Proceedings of the Second International Conference on Electronics and Sustainable Communication Systems (ICESC-2021)
IEEE Xplore Part Number: CFP21V66-ART; ISBN: 978-1-6654-2867-5

Leaf Disease Detection Using Deep Learning

Teenu Sahasra M
4th Year B.Tech, Department of IT
VR Siddhartha Engineering College
Vijayawada, India
teenusahasra@gmail.com

Sai Kumari S
4th Year B.Tech, Department of IT
VR Siddhartha Engineering College
Vijayawada, India

Sai Meghana S
4th Year B.Tech, Department of IT
VR Siddhartha Engineering College
Vijayawada, India
meghana301199@gmail.com

Mrs. P. Rama Devi
Professor, Department of IT'
VR Siddhartha Engineering College
Vijayawada, India
rama.polagani99 @gmail.com

2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC) | 978-1-6654-2867-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICESC51422.2021.9532909

Abstract— Economy contributes the most for the productivity of the agriculture. In agricultural field, the disease in plants is more common and the detection of disease in plants has become more feasible due to the above reason. These days's plant disease detection has acquired enlarging scrutiny in surveilling crops of large and various fields. Farmers undergo significant hassles in chop and changing from one disease administer principle to a different one. We can identify or spotting the tomato leaf diseases for detection for surveillance and monitoring experts is the standard approach for detection. The plants get seriously affected if the proper control hasn't been taken and this represents the quality of the pants the production of the plants will be affected. Detection of disease through some mechanized technique and methodology is efficient and constructive because it decreases an outsized toil of surveilling in the large cultivation. In the premature phase we can detect the symptoms of the plant diseases since their first appearance on their leaves of the plants. By using this paper we can identify the algorithm which is used for image segmentation and for automated classification used for the detection of diseases of leaves in the plants. It also covers distinct disease classification methods of working which is used for the detection of diseases in plants.

Keywords—CNN; VGG16; Deep learning; Pre-processing; Keras; Tensorflow; Opencv

I. INTRODUCTION (*PROBLEM STATEMENT)*

Now a day's tomato has become the more welcomed crop around the globe and it is often found in the kitchen, using it in various forms, ignoring the type of the cuisine. Following potato, tomato which is cultivated world widely. India stands in the 2nd position in the fabrication of tomatoes [2]. In tomato, different forms of diseases come with the quality of the farm production. So, to detect the disease we used a DL based approach. For disease recognition and classifying, we are using a CNN based approach. In this pre-trained model, the CNN consists of convolution layers and max-pooling layers and the later one is the one which is 2 fully connected layers. The trial and error result has shown the [11] efficiency of the proposed model than VGG16 which is the pre-trained model. The accuracy of the classification differs from 75% to 100% concerning classes and the proposed model's mean accuracy is 90.2% for 9 types of diseases and 1 healthy class.

Agriculture has been one of the foremost predominant fields of the Indian Economy. In the agricultural field, a serious role is being played by the disease detection in plants. The growth of the plants is affected [14] because of the presence of the diseases in plants. With the help of disease detection methodology and techniques the plant disease can be detected at premature phases. In various roles of the plant such as leaves, roots, etc and detects the traits of leaf diseases that are observed. It uses images of the leaves for monotonous job which can make the [3] detection and classification method of the disease using images of the leaves automatic.

II. BASIC TERMINOLOGY

II-A) DEEP LEARNING

Deep learning is used convert the human brain into process the information and creating different design in artificial intelligence (AI) that is used for making conclusion. It is a type of machine learning footing artificial neural networks during which multiple layers of transform are won't to extract more and more high-level features from data.

II-B) VGG16

VGG16 is used in numerous deep learning image classification drawbacks; however, smaller network architectures are often more worthwhile (such as SqueezeNet, GoogLeNet, etc.). But it's a superb building block for learning purposes because it's straight forward to implement.

II-C) KERAS

Keras is an API library it is designed for humans. It is deep learning framework. Keras used to run on it.

II-D) TENSORFLOW

Tensorflow is a deep learning framework. It is an end to end open source platform used to train ML models. It mainly focuses on the related version that is suitable for running the program. It is used in large scale for ML, DL projects in our day to day life.

III. LITERATURE REVIEW

We have referred to some of the latest conference papers which are related to our project on change detection and also observed some of the methods as well as procedures which are implemented and written those observations of each paper that will be useful in development of our project.

We have number of papers published for leaf disease detection and this is the Journal Selected Topics in [1] Applied Plant Disease Detection (2020) and the methods proposed are Faster RCNN, K-Means, and VGG16.We have different proposed methods from different number of papers some of the uses CNN, Support Vector Machine(SVM), VGGNet and other pre-trained models likes AlexNet, DenseNet, VGG19 etc. Disease in the plant can be identified in root, stem, leaf, fruit etc. We have number of paper and journals for leaf disease detection for tomato leafs, apple leafs, potato leafs and soon. [5] They use different neural networks such as CNN, ANN etc in the leaf disease detection. In [3] Detection of plant disease we have use different techniques like image segmentation, soft computing and confusion metrics.

By using confusion metrics we can find the accuracy of the model and other variables like F1 score, Precision and recall. Segmentation and soft computing are the techniques used for the feature extraction [7] and detection in the images. These CNN pre-trained models are used detect whether the leaf is healthy or diseased. If the leaf is healthy it tell us that it is healthy or tells us the leaf is diseased and tells us the name of the disease and required remedies for it. [6] VGG16 is a pre-trained model of Convolution Neural Network it tells us about better performance and accuracy of the model depending on it features in the model. [4] After the detection of the disease we can also apply different techniques on it.

Information collected from the different paper we mainly applied these on tomato, apple, potato, cherry, and corn leafs and frequently used model are [2] CNN pre-trained models.

Our Work:

Leaf disease detection is used for detection of disease in the leaves by using CNN pre-trained models and also improves the accuracy of the model. We used tomato leafs from the “Plant Village Dataset” for this project we also have different diseases in them like Target spot, Early Blight etc. By using CNN pre-trained model (VGG16) we can better accuracy and performance of the model. We have both advantages and disadvantages for using this model like have more parameters and model is somewhat expensive.VGG16 used for plant disease detection had 98.2% high accuracy and faster detection.

IV. PROPOSED WORK

IV-A)PROJECT DESIGNMETHODOLOGY:

Use case Diagram:

In Unified Modelling Language (UML), a use case diagram to build one, you'll use a group of specialized symbols and connectors. In figure-1.1, when we give a new input image first the module extracts the leaf features. Then it goes through the CNN model. It then contrasts the attributes with an already trained dataset. Then it goes through dense CNN and therefore the leaf attributes are extricated independently. Then the module will predict whether the plant leaf healthy or diseased.

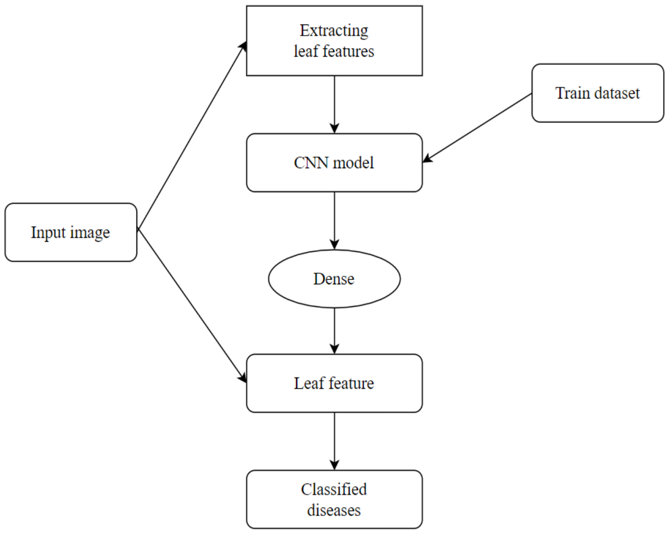


Figure-1.1: Use case diagram for leaf disease detection

Class Diagram:

The class diagram is the building block of object-oriented modeling used for conceptual modeling of the structure of an appeal, and modeling explain the model into a code. Class diagrams also be used for data whittle. As we see the normalization class comprises a raw image and it is fed to the CNN model which comprises dense and weight. The CNN model systematizes and distinguishes by using the training model. Leaf detection gets used to the features.

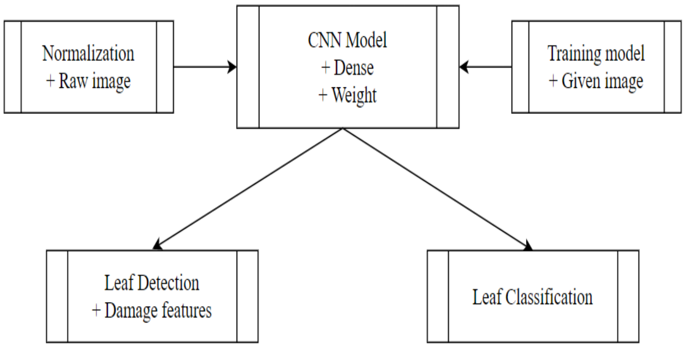


Figure-1.2: Class diagram for leaf disease detection

Activity Diagram:

The activity diagram is a behavioral diagram, it portray the behavior of a system in UML diagram. An activity diagram is similar to a flow chart that whittles the flow of one activity to another activity. Activity Diagrams relate how activities are interlinked to provide service that can be at conflicting levels of ancestry. It is also appropriate for modeling. Collection of use cases coordinate to constitute workflows.

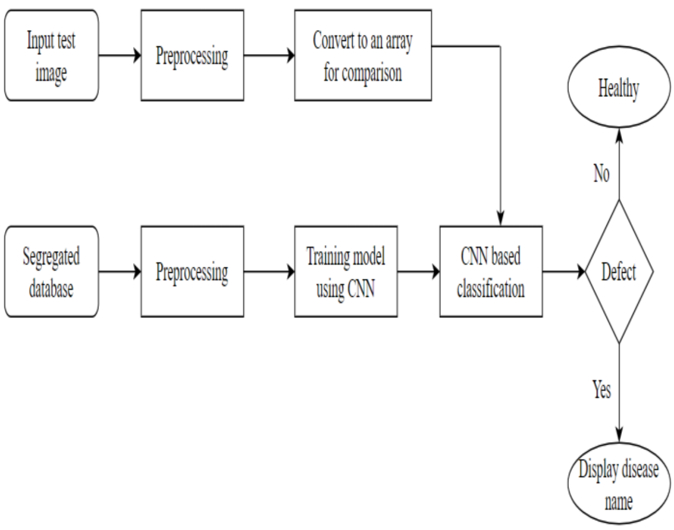


Figure-1.3: Activity diagram for leaf disease detection

IV-B) ARCHITECTURE DIAGRAM

• Download the leaf disease dataset from the Kaggle website. Load the images from the dataset.
• After loading the dataset we have to Pre-processing the plant's images.
• Then we have to extract various features like color, shape, and size from the leaf images.
• We have to train the leaf dataset and process them by using RGB leaf images
• CNN algorithm is used for leaf disease detection. In that, we use the model which is trained as VGG16 for classifying.
• The VGG16 model finds the accuracy and loss of it by training and validating the model.

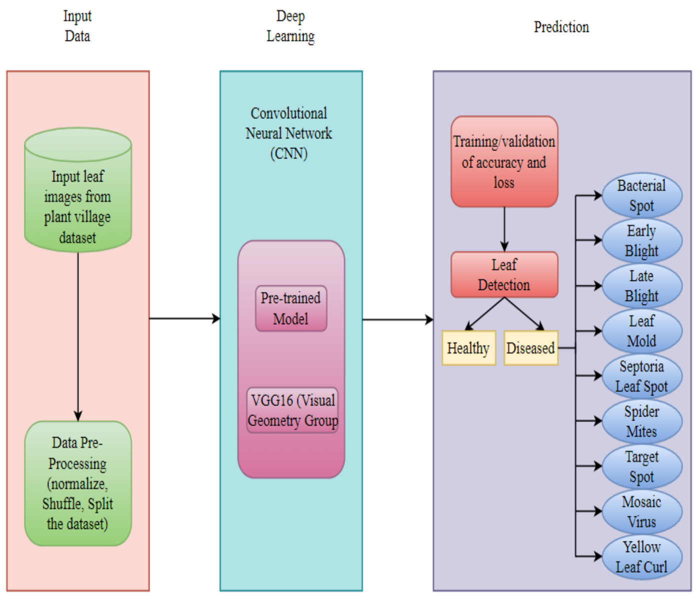


Figure 2: Leaf Disease Detection Architecture

IV-C) OVERVIEW OF ALGORITHM USED

CNN Algorithm:

A Convolutional Neural Network (CNN) is also a Deep Learning algorithmic rule and functionality that can take image as an input. We can extract the information from an input image with the help of the multiple layers that are hidden in the convolution neural network. There are four important layers in CNN. They are Convolution layer (first layer), ReLU layer (middle layer), Pooling layer (middle layer), Fully connected layer (last layer). [8] The pixels that are present in an image are recognized by CNN and that pixels of an image are fed to the convolutional layer, then the convolutional layer will perform some convolution operations then the following result will be the convolution map. The rectified feature map will be obtained when the convoluted map is given to the ReLU layer. The image should be and is processed with number of convolutions and ReLU layers [13] for locating the features. Now to identify specific and particular parts of an image we use different layers of pooling with various filters. Now the featured map will be flattened and is fed to the last layer, a fully connected layer to get the final output.

VGG16:

In VGG16, it consists of 16 layers.

The input to VGG is based on ImageNet for RGB images. Preprocessing layer takes the RGB image with pixel values in the range of 0–255 and subtracts the mean image values which are calculated over the entire ImageNet training set.

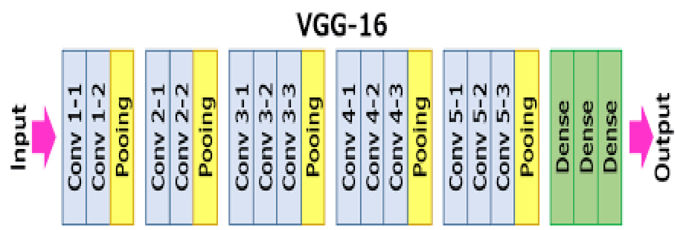


Figure-3: VGG16 Representation

In VGG16, it consists of 16 layers. VGG16 uses ImageNet for the classification of the images. Convolution layers of 3 filters, maxpool layer of 2 filters. In the end it has 2 Fully connected layers (2FC) and last layer is used as softmax for output.

Convolutional Layer:
In a convolutional layer, a neuron is associated with the specific extent of input neurons alternatively full-connection in which the narrowing of the parameters that are to be learned is done. As a result the growth of the network will be deeper without any difficult parameters [9].

Pooling Layer:
These layers wont to minimize the dimension of the feature maps. By reducing the parameters that are to be found and used for the performed computation within the network. [12] The features present during a region of the feature map can be summarized by the pooling layer, which are generated by a convolution layer.

Dense layer:
The neurons which are present are contributed by the previous layers input for [1] every particular neuron, during a layer so they are connected densely. We can also say it as fully connected layer, meaning all neurons are connected during a layer for the upcoming layer

Working Procedure of the Algorithm:

1. Begin the process of VGG16 algorithm.
2. Take the leaf of different types of shape and size to generate their images.
3. Do the data pre-processing to normalize, shuffle, split the dataset implement different other operations like feature extraction also.
4. Pre-processing images [15] mainly depends on RGB images for the colour classification of the image.
5. This VGG16 models uses 80% of the data as trained images and 20% of the data as testing images for classification of the healthy and diseased leafs.
6. In this model we used 9 diseased leaf groups and 1 healthy diseased leaf group for tomato leaf dataset.
7. Dataset is available at kaggle website of different leaf images.
8. VGG16 model is used to train/validate the dataset of the leaf images.
9. After the training/validating the model we uses confusion matrix on leaves and find out f1 score, Accuracy, Confusion metrics etc.
10. By using the algorithm we display the disease name and the remedies for it and calculate the accuracy rate, loss rate of the give model.

V. DATASET AND TOOLS

Plant Village Dataset:

Principally, the complete sets of images have been classified among 2 classes i.e. healthy and diseased. The entire dataset has been divided into 10 subject categories. We have collected about 16,012 images of which contain 1,591 images of healthy leaf and 14,421 images of the diseased leaf. Our target size is {(800, 200)}. It means 800 are trained images and 200 are testing images.

SOURCE: https://www.kaggle.com/emmarex/plantdisease

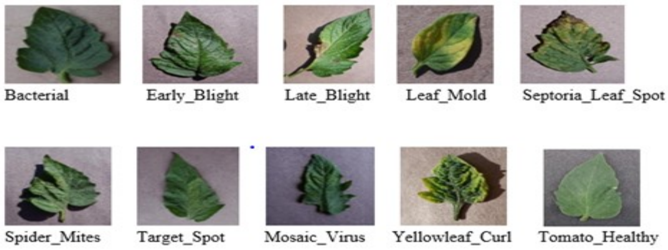


Figure-4: Healthy and Diseased leaf images

TOOLS AND TECHNOLOGIES USED

1. Python – 3.7(64bit):

Python 3.0 was introduced in the year 2008. Python may be an application-oriented language and object-oriented scripting language even through this version is used to be incompatible; afterward many of the attributes are converted into compatible with the help of current versions. This tells us about the gives adequate knowledge of the Python 3 version programming

2. Anaconda3-64 bit:

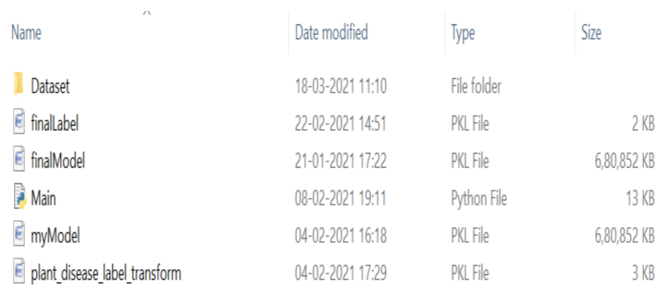
Anaconda is a package manager, clustering of many more open-source packages. It is free and easy to install required modules, libraries and packages which provide support from the community. Anaconda is used from writing programs of Python, R and data science for classification. Open the Anaconda Cheat Sheet from the browser and download Anaconda. Install conda and use it to install required packages by using it you will get Miniconda.

VI. RESULTS & OBSERVATIONS OF THE PROJECT

VI-A) DESCRIPTION OF RESULTS OBTAINED IN THE PROJECT

The VGG16 model is one of the deep learning techniques to build our project. We trained the leaf dataset to reduce noises and disturbances that present in the leaf. The libraries that are imported are tensorflow, keras, numpy, opencv.

The structure of files for the detection is as follows:



Name	Date modified	Type	Size
Dataset	18-03-2021 11:10	File folder	
finalLabel	22-02-2021 14:51	PKL File	2 KB
finalModel	21-01-2021 17:22	PKL File	6,80,852 KB
Main	08-02-2021 19:11	Python File	13 KB
myModel	04-02-2021 16:18	PKL File	6,80,852 KB
plant_disease_label_transform	04-02-2021 17:29	PKL File	3 KB

A) Dataset – This file is made up of both train and val (validation) images of leaves. Train file consists of 1 healthy leaf images and 9 diseased leaf images. Val file also consists of 1 healthy and 9 diseased leaf images.

B) finalLabel– This file is used for label binarizer.

C) finalModel–This file is used for loading the images.

D) Main – This file consists of the code that is used to run to detect the images.

E) myModel– This model is used for loading the model.

F) plant_disease_label_transform– This file is used for loading the labels.

The model is built by using the following procedure:

1. We need to import required libraries, modules and the images are loaded from it–

Create a python file in jupyter and name it as main.py and import required needed modules and libraries. Load the required tomato images from the particular dataset.

```
[INFO] Loading images ...
[INFO] Processing Tomato__Bacterial_spot
[INFO] Processing Tomato__Early_blight
[INFO] Processing Tomato__healthy
[INFO] Processing Tomato__Late_blight
[INFO] Processing Tomato__Leaf_Mold
[INFO] Processing Tomato__Septoria_leaf_spot
[INFO] Processing Tomato__Spider_mites Two-spotted_spider_mite
[INFO] Processing Tomato__Target_Spot
[INFO] Processing Tomato__Tomato_mosaic_virus
[INFO] Processing Tomato__Tomato_Yellow_Leaf_Curl_Virus
[INFO] Image loading completed

Total number of images: 1000
Total number of classes:  10
[INFO] Splitting data to train and test...
```

2. Preprocessing the leaf disease dataset -

Pre-processing techniques are used due to the clash of working with raw data for the system to easily understand. For this model go through the following,

a. Reading the images – First the images are taken from the dataset.

b. Resizing the images – The images taken from the dataset that is each image in the dataset contains different size and shape. In this step, it compresses and reduces the size of each image into one particular dimension. So that all the images contain same size. From this it will be easy to classify.

c. Removing the noises in the images – After resizing the images the images may contain some noisy data like scratches in the image, blurred images and the images with no clarity and so on. In this step, it removes all such type of data. So that it will be easy to classify the images.

d. Segmentation, separate the background from the foreground images. The background is separated to get more clarity of the images.

e. Morphology, smoothing edges of the images. After separating background from the images the edges are smoothened to remove any noisy and disturbances if any that are present.

f. Training and testing dataset are created.

Training data take the input entry and gives the output by creating the trained data.

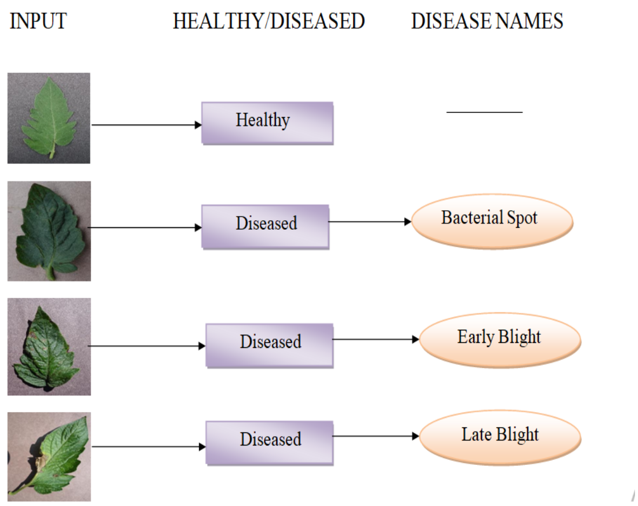
3. Training VGG16 model –

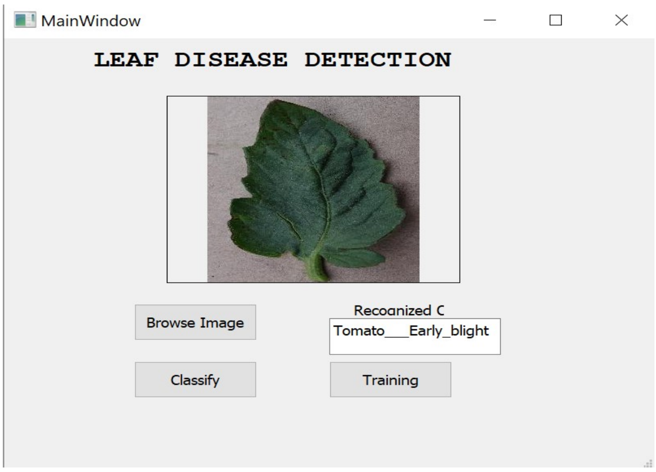
The architecture (vgg16) deep learning models consist of 16 layers. Convolution layers have stride 1 and filer 3x3 and consistently use zero padding and max pool layer have stride 2 and filter 2x2. The last layer is dense layer. After the training model of epochs is completed, we can only save the trained model by finding the model's accuracy reached 100%.

```
Epoch 1/10
  1/100 [..............................] - ETA: 1:35:55 - loss: 0.5679 - acc: 0.8125
  2/100 [..............................] - ETA: 1:14:40 - loss: 0.6001 - acc: 0.8438
  3/100 [..............................] - ETA: 1:02:37 - loss: 0.5544 - acc: 0.8583
  4/100 [>.............................] - ETA: 1:04:59 - loss: 0.5133 - acc: 0.8656
  5/100 [>.............................] - ETA: 1:02:46 - loss: 0.5233 - acc: 0.8600
  6/100 [>.............................] - ETA: 53:25 - loss: 0.4926 - acc: 0.8604
```

VI-B) EXPERIMENT AND TEST SUIT RESULTS:

In our experiment and test suit, the images are loaded from the dataset; the dataset contains 1 healthy and 9 diseased leaf images. Any image can be loaded. We can see the loaded image in the user interface. The model is already trained with the related information that consists in the files and the load the images related to the dataset, it verify the data and gives the feedback according to the image dataset. The loaded image is health, then it displays as healthy. If the loaded image is diseased, it displays the disease name.





After getting the output as whether the given leaf image is healthy or diseased we will find the accuracy and loss.

The respective formula for accuracy and loss is

$$\text{Accuracy} = \frac{\text{No. of images correctly classified}}{\text{Total no. of images}} \qquad \text{Loss} = -\sum_{i=1}^{n}\sum_{j=1}^{m} y\,i,j \log(p\,i,j)$$

- Accuracy is defined as no. of images classified correctly to the total no. of images.
- Loss is determined as the dissimilarity between the value predicted by our model and the true value. The formulas of accuracy and loss are represented above,

  y i,j means the true value. 1 if the given sample i be included in to the group j and otherwise it is 0.

  p i,j represents the probability that our model predicted of the given sample i be included in to the category j.

VI-C) METRICS VALUES, RESULTS AND GRAPH

1. The following are the metrics that are obtained for this project.

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + TN + FP}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

Here, TP = True Positive
TN = True Negative
FP = False Positive
FN = False Negative

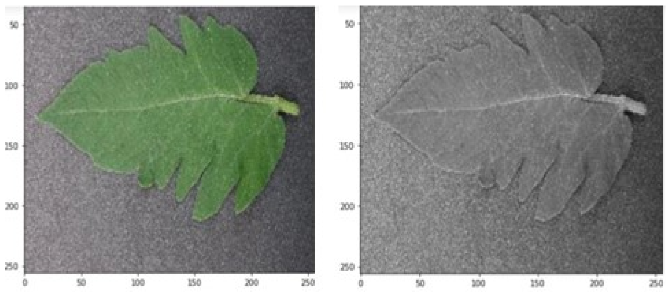
WARNING: tensorflow: Can save best model only with acc available, skipping.

	precision	recall	f1-score	support
0	1.00	1.00	1.00	336
1	1.00	0.99	0.99	179
2	1.00	0.99	0.99	47
3	0.99	1.00	1.00	511
4	0.99	1.00	0.99	49
5	0.99	1.00	0.99	51
6	1.00	0.99	0.99	220
7	1.00	1.00	1.00	234
8	0.99	1.00	0.99	83
9	1.00	0.99	0.99	415
accuracy			1.00	2125
macro avg	0.98	0.99	0.99	2125
weighted avg	1.00	1.00	1.00	2125

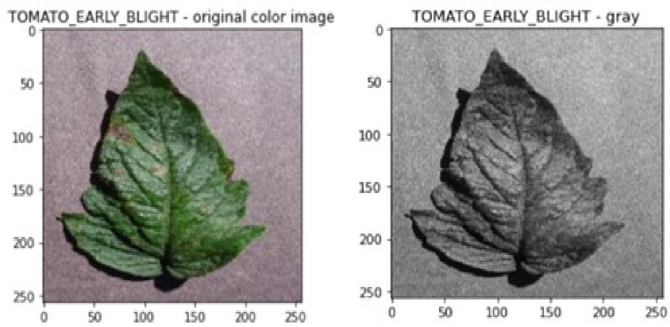
1000/1000 [==============================] - 90s 398ms/step - loss: 0.0201 - accuracy: 0.9997

2. The following are the metrics for the leaf images. The first one is healthy leaf of tomato and the second leaf is diseased leaf of the tomato i.e., Tomato Early Blight. Here we can see both color and gray images of the leaves. And the comparison and difference in the colored and gray images can be seen.

a. Tomato_ Healthy



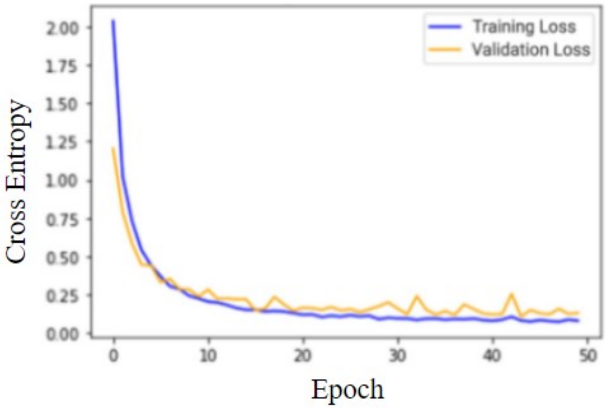
b. Tomato_Early_Blight



3. The below table shows us the attributes values represented in the table which used for calculating the accuracy of the VGG16 model that occurred at the time of each epoch.

Epoch	Train_loss	Valid_loss	Accuracy	Error_rate	Time
0	0.9598	0.0707	0.9997	0.0003	04:37
1	0.9820	0.0380	0.9998	0.0002	04:36
2	0.9900	0.0639	0.9999	0.0001	04:36
3	0.9820	0.2217	0.9997	0.0003	04:37

4. The below graph is drawn from the above table.



VI. CONCLUSION & FUTURE WORK

VII-A) CONCLUSION:

The conclusion on distinct diseases categorization techniques used for leaf disease detection and the algorithm used for the image segmentation and pre-processing technique that is used for automatic leaf diseases detection which is well known as classification. Apple and tomato types are used for testing the suggested algorithms. Therefore, allied diseases are taken from the plants for recognition. Estimating the slighter efforts of the superlative outcome which are produced, it expresses the competence of the suggested algorithm to identify and classification for the tomato leaf diseases. To provide better perform [16] for identification rate in the categorization process convolutional algorithms can also be used.

VII-B) FUTURE WORK:

Our future scope is to create a mobile app which will be useful for the farmers as a proper guide to do agriculture. In future, we can do the disease detection techniques using various parts of the crops or plants like stem, flower, and root. The project mainly focuses on the enlargement the given [10] methodology used to recognition and grouping for tomato leaf plant disease symptoms affected on leaves. However, the systems have a poor performance in Leaf miner and Septoria leaf spot. Future works may include enhancing the visual characterization and an additional evaluation.

References

[1] V. S. Babu, R. Satheesh Kumar, R. Sunder – "A comparative Study on Disease Detection of Plants Using Machine Learning Techniques", 2021 7th International Conference on Advanced Computing and Communicating Systems (ICACCS 2021)

[2] Yang Zhang, Chenglong Song, and Dongwen Zhang, "Deep Learning-Based Object Detection Improvement for Tomato Disease", IEEE, 2020

[3] Jun Sun, Yu Yang, Xiaofei He, and Xiaohong Wu, "Northern Maize Leaf Blight Detection under Complex Field Environment Based on Deep Learning", IEEE, 2020

[4] Peng Jiang, Yeuhan Chen, Bin Liu, Dongiian He, Chunquan Liang, "Real-Time Detection of Apple Leaf Diseases Using Deep Learning Approach Based on Improved Convolutional Neural Networks". IEEE, 2019

[5] Nikita Jadhav, HimaliKasar Prof. SumitaChandak, Shivani Machha, "Crop Leaf Disease Diagnosis using Convolutional Neural Network", Published 2020 Biology International Journal of Trend in Scientific Research and Development

[6] G. Geetha, S. Samundeswari, Saranya Gangadhara Moorthy, Meenakshi K., M. Nithya – "Plant Leaf Disease Classification and Detection System Using Machine Learning" – December 2020 Journal of Physics Conference Series (ICCPET 2020)

[7] M. Francis, C. Deisy, "Disease Detection and Classification in Agricultural Plants Using Convolutional Neural Networks", - A Visual Understanding Computer Science 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN) 2019

[8] S. Hari, M. Siva Kumar, P. Renuga, S. Karthikeyan, S. Suriya, - "Detection of Plant Disease by Leaf Image Using Convolutional Neural Network", Computer Science 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN) 2019

[9] Md Rahmat Ullah, Nagifa Anjum, Abdus Sattar – "Plant Disease Recognition Using Machine Learning", 2019 8th International Conference System Modeling and Advancement in Research Trenda (SMART 2019)

[10] MelikeSardogan, AdemTuncer, YunnusOzen, "Plant Leaf Disease Detection and Classification Based on CNN with LVQ Algorithm", 3rd International Conference on Computer Science and Engineering (UBMK), Posted 2018

[11] Shima Ramesh Maniyath, Vinod P V, Niveditha M, P. R, P. N, Shashank N, R.Hebbar – "Plant Disease Detection Using Machine Learning" – 2018 International Conference on Design Innovations for 3C's Compute Communicate Control (ICDI3C)

[12] P. Maheswari, P. Raja, N. M. Ghangoankar – "Intelligent Disease Detection System for Early Blight of Tomato Using Foldscope: A Pilot Study, 2018 IEEE 4th International Symposium in Robotics and Manufacturing Automation (ROMA)

[13] Yow-Wen Tian, Peng-Hui Zheng, Rui-Yao Shi, "The Detection System for Greenhouse Tomato Disease Degree Based on Android Platform", 3rd International Conference on Information Science and Control Engineering, Posted: 2016

[14] C Ravi, Jibu Shinde, C Mathew, Prof C Y Patil, "Segmentation Technique for Soybean Leaves Disease Detection", International Journal of Advanced Research, volume 3, issue 5, Posted: 2015

[15] M Ramakrishna and A SahayaAnselin Nisha, "Groundnut Leaf Disease Detection and Classification by using Nack Propagation Algorithm", IEEE International Conference, Melmaruvathur 2015

[16] U. Mokhtar, Mona A. S. Ali, Aboul Ella Hassenian, H. Hefny – "Tomato Leaves Diseases Detection Approach Based on Support Vector Machines", 2015 11th International Computer Engineering Conference (ICENCO 2015)