Artificial Intelligence
Project

Image Classification

PLANT DISEASES DETECTION

TEAM 50

Rushabh Thakkar (18BIT094)
Shubh Patel (18BIT136D)
Siddharth Oza (18BIT108)
Palak Ashar (18BIT081)
Dev Savsani (18BIT018)
Parthiv Gandhi (17BIT030)

CONTENTS

INTRODUCTION

Plant diseases affect the growth of their respective species, therefore their early identification is very important. Many Machine Learning (ML) models have been employed for the detection and classification of plant diseases but, after the advancements in a subset of ML, that is, Deep Learning (DL), this area of research appears to have great potential in terms of increased accuracy. Many developed/modified DL architectures are implemented along with several visualization techniques to detect and classify the symptoms of plant diseases. Moreover, several performance metrics are used for the evaluation of these architectures/techniques. This review provides a comprehensive explanation of DL models used to visualize various plant diseases. In addition, some research gaps are identified from which to obtain greater transparency for detecting diseases in plants, even before their symptoms appear clearly.

Problem statement

04

The main objective of the project is to develop machine learning-based models to accurately classify a given leaf image from the test dataset to a particular disease category, and to identify an individual disease from multiple disease symptoms on a single leaf image.

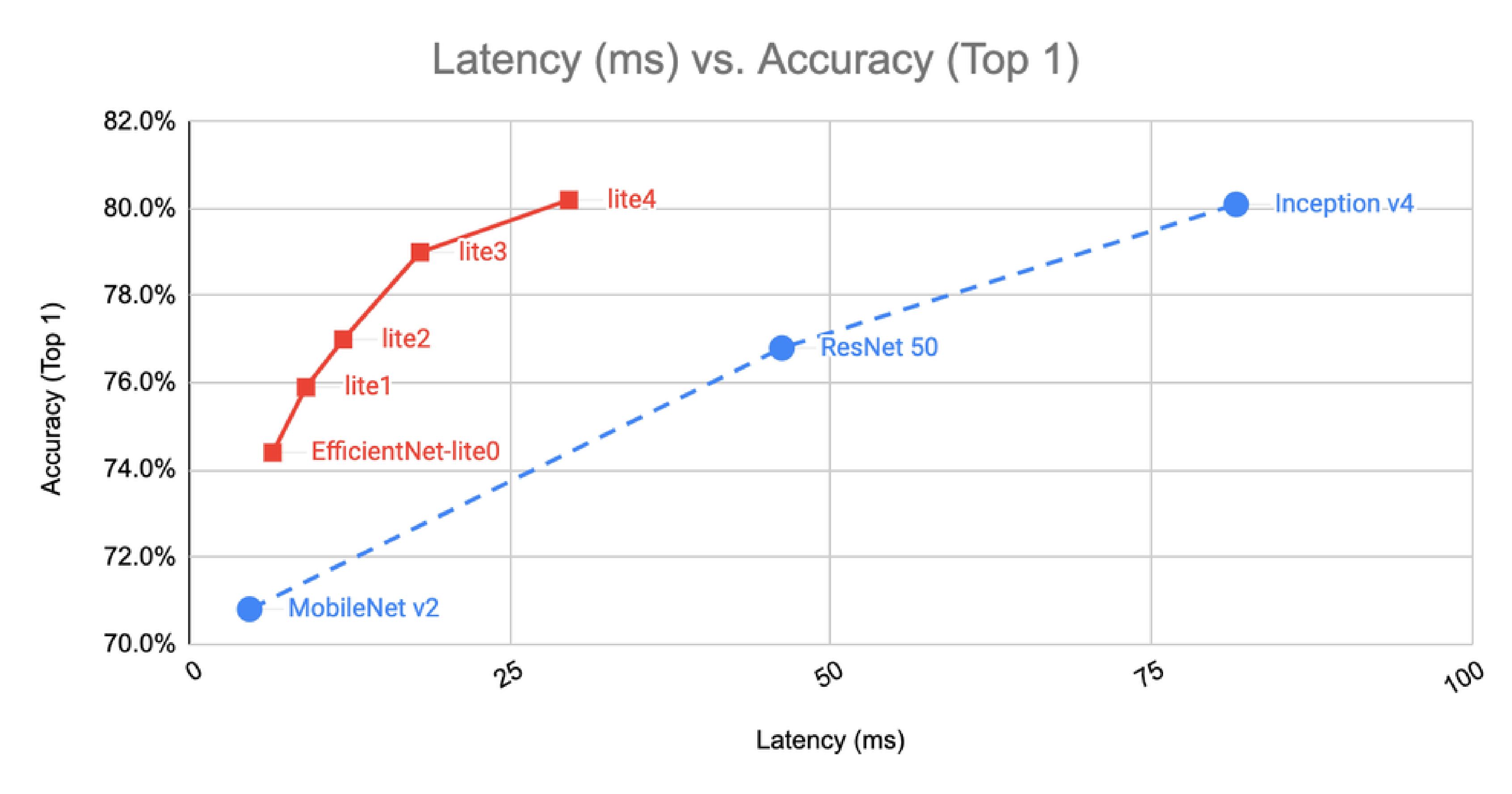
Literature Survey

- TensorFlow Lite(TFL) is the evolution of TensorFlow Mobile(TFM), which already supports deployment on mobile and embedded devices. As there is a trend to incorporate ML in mobile applications and as users have higher expectations on their mobile applications in terms of camera and voice it is highly incentivised to further optimise TFM for lightweight mobile use. Some of the optimisations included in TFL are hardware acceleration through the silicon layer, frameworks such as the Android Neural Network API and mobile-optimised ANNs such as MobileNets and SqueezeNet. TF-trained models are converted to the TFL model format automatically by TF.

	float32	int8	Improvement
model size	17.7MB	5.17MB	3.42x
accuracy (top1)	75.1%	74.4%	-0.7 percent point
latency (CPU)	12ms	6.5ms	1.85x

Literature Survey

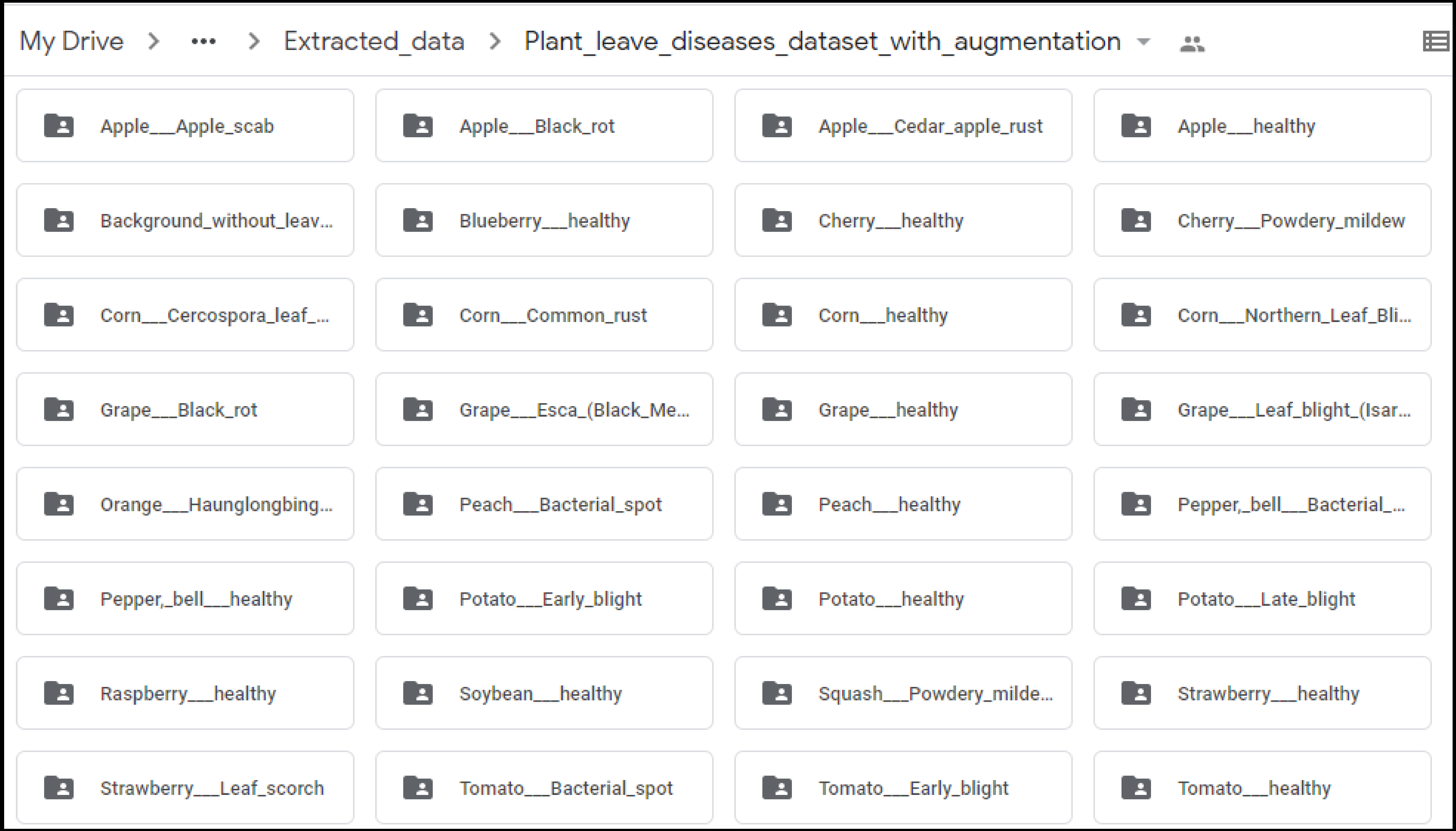
- Google released a family of image classification models called EfficientNet, which achieved state-of-the-art accuracy with an order of magnitude of fewer computations and parameters.EfficientNet-Lite brings the power of EfficientNet to edge devices and comes in five variants, allowing users to choose from the low latency/model size option (EfficientNet-Lite0) to the high accuracy option (EfficientNet-Lite4). The largest variant, integer-only quantized EfficientNet-Lite4, achieves 80.4% ImageNet top-1 accuracy, while still running in real-time (e.g. 30ms/image) on a Pixel 4 CPU. Below is how the quantized EfficientNet-Lite models perform compared to similarly quantized version of some popular image classification models.



DATASET

We analyze 39000 images of plant leaves, which have a spread of 39 class labels assigned to them. Each class label is a crop-disease pair, and we make an attempt to predict the crop-disease pair given just the image of the plant leaf. In all the approaches described in this paper, we resize the images, and we perform both the model optimization and predictions on these downscaled images. Across all our experiments, we use three different versions of the whole Plant Disease dataset.

- There are 39 data classes.
- We got the augmented data with 1000 images each.
- We have a total of 39000 images.
- We have 14 types of plant leaves which also includes healthy images.


My Drive > ••• > Extracted_data > Plant_leave_diseases_dataset_with_augmentation
Apple___Apple_scab
Apple___Black_rot
Apple___Cedar_apple_rust
Apple___healthy
Background_without_leav...
Blueberry___healthy
Cherry___healthy
Cherry___Powdery_mildew
Corn___Cercospora_leaf_...
Corn___Common_rust
Corn___healthy
Corn___Northern_Leaf_Bli...
Grape___Black_rot
Grape___Esca_(Black_Me...
Grape___healthy
Grape___Leaf_blight_(Isar...
Orange___Haunglongbing...
Peach___Bacterial_spot
Peach___healthy
Pepper,_bell___Bacterial_...
Pepper,_bell___healthy
Potato___Early_blight
Potato___healthy
Potato___Late_blight
Raspberry___healthy
Soybean___healthy
Squash___Powdery_milde...
Strawberry___healthy
Strawberry___Leaf_scorch
Tomato___Bacterial_spot
Tomato___Early_blight
Tomato___healthy

Proposed Methodology

INSTALLING REQUIRED LIBRARIES

Installing tflite-model-maker

MAKE THE NECESSARY IMPORTS:

numpy, tensorflow, image_classifier, tflite_model_maker

UNZIPPING DATA

Unzipping the data and saving the data in drive

LOADING DATA

We will load the data using DataLoader

SPLITTING THE DATASET

Spliting the dataset into training and testing sets! Let’s keep the test set 20% of everything using data.split()

Proposed Methodology

OBSERVE THE SHAPE OF THE TRAINING AND TESTING DATASETS

INITIALIZE AN IMAGE_CLASSIFIER

creates an image classifier using a keras.Sequential model, and loads data using preprocessing.image_dataset_from_directory

FIT/TRAIN THE MODEL.

CALCULATING THE ACCURACY

We will get two accuracy, training data accuracy and testing data accuracy.

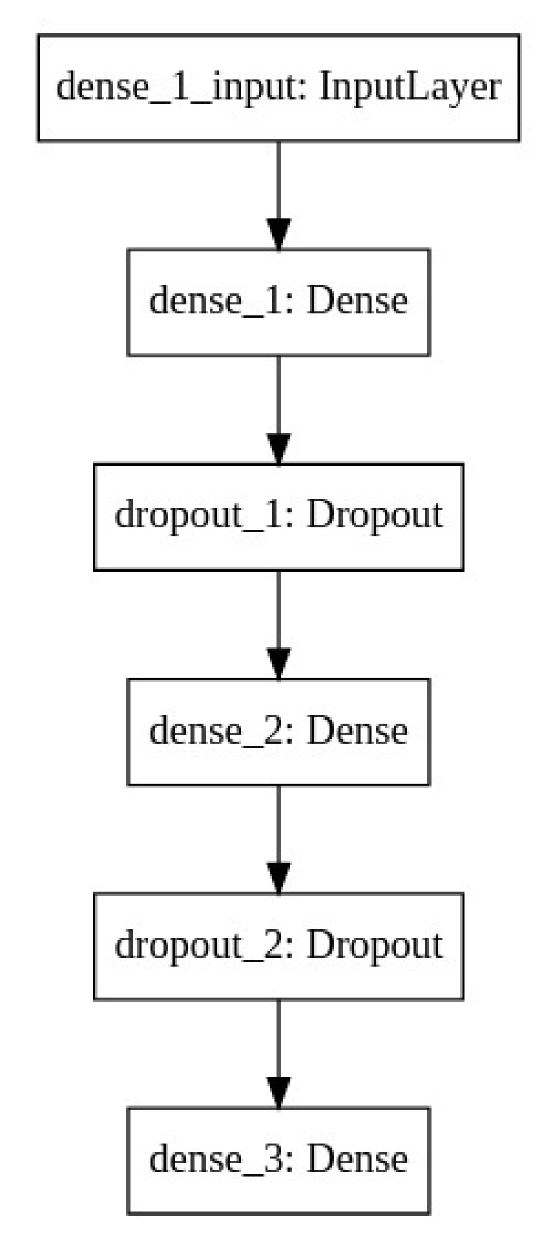
EXPORTING FILES

Exporting tfmodel.lite and label.txt files after successfully training the model

Model Description

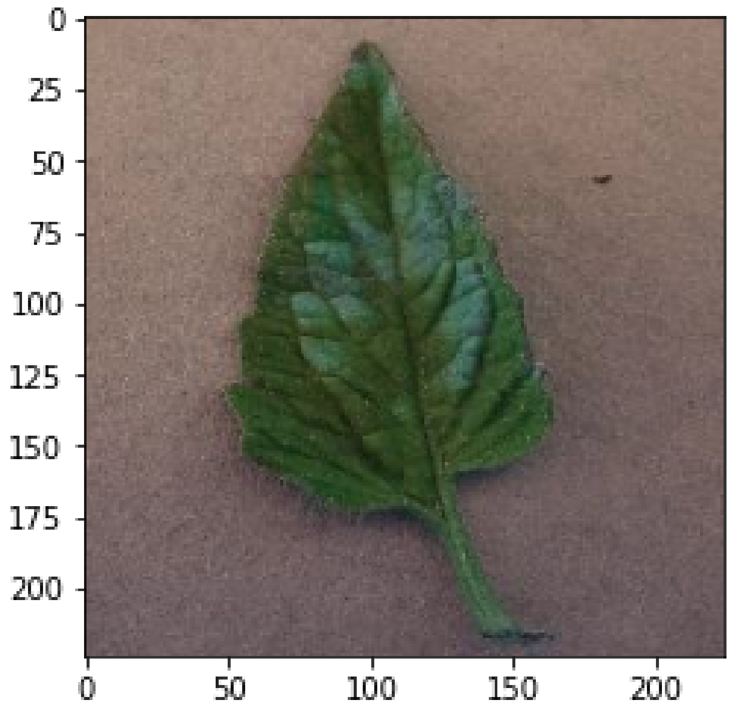
We are using tf.keras.Sequential model which is based on Transfer Learning.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
hub_keras_layer_v1v2 (HubKer (None, 1280)              3413024
_________________________________________________________________
dropout (Dropout)            (None, 1280)              0
_________________________________________________________________
dense (Dense)                (None, 39)                49959
=================================================================
Total params: 3,462,983
Trainable params: 49,959
Non-trainable params: 3,413,024
```

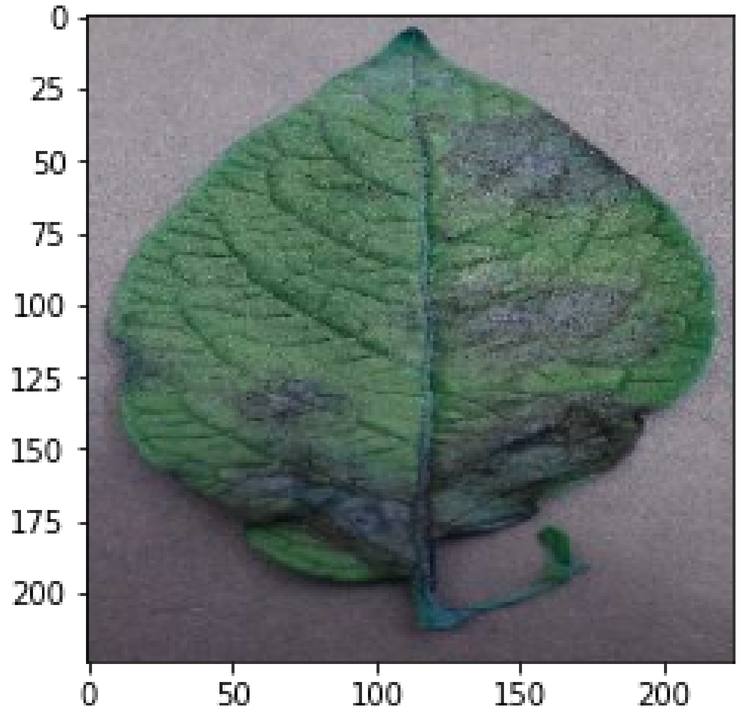


Output Samples

SOURCE: class: Tomato___Leaf_Mold, file: Tomato___Leaf_Mold/22cb45f2-6368-4e94-8deb-939d1f6b85ca___Crnl_L.Mold 7084.JPG
PREDICTED: class: Tomato___Leaf_Mold, confidence: 0.998060



SOURCE: class: Potato___Late_blight, file: Potato___Late_blight/2d736aa6-79a6-42b0-9e92-d859bcd72824___RS_LB 5244.JPG
PREDICTED: class: Potato___Late_blight, confidence: 0.999595



Accuracy

Our accuracy on training data:- 95.45%

Our accuracy on Testing data:- 96.39%

Project Application

- Plant classification
- Plant disease detection
- Suggest the action needed to be taken
- Ease of maintaining plants for beginners and Plant health monitoring

Future Advancement

The sub-area itself has a wide scope of future research to automate the process of plant disease recognition. Some of the future aspects are following which can help the researchers working in the same field of research

- The mobile based application can be designed by using an automate plant disease identification and classification techniques that will help field farmers for providing instant solutions.
- Plant disease prediction and forecasting are also the area of related research.
- The research area is not limited to detection and recognition of the disease from which the plant is suffering but also detects the various stages of the particular disease and can provide various solutions to the problem at that very stage.
- These techniques can also help in designing plant disease recognition based on remote sensing images.

References

- https://ieeexplore.ieee.org/abstract/document/6169716
- https://www.sciencedirect.com/science/article/pii/S2214317320300196
- https://www.mdpi.com/2223-7747/8/11/468
- https://apsjournals.apsnet.org/doi/full/10.1094/PDIS-03-15-0340-FE
- https://www.frontiersin.org/articles/10.3389/fpls.2016.01419/full

Github Link

https://github.com/rushabh1605/Plant-Disease-Classification

THANK YOU