequential_4"

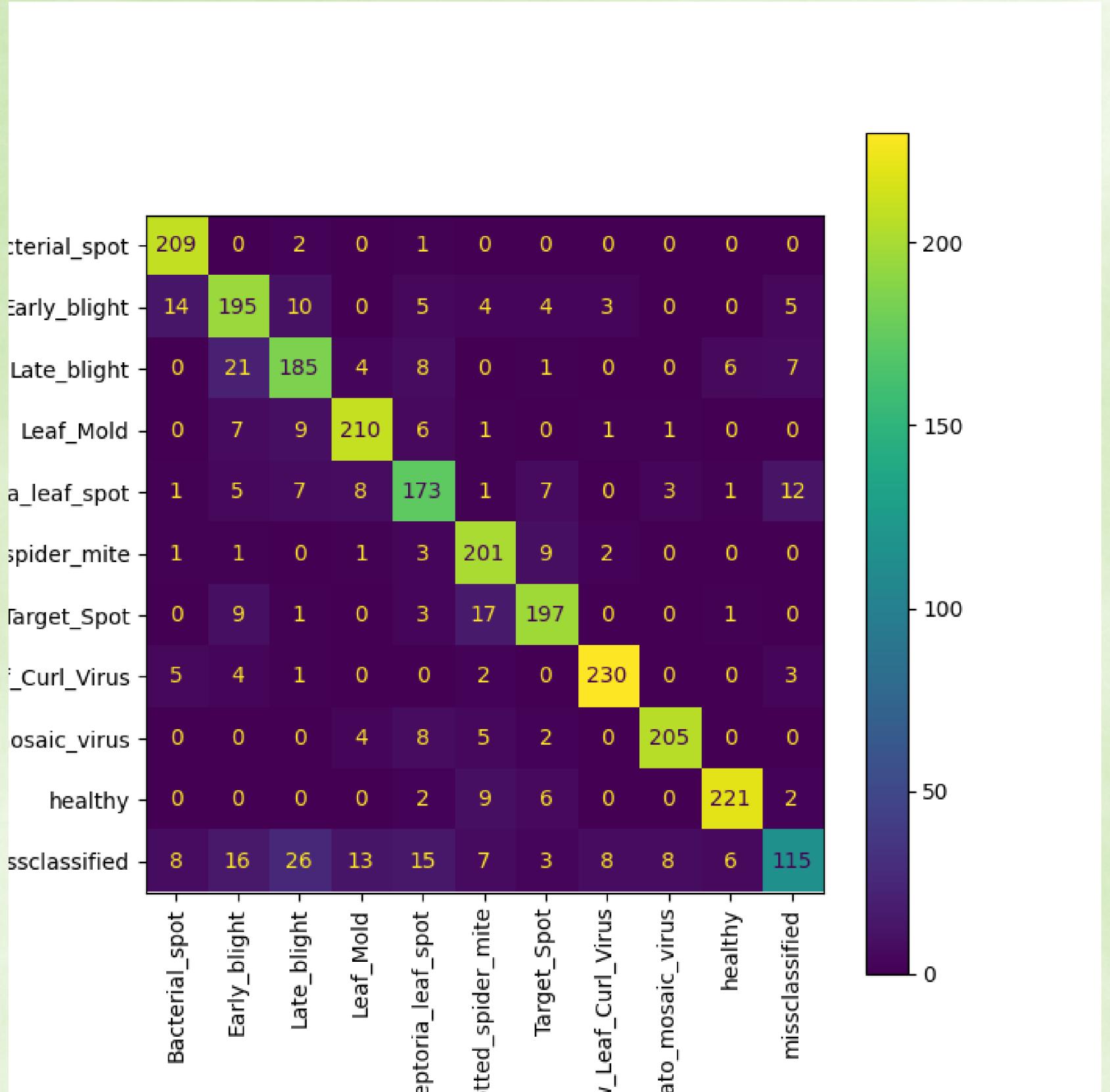
Layer (type)	Output Shape	Param #
<hr/>		
conv2d_12 (Conv2D)	(None, 254, 254, 16)	448
<hr/>		
max_pooling2d_12 (MaxPooling2D)	(None, 127, 127, 16)	0
<hr/>		
conv2d_13 (Conv2D)	(None, 125, 125, 32)	4640
<hr/>		
max_pooling2d_13 (MaxPooling2D)	(None, 62, 62, 32)	0
<hr/>		
conv2d_14 (Conv2D)	(None, 60, 60, 64)	18496
<hr/>		
max_pooling2d_14 (MaxPooling2D)	(None, 30, 30, 64)	0
<hr/>		
flatten_4 (Flatten)	(None, 57600)	0
<hr/>		
dense_12 (Dense)	(None, 512)	29491712
<hr/>		
dense_13 (Dense)	(None, 256)	131328
<hr/>		
dense_14 (Dense)	(None, 11)	2827
<hr/>		
Total params: 29,649,451		
Trainable params: 29,649,451		
Non-trainable params: 0		

EVOLUTION OF ACCURACY AND LOSS OVER THE EPOCHS

cnn_tomato_train
cnn_tomato_valid



CLASSIFICATION REPORT

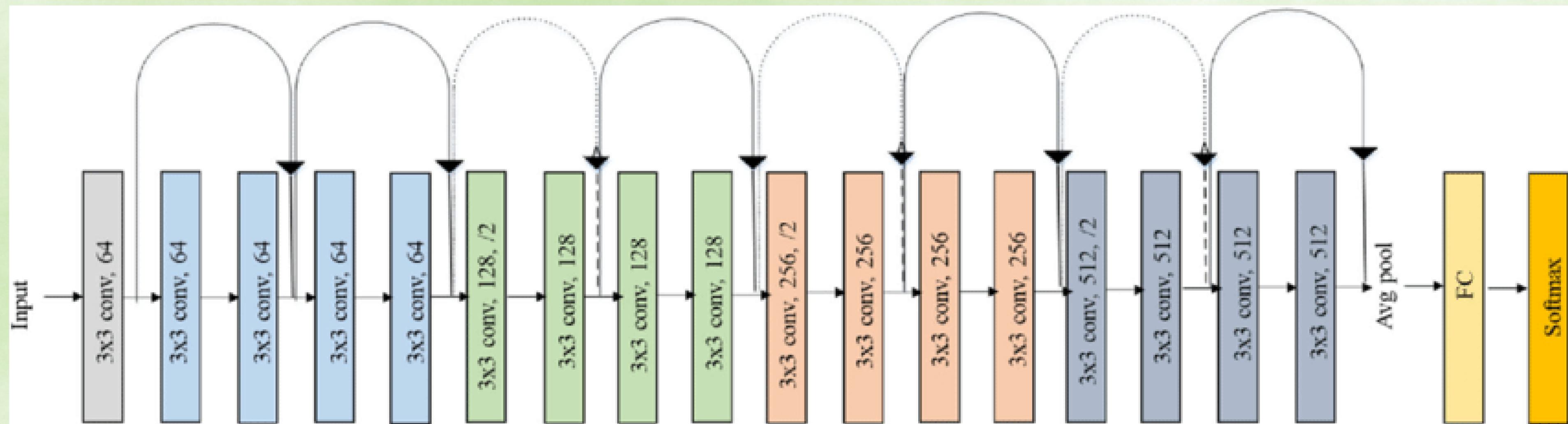


FURTHER IMPROVEMENT

- Try new architectures such as ResNet18.
- Implement image processing algorithms to avoid gray background in photos.
- Try different hyperparameter tuning.
- Use the current neural networks for transfer learning.



ResNet18



FURTHER IMPROVEMENT

- Divide plants into groups:
 - nightshade (Tomato, Potato etc.),
 - grasses (corn, rice, etc.),
 - fruit trees (apple, cherry),
 - cereals, etc.



Cercospora Leaf Spot