LEAF DISEASE DETECTION
(LEAFX)

MINI PROJECT REPORT

Submitted by

THIRUMURUGAN C	**REG.NO:19TD0699**
VIMALRAJI K	**REG.NO:19TD0705**

In fulfilment of the requirement for the award of the Degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

PONDICHERRY UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MANAKULA VINAYAGAR INSTITUTE OF TECHNOLOGY

KALITHEERTHALKUPPAM, PUDUCHERRY – 605107.

DECEMBER– 2021

LEAF DISEASE DETECTION
(LEAFX)

MINI PROJECT REPORT

Submitted by

THIRUMURUGAN C **REG.NO:19TD0699**

VIMALRAJI K **REG.NO:19TD0705**

In fulfilment of the requirement for the award of the Degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

PONDICHERRY UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MANAKULA VINAYAGAR INSTITUTE OF TECHNOLOGY

KALITHEERTHALKUPPAM, PUDUCHERRY – 605107.

DECEMBER– 2021

MANAKULA VINAYAGAR INSTITUTE OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE TECHNOLOGYAND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that the Mini Project work titled "**LEAF DISEASE DETECTION (LEAFX)**" is a bonafide word done by **THIRUMURUGAN C [REGIDTER NO: 19TD0699]** and **VIMALRAJI K [REGISTER NO: 19TD0705]**in fulfilment for the award of the degree of Bachelor of Technology in Computer Science Engineering of the Pondicherry University during the academic year 2021.

MR.R.RAJ BHARATH	**DR. S. PARISELVAM**
Assistant Professor	**Professor/ Head of Department**
Department of CSE	Department of CSE

ACKNOWLEDGEMENT

I am personally indebted of a number of persons that a complete acknowledgement would be encyclopaedia. First of all, I love to record my deepest gratitude to the Almighty Lord and my family.

I express my sincere and heartfelt thanks to our Chairman and Managing Director **Shri.M.DHANASEKARAN** for all this encouragement and moral support. I also thank our Vice Chairman **Shri.S.V.SUGUMARAN** and our Secretary **Dr. K. NARAYANASAMY** for this support and encouragement.

It gives me great ecstasy of pleasure to convey my deep and sincere thanks to our Principle **Dr. S. MALARKKAN**, for giving constant motivation in succeeding my goal. With profoundness I would like to express my sincere thanks to our beloved Head of Department, **Dr. S. PARISELVAM**, Professor, Computer Science and Engineering, for his kindness in extending the infrastructural facilities to carry out my project work successfully.

I extend my sincere gratitude and heartfelt thanks to our staff co-ordinator **MR.R.RAJ BHARATH,** Assistant Professor, Computer Science and Engineering, for providing the right ambience for carrying out this work and his valuable guidance and suggestions for our work. I think him, for the continuous encouragement and the interest shown to bring out our work at this stage and also for providing the freedom that I need and wanted.

I would like to express my gratitude to all teaching and non-teaching stall members of our department for the support they extended during the course of this project

ABSTRACT

Plant disease automation in agriculture science is the primary concern for every country, as the food demand is increasing at a fast rate due to an increase in population. Moreover, the increased use of technology today has increased the efficacy and accuracy of detecting diseases in plants and animals. The detection process marks the beginning of a series of activities to fight the diseases and reduce their spread. Some diseases are also transmitted between animals and human beings, making it hard to fight them. For many years, scientists have researched how to deal with the common diseases that affect humans and plants. However, there are still many parts of the detection and discovery process that have not been completed. The technology used in medical procedures has not been adequate to detect all diseases on time, and that is why some diseases turn out to become pandemics because they are hard to detect on time. Our focus is to clarify the details about the diseases and how to detect them promptly with artificial intelligence. We discuss the use of machine learning and deep learning to detect diseases in plants automatically. Our study also focuses on how machine learning methods have been moved from conventional machine learning to deep learning in the last five years. Furthermore, different data sets related to plant diseases are discussed in detail. The challenges and problems associated with the existing systems are also presented.

TABLE OF CONTENTS

1. CHAPTER INTRODUCTION

1.1. OVERVIEW

The use of technology in the detection and analysis process increases the accuracy and reliability of these processes. For example, the people who use the latest technology to analyze the diseases that arise unexpectedly are at a higher chance of controlling them than those that do not. In the recent occurrence of coronavirus, the world relied on the latest technology to develop preventive measures that have helped reduce the rate at which the disease is transmitted. Crop diseases are a significant threat to human existence because they are likely to lead to droughts and famines. They also cause substantial losses in cases where farming is done for commercial purposes. The use of computer vision (CV) and machine learning (ML) could improve the detection and fighting of diseases. Computer vision is a form of artificial intelligence (AI) that involves using computers to understand and identify objects. It is primarily applied in testing drivers, parking, and driving of self-driven vehicles and now in medical processes to detect and analyze objects. Computer vision helps increase the accuracy of disease protection in plants, making it easy to have food security.

One of the areas that CV has helped most is the detection of the severity of the diseases. Deep learning (DL), a part of the CV, is useful and promising in determining the severity of diseases in plants and animals. It is also used to classify diseases and avoid the late detection of diseases. Plant diseases are slightly different from those that affect human beings. Many factors make diseases similar as well. However, the diseases that can be transmitted from humans to plants and vice versa are rare. The analysis of the data related to this field helps identify how the use of the latest technology can be improved. The images of leaves and other parts of the plants can be used to detect diseases in plants. The technology could be applied in analysing images in human beings that also prove the presence of diseases and determine the extent of their destruction. This research study is aimed at analysing the way image-based technology can be used in detecting diseases in both plants and animals.

1.2. BACKGROUND

ML is the technology that allows machines to communicate with human beings and understand their needs. It also makes machines act like human beings and make the decision on behalf of humans. It is one of the areas that have grown fast over the past few years. ML helps in classifying plant diseases. The use of this technology is seen as a significant beginning and achievement in dealing with plant diseases. It has also increased productivity in the field of cultivation. Visualization techniques have also been included in this technology, and it has been improved over the last three years to the current improved levels. The challenges that face the world today, related to the diseases affecting plants and humans, can be reduced if the diseases are identified before they spread to vast areas. The use of ML is widespread in the world today. Diverse methods used in ML and DL help the experts to analyze the plant diseases and know their source in time. The detection of these diseases is affected mainly by several challenges that affect the effectiveness and accuracy of this technology.

The first challenge is the time complexity associated with the use of ML and DL, whereby some of the technologies used in the detection of these diseases are outdated or based on some information from the past. The other challenge is segmentation sensitivity. It means that the region of interest (RoI) requires a high level of accuracy and sensitivity to acquire the required usage and accuracy. The other challenge is that there is a language barrier that affects the way the technology is applied. Another challenge is the inadequate resources that are required to support the application of this technology. Most of the ML and DL activities need many resources to use and implement. Private and government entities usually fund the institutes that use this technology to detect diseases in humans and plants, which could affect the success of the research and implementation of the technology.

The importance of plants in the world has increased over time. The discoveries about the critical roles that plants could play in medicine, energy production, and the recent concerns about the reduction of global warming have for long been a significant part of science and technology. A reduction in the plant cover in the world increases the risk of higher global warming and an increase in the related challenges. The need to build a state-of-the-art convolutional system that supports the image detection technology and classification of plant diseases has led to many research programs to provide the scientists with the required knowledge. Image detection could be applied when necessary to differentiate healthy leaves from those that are not healthy. The convolutional neural networks (CNNs) provide the differences among plant images that help determine the abnormalities that could exist in the plants in the natural environment. The background study shows that the scanning of the images that show the healthy and unhealthy plants forms a basis for comparison by the scientists in this field.

DL can be used to detect abnormalities in both humans and plants. The pixel-wise operations are used to analyze the leaves collected from sick plants, and this is used to classify the diseases according to their impact on the plants. The visible patterns in these leaves are used to decide the diseases that affect the plants and how they can be dealt with to prevent them from spreading. Research shows that the use of DL technology is up to 98.59% accurate. The field of plant pathology has contributed immensely to the control of diseases and reduced global warming. One of the essential background knowledge areas that guides the use of image detection technology is that the leaves of the infected plants are different from the healthy ones. The leaves are likely to have dark parts, and some may be dry along the edges. The dried parts are also likely to fold up, and this is easy to detect even with a bare eye. The use of ML is to detect these differences without human intervention.

The application of the latest technology to detect and control the disease that affects production is a reliable way of reducing the challenges of food inadequacy. Most of the plants are also used as raw materials in many industries. If such plants are of low quality, it means that they will lead to the production of low-standard products.

2. CHAPTER METHODS

2.1. DATASET DESCRIPTION

We analyze 54,306 images of plant leaves, which have a spread of 38 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. Figure 1 shows one example each from every crop-disease pair from the Plant Village dataset. In all the approaches described in this paper, we resize the images to 256 × 256 pixels, 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 Village dataset. We start with the Plant Village dataset as it is, in colour; then we experiment with a grey-scaled version of the Plant Village dataset, and finally we run all the experiments on a version of the Plant Village dataset where the leaves were segmented, hence removing all the extra background information which might have the potential to introduce some inherent bias in the dataset due to the regularized process of data collection in case of Plant Village dataset. Segmentation was automated by the means of a script tuned to perform well on our particular dataset. We chose a technique based on a set of masks generated by analysis of the colour, lightness and saturation components of different parts of the images in several colour spaces (Lab and HSB). One of the steps of that processing also allowed us to easily fix colour casts, which happened to be very strong in some of the subsets of the dataset, thus removing another potential bias.

This set of experiments was designed to understand if the neural network actually learns the “notion” of plant diseases, or if it is just learning the inherent biases in the dataset.

2.2. MEASUREMENT OF PERFORMANCE

To get a sense of how our approaches will perform on new unseen data, and also to keep a track of if any of our approaches are overfitting, we run all our experiments across a whole range of train-test set splits, namely 80–20 (80% of the whole dataset used for training, and 20% for testing). It must be noted that in many cases, the Plant Village dataset has multiple images of the same leaf (taken from different orientations), and we have the mappings of such cases for 41,112 images out of the 54,306 images; and during all these test-train splits, we make sure all the images of the same leaf go either in the training set or the testing set. Further, for every experiment, we compute the mean precision, mean recall, mean F_1 score, along with the overall accuracy over the whole period of training at regular intervals (at the end of every epoch). We use the final mean F_1 score for the comparison of results across all of the different experimental configurations.

2.3. APPROACH

MobileNetV2, that improves the state-of-the-art performance of mobile models on multiple tasks and benchmarks as well as across a spectrum of different model sizes. We also describe efficient ways of applying these mobile models to object detection in a novel framework we call SSD Lite. Additionally, we demonstrate how to build mobile semantic segmentation models through a reduced form of DeepLabv3 which we call Mobile DeepLabv3. The MobileNetV2 architecture is based on an inverted residual structure where the input and output of the residual block are thin bottleneck layers opposite to traditional residual models which use expanded representations in the input an MobileNetV2 uses lightweight depth wise convolutions to filter features in the intermediate expansion layer. Additionally, we find that it is important to remove non-linearities in the narrow layers in order to maintain representational power. We demonstrate that this improves performance and provide an intuition that led to this design. Finally, our approach allows decoupling of the input/output domains from the expressiveness of the transformation, which provides a convenient framework for further analysis. We measure our performance on ImageNet classification, COCO object detection, VOC image segmentation. We evaluate the trade-offs between accuracy, and number of operations measured by multiply-adds (MAdd), as well as the number of parameters

2.3.1. CHOICE OF TRANING MODEL

- Transfer Learning
- Training from Scratch

2.3.2. CHOICE OF DATASET TYPE

- Colour
- Gray Scale
- Leaf Segmented

2.3.3. SOME OF HYPER-PARAMETERS

- Batch Size: 32
- Early Stop: True

- Optimizer: Adam
- Loss: Categorial Cross Entropy
- Epochs: 15

2.4. WORKING OF OUR NEURAL NETWORK(MBNET_V2)

The model was developed by using Google's MobileNet_v2 model we use it as in the way of transfer learning, so that it can easily go their weights so that our module will give the high accuracy. First layer will be MobileNet_v2 transfer learning layer after the end of the Google's MobileNet_v2 we add a layer called GlobalAveragePooling2D and then we add a layer Dropout with the value of **0.2** this is because of that our model should not memorise the pattern so we will add one dropout layer with that value. The last layer will be a dense with the length of **32 node** and the activation is **SoftMax**.

2.5. SUMMARY OF OUR MODEL

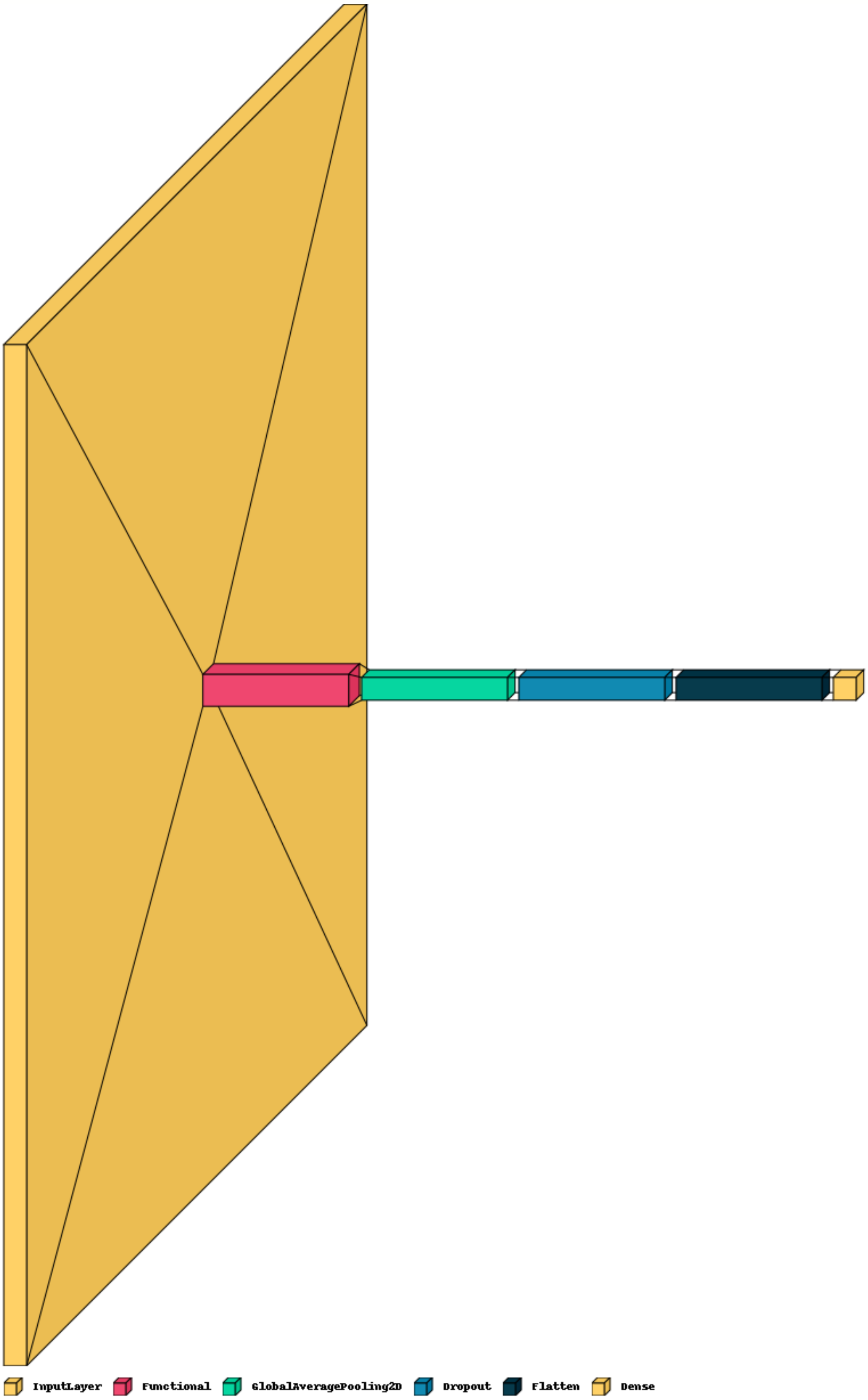
```
Layer (type)                Output Shape              Param #
=================================================================
input_2 (InputLayer)        [(None, 224, 224, 3)]     0

mobilenetv2_1.00_224           (None, 7, 7, 1280)     2257984
(Functional)

global_average_pooling2d    (G  (None, 1280)          0
lobalAveragePooling2D)

dropout (Dropout)           (None, 1280)              0

fatten (Fatten)             (None, 1280)              0

dense (Dense)               (None, 38)                48678

=================================================================
Total params: 2,306,662
Trainable params: 48,678
Non-trainable params: 2,257,984
_________________________________________________________________
```

2.6. LAYERS INVOLVED IN THE MODEL

- MobileNet_V2(Transfer Learning)
- GlobalAveragePooling2D
- Dropout
- Flatten

- Dense

2.7. VISUALIZATION OF THE MODEL



MODULES INVOLVED

3.1. CLASSIFICATION OF THE APPLICATION

Our application is worked on the way where in the principle of client server scenario such that our mobile application act as a client and there is server which is run as localhost in our system. Application is developed with the languages **Flutter & Dart** for the Server it was developed by the **Flask Python**. The work flow diagrams are shown below, the working process is the client that is the user will capture the image from the mobile and then it will send it to the server. Then the server will process the image and detect the disease of the leaf. If the leaf is affected by the disease the server will shows the remedies otherwise if the leaf is not affected by the any disease, it will show the use of that plant or fruit.

3.1.1. MODULES IN APP

- Home Module
  - ✓ Price Scrapper
  - ✓ Weather Scrapper
- Camera Module
- Info Module
  - ✓ Defects Screen
  - ✓ Usage Screen

3.1.2. MODULES IN SERVER

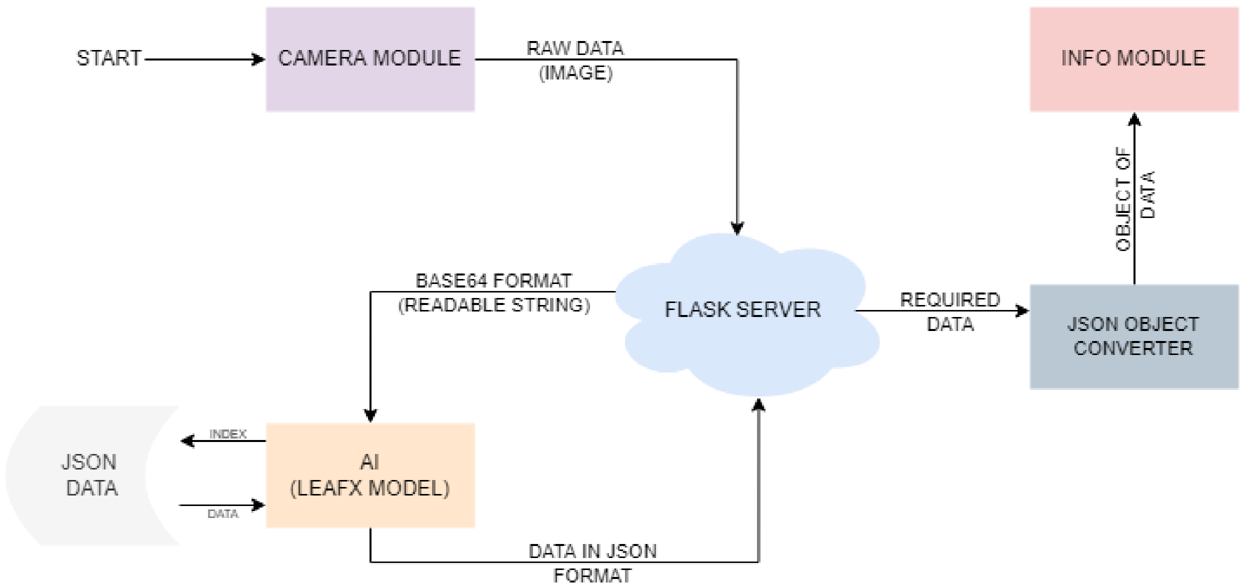
- AI Model Trainer
- Flask Server
  - ✓ Model Predicter

3.2. WORK FLOW OF MODEL

3.2.1. MAIN WORK FLOW

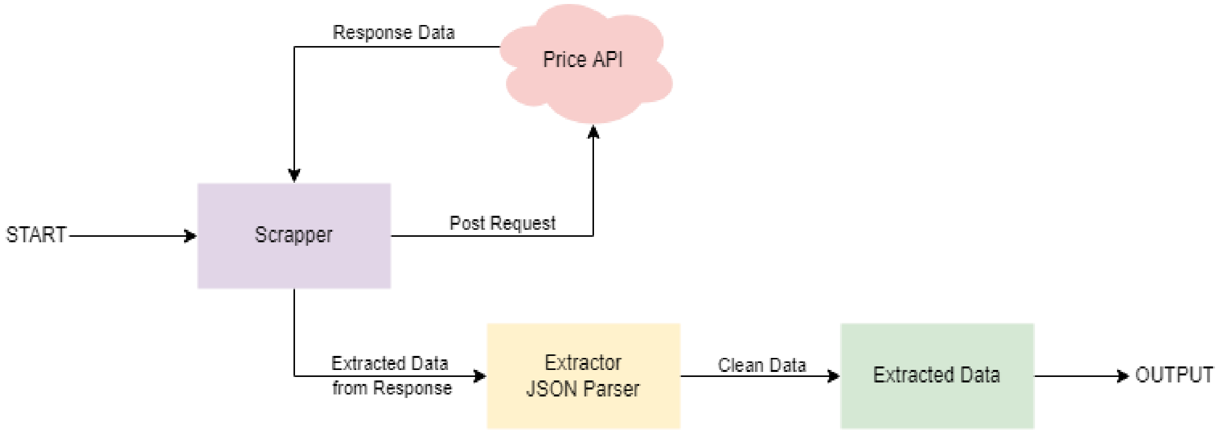
The main work flows our project will show. It will show how the data will flow through the client and server how will be the data goes to the server and then it will show the data selection of data then how it converts into the object all these functions are pictorially represented here. So, we are able to understand the data flow and control flow of our project. And we are also having some disadvantage which our model is not able to detect either it is a leaf or not for that we have to first detect the whether it is a leaf or not if it is a leaf then do the further process or else just show that it is not a leaf.

3.2.2. OVERALL FLOW OF DATA

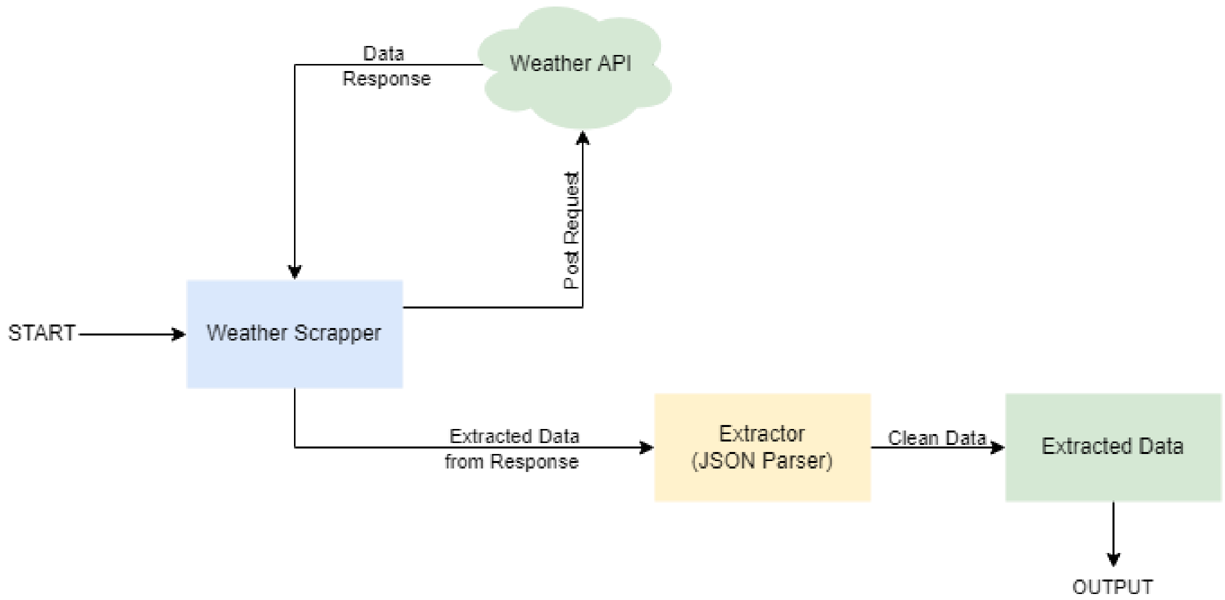


The overall work flow of the app is divided in two main part like APIs call and Server call. Where in APIs call from the user side the request is send to the standard APIs so that it will return the result in JSON format from that we have to convert it into an Object model so that we are able to use it easily. In the second part the app is going take an image of the leaf or select the image of the leaf and then it sends to the server in the form of base64 format for now we have used local server. In Server side the give base64 format image is converted into greyscale image then flatten the image and give the flattened image to the model. The model will return the probability base on the accuracy among the give dataset we have to get the max accuracy and resend the appropriate json response to the client and display it to the user. Work flow of all process was represented as block diagram for better diagram.

3.2.3. PRICE SCRAPPER:



3.2.4. WEATHER SCRAPPER:

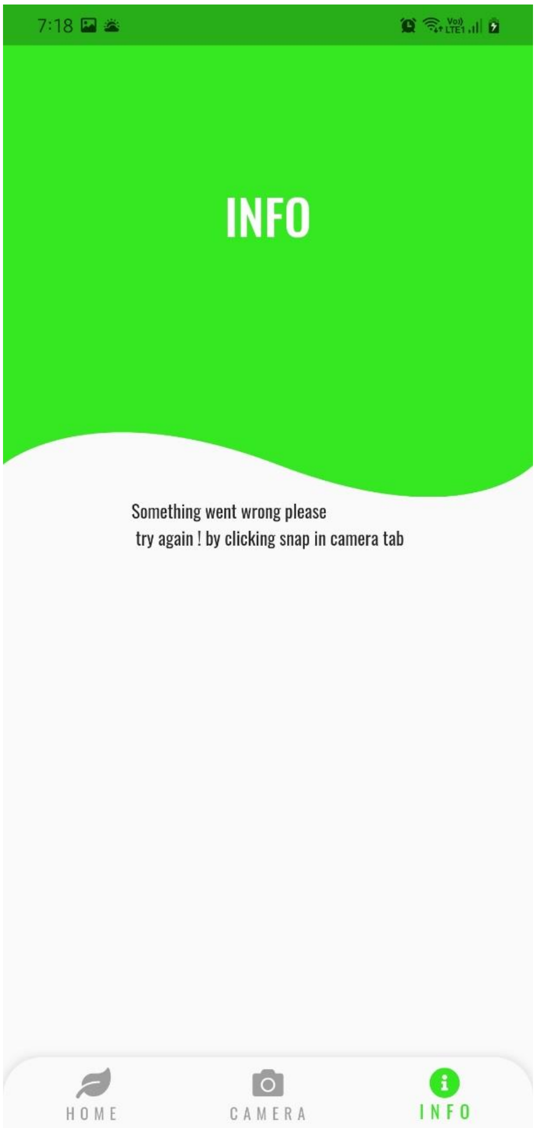


SCREENSHOT

Home Page

Camera Page

Info Page



Home Page On API Request Loading

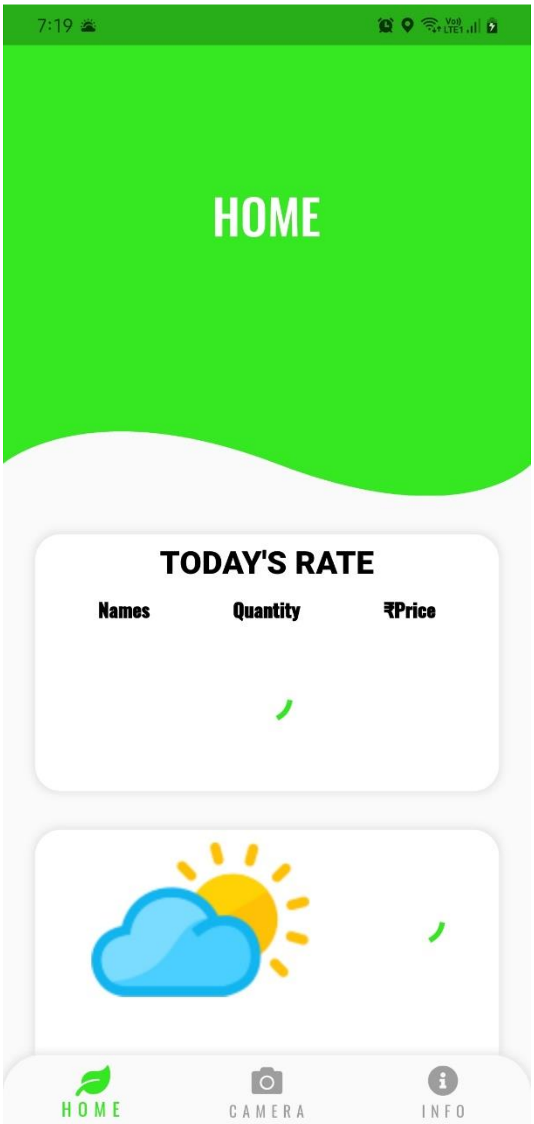
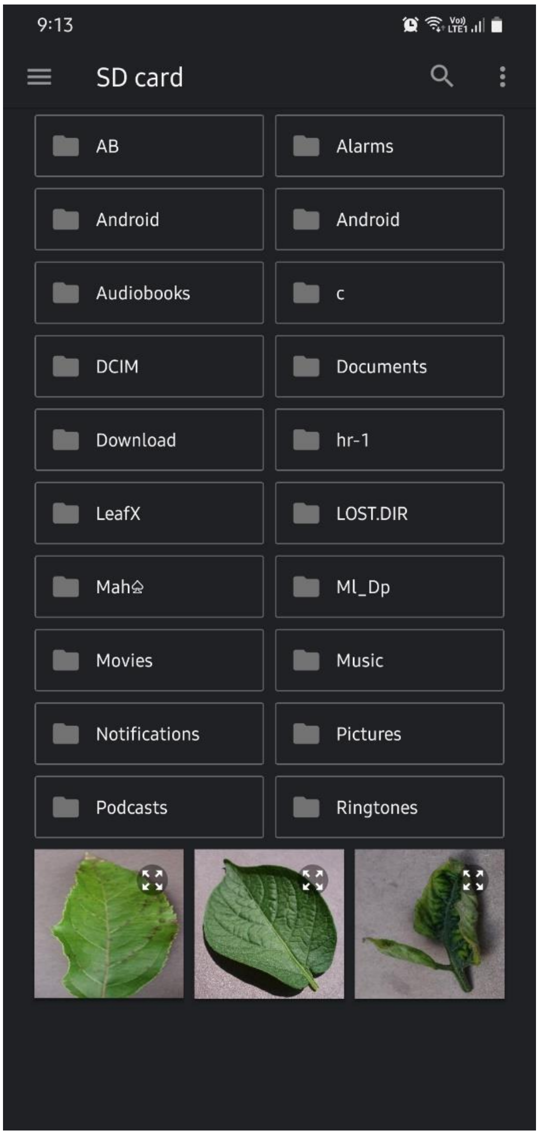


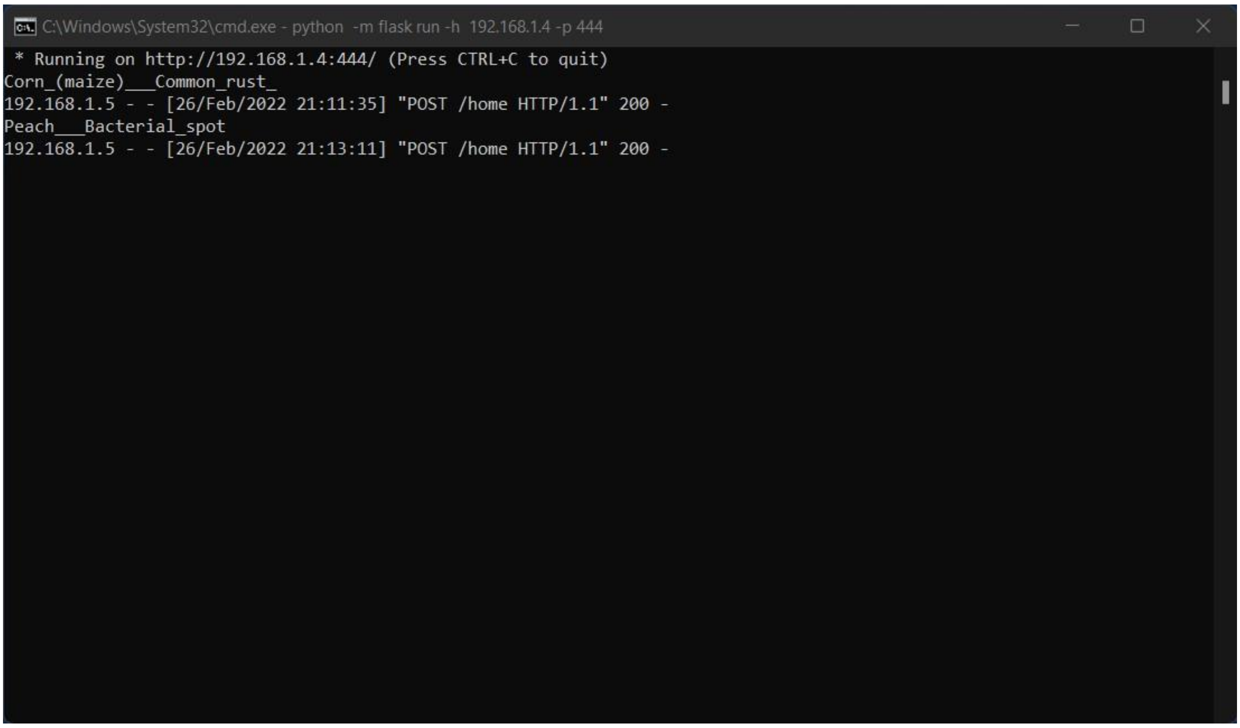
Image Selecting Page



Info Page Dieases Disctiption

Info Page Benefits Description

Server Output



5. CHAPTER SOFTWARE TOOLS, PROGRAMMING LANGUAGE

5.1. SOFTWARE TOOLS

- Vscode
- Postman API
- Dart Widget Inspector
- GitHub (Version Tracking)

5.2. PROGRAMMING LANGUAGE

- Python
- Dart
- Flutter

Vscode

Visual Studio Code is a streamlined code editor with support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs, such as Visual Studio IDE.

Postman API

Postman is an API (application programming interface) development tool which helps to build, test and modify APIs. ... It has the ability to make various types of HTTP requests (GET, POST, PUT, PATCH), saving environments for later use, converting the API to code for various languages (like JavaScript, Python).

Dart Widget Inspector

The Flutter widget inspector is a powerful tool for visualizing and exploring Flutter widget trees. The Flutter framework uses widgets as the core building block for anything from controls (such as text, buttons, and toggles), to layout (such as centering, padding, rows, and columns).

GitHub

GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. This tutorial teaches you GitHub essentials like repositories, branches, commits, and pull requests. Create and use a repository. Start and manage a new branch.

Python

Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small- and large-scale projects.

Dart

DART systems are designed to sense pressure changes at the bottom of the ocean caused by passing tsunamis and to communicate these changes to the tsunami warning centres. Each DART system consists of a bottom pressure recorder anchored to the ocean floor and a separately moored companion surface buoy.

Flutter

Flutter is Google's platform-independent technology for creating applications used on mobile, desktop, and web platforms. Flutter has the advantages of native development and allows you to reuse most of the code among different platforms, which can reduce your budget costs considerably.

6. CHAPTER CONCLUSION

This paper presents the survey on different diseases classification techniques used for plant leaf disease detection and an algorithm for image segmentation technique that can be used for automatic detection as well as classification of plant leaf diseases later. Apple, Blueberries, Corn, Grapes, Peach, Pepper Bell, Potato, Raspberry, Sour cherry, Soyabeans, Strawberry and Tomato are some of those **twelve species** on which proposed algorithm is tested. Therefore, related diseases for these plants were taken for identification. With very less computational efforts the optimum results were obtained, which also shows the efficiency of proposed algorithm in recognition and classification of the leaf diseases. Another advantage of using this method is that the plant diseases can be identified at early stage or the initial stage. To improve recognition rate in classification process Artificial Neural Network, Bayes classifier, Fuzzy Logic and hybrid algorithms can also be used.

7. CHAPTER REFERENCES

[1] Ghaiwat Savita N, Arora Parul. Detection and classification of plant leaf diseases using image processing techniques: a review. Int J Recent Adv Eng Technol 2014;2(3):2347–812. ISSN (Online).

[2] Dhaygude Sanjay B, Kumbhar Nitin P. Agricultural plant leaf disease detection using image processing. Int J Adv Res Electr Electron Instrum Eng 2013;2(1).

[3] Mrunalini R Badnakhe, Deshmukh Prashant R. An application of K-means clustering and artificial intelligence in pattern recognition for crop diseases. Int Conf Adv Inf Technol 2011;20. 2011 IPCSIT.

[4] Arivazhagan S, Newlin Shebiah R, Ananthi S, Vishnu Varthini S. Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features. Agric Eng Int CIGR 2013;15(1):211–7.

[5] Kulkarni Anand H, Ashwin Patil RK. Applying image processing technique to detect plant diseases. Int J Mod Eng Res 2012;2(5):3661–4.

[6] Bashir Sabah, Sharma Navdeep. Remote area plant disease detection using image processing. IOSR J Electron Commun Eng 2012;2(6):31–4. ISSN: 2278-2834.

[7] Naikwadi Smita, Amoda Niket. Advances in image processing for detection of plant diseases. Int J Appl Innov Eng Manage 2013;2(11).

[8] Patil Sanjay B et al. Leaf disease severity measurement using image processing. Int J Eng Technol 2011;3(5):297–301.

[9] Chaudhary Piyush et al. Color transform based approach for disease spot detection on plant leaf. Int Comput SciTelecommun 2012;3(6).

[10] Rathod Arti N, Tanawal Bhavesh, Shah Vatsal. Image processing techniques for detection of leaf disease. Int J Adv Res Comput Sci Softw Eng 2013;3(11).

[11] Beucher S, Meyer F. The morphological approach to segmentation: the watershed transforms. In: Dougherty ER, editor. Mathematical morphology image processing, vol.12. New York: Marcel Dekker; 1993. p. 433–81.

[12] Bhanu B, Lee S, Ming J. Adaptive image segmentation using a genetic algorithm. IEEE Trans Syst Man Cybern Dec 1995;25:1543–67.

[13] Bhanu B, Peng J. Adaptive integrated image segmentation and object recognition. IEEE Trans Syst Man Cybern Part C 2000;30:427–41.

[14] Eric Windmill. Flutter and Dart **2019**