A PROJECT REPORT

ON

”Predictive Model for Crop Disease Detection”

Submitted to
UNIVERSITY OF MUMBAI

In partial fulfillment of the requirements for the degree of
Bachelor of Engineering

BY

(KAPADIA MOHAMMED ANAS TARIQ HEENA) Roll No. 22DCO01
((SOBAN WAJUDDIN MARUF BILKIS) Roll No. 21CO58
((SHAIKH SHADULLA KHURSHID ALAM WAHIDA TABBASUM) Roll No. 21CO51
((SHAIKH AMAAN RAFIQUE NOORJAHAN) Roll No. 21CO43

UNDER THE GUIDANCE OF
(Prof.Abdul Majid Ansari)

Department of Computer Engineering,
Anjuman-I-Islam’s Kalsekar Technical Campus
School of Engineering and Technology
Plot No. 2 & 3, Sector -16, Near Thana Naka, Khanda Gaon,
New Panvel, Navi Mumbai. 410206

Academic Year : 2023-2024

AFFILIATED TO

UNIVERSITY OF MUMBAI

Anjuman-I-Islam's Kalsekar Technical Campus

Department of Computer Engineering,

School of Engineering and Technology,
Anjuman-I-Islam's Kalsekar Technical Campus

Khanda Gaon,New Panvel, Navi Mumbai. 410206

(KAPADIA MOHAMMED ANAS TARIQ HEENA)	Roll No. 22DCO01
((SOBAN WAJUDDIN MARUF BILKIS)	Roll No. 21CO58
((SHAIKH SHADULLA KHURSHID ALAM WAHIDA TABBASUM)	Roll No. 21CO51
((SHAIKH AMAAN RAFIQUE NOORJAHAN)	Roll No. 21CO43

CERTIFICATE

This is to certify that the project entitled
"Predictive Model for Crop Disease Detection"
submitted by

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Engineering) at **Anjuman-I-Islam's Kalsekar Technical Campus, Navi Mumbai** under the University of MUMBAI. This work is done during year 2023-2024, under our guidance.

Date: 25/4/2024

Prof.Abdul Majid Ansari — Supervisor

Prof. Nusrat Jahan — Project Coordinator

Prof. Tabrez Khan — Head of Department

Dr. Rajendra Magar — Dean, SoET

External Examiner

ACKNOWLEDGMENT

We would like to take the opportunity to express our sincere thanks to our guide **Prof. Abdul Majid** , Assistant Professor, Department of Computer Engineering, AIKTC, School of Engineering, Panvel for his/her invaluable support and guidance throughout our project research work. Without his/her kind guidance & support this was not possible.

We are grateful to his/her for his/her timely feedback which helped us track and schedule the process effectively.His/her time, ideas and encouragement that he/she gave is help us to complete our project efficiently.

We would like to express deepest appreciation towards **Dr. Rajendra Magar**, Dean, SoET, AIKTC, Navi Mumbai, **Prof. Tabrez Khan**, Head of Department of Computer Engineering and **Prof. Nusrat Jahan**, Project Coor- dinator whose invaluable guidance supported us in completing this project.

At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped us directly or indirectly during this course of work.

(KAPADIA MOHAMMED ANAS TARIQ (22DCO01)

(SOBAN WAJUDDIN MARUF BILKIS (21CO58)

(SHAIKH SHADULLA KHURSHID ALAM (21CO51)

(SHAIKH AMAAN RAFIQUE (21CO43)

Mini Project Approval for Bachelor of Engineering

This project entitled ***Predictive Model for Crop Disease Detection*** by ***Soban Wajuddin Maruf (21CO58)***, **Kapadia Mohammed Anas Tariq(22DCO01)** , **Shaikh ShadullaKhurshid Alam(21CO51)** and **Shaikh Amaan Rafique(21CO43)**is approved for the degree of ***Bachelor of Engineering in Department of Computer Engineering.***

Examiners

1...............................
2...............................

Supervisors

1...............................
2...............................

Chairman

....

Declaration

We declare that this written submission represents our ideas in our own words and where other ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(KAPADIA MOHAMMED ANAS TARIQ (22DCO01)

(SOBAN WAJUDDIN MARUF BILKIS (21CO58)

(SHAIKH SHADULLA KHURSHID ALAM (21CO51)

(SHAIKH AMAAN RAFIQUE (21CO43)

Abstract

Title: Predictive Model for Crop Disease Detection

This project presents a predictive model for the detection of crop diseases using machine learning techniques. The aim is to address the challenges faced by farmers in timely identification and management of crop diseases, which significantly affect agricultural productivity. The proposed model utilizes a dataset of images depicting various crop diseases and healthy crops, employing deep learning algorithms for feature extraction and classification. Through extensive experimentation and validation, the model demonstrates promising accuracy in identifying and classifying crop diseases, thereby enabling early detection and intervention. The implementation of this model holds potential to revolutionize agricultural practices, offering farmers a proactive tool for disease management and crop yield optimization.

Keywords:
1. **Predictive Model**
2. **Crop Disease Detection**
3. **Machine Learning**
4. **Agricultural Expertise**
5. **Historical Data**

Name: Anas, Soban, Shadulla, Amaan.
Roll No: 22DCO01, 21CO58, 21CO51, 21CO43
B.E. (Computer Engineering)
University of Mumbai.

Contents

Chapter 1

Introduction

Agriculture is considered as driving sector of a nation's economy. One of the unsolved problems in Agriculture is, Detection of Type of disease for a crop during the right time.

In Traditional method, Farmers by their experience, or through the knowledge of other or by any other Media, they may come to know about the Diseases that occurred. From past few decades, Government is establishing Farmer and Crop Welfare centres in large scale such that farmers with any of their queries, can visit the centre and clarify them. When there is any outbreak of a disease, professionals from those centres visit the field, take some sample, conduct some research and declare the result and remedies.

Though it is an effective process, sometimes the results of the tested samples aren't quick enough. By the time enough damage might have been done. Hence to solve this problem, a tool can be developed that uses computer vision, that can classify the image based on the disease.

This tool works after it gets trained by the images, of disease that already occurred at some point of time. Based on those data, the tool extracts the features of the diseases (like brown spots, weakened stems, folded leaf's, insets, etc) from the images. Stores those features. In future when the farmer uses that tool, if the data of that disease is already trained, then it easily identifies the disease with in a second. We can also get the remedy from further spread of the disease and detect the disease at it initial state itself. In case if it is a new disease then it will be a professional work.

There are many ways to classify the image, but one of them, the most popular way is by using CNN (Convolutional Neural Network) a Deep Learning concept, which is used mostly for image classification and Object Detection. In order to make the task faster we are using the concept of Transfer Learning. This reduces the cost of building and training the algorithm, and we will get a more accurate result, than by using traditional CNN.

1.1 Purpose

The primary purpose of the project is to enable early detection of crop diseases. By leveraging machine learning and image processing techniques, the model can analyze visual data from crops and identify symptoms of diseases at their earliest stages. Early detection allows farmers to take timely action to prevent the spread of diseases and minimize crop losses.

The project aims to contribute to the implementation of precision agriculture practices.

By providing farmers with accurate and timely information about crop health, the predictive model enables targeted interventions, such as precise application of pesticides or fungicides, thereby optimizing resource use and minimizing environmental impact.

1.2 Project Scope

1. Dataset Collection: - The process starts with collecting input images representing different types of crop leaves, such as potatoes, tomatoes, and peppers. - Raw images can be gathered using a real-time camera or mobile device. - For training and testing the deep learning model, a publicly available dataset is used during the framework's testing and training phases. 2. Pre-

processing: - Raw images collected from the dataset may contain noise and imperfections that need to be addressed before feeding them into the learning module. - Preprocessing techniques such as rotation, resizing, and shearing are applied to enhance the quality of the images. 3.

Training and Building the Model: - This step consists of two main phases: training the Transfer Learning (TL) models using a training image dataset and validating the architecture using test images for performance evaluation. - The TL models are trained in the initial phase, and the model's performance is evaluated using test images in the later phase. 4. Model Construction:

- The process of building the predictive model involves the following steps: - Collecting images from the dataset. - Preprocessing image data by resizing and rotating images. - Creating convolute feature connections into Fully Connected Layers. - Flattening the features, converting them into a one-dimensional array, and connecting them to one or more fully connected layers. - Extracting features for different classes of input

1.3 Project Goals and Objectives

The utilization of a large dataset, diverse pre-trained deep learning models, versatile leaf classification capabilities, and the development of a smart web application enable accurate crop disease prediction, proactive intervention, and resource conservation for farmers, promoting sustainable agriculture practices and enhancing overall crop management strategies.

1.3.1 Goals

The goal of the project is to develop a robust predictive model for crop disease detection that aids farmers in early identification and management of crop diseases, thereby enhancing agricultural productivity and food security.

1.3.2 Objectives

The primary objective of this project is to develop a machine learning-based system for the automatic classification of plant diseases using image analysis techniques.

1. Dataset Collection: Gather a comprehensive dataset comprising images of various crop diseases along with images of healthy crops across different growth stages and environmental conditions.

2. Data Preprocessing: Preprocess the collected dataset by performing tasks such as image resizing, normalization, and augmentation to ensure uniformity and enhance model training.

3. Model Selection and Training: Select appropriate machine learning algorithms, such as convolutional neural networks (CNNs), and train the model using the preprocessed dataset to accurately classify crop diseases.

4. Model Evaluation: Evaluate the performance of the trained model using relevant metrics such as accuracy, precision, recall, and F1-score on a separate validation dataset to ensure robustness and generalization.

5. Interface Development: Develop an intuitive user interface that allows farmers to easily upload images of their crops and receive timely predictions about the presence of diseases.

6. Impact Assessment: Assess the impact of the predictive model on agricultural productivity, crop yield, economic returns, and food security, and analyze its contribution to mitigating the effects of crop diseases.

Chapter 2

Literature Survey

2.1 DeepCrop: Deep learning-based crop disease prediction with web application [Volume 14, December 2023]

Journal of Agriculture and Food Research
(Author:- Md. Manowarul Islam a, Md Abdul Ahad Adil)

2.1.1 Methodology

The methodology of DeepCrop involves collecting input images of crop leaves, preprocessing them to address noise and imperfections, training Transfer Learning (TL) models using a dataset, and validating the model's performance with test images. The model construction process includes collecting and preprocessing images, creating convolute feature connections, flattening features, and extracting features for different input classes, optimization algorithms are likely utilized in the training and optimization of the models. Overall, DeepCrop's methodology focuses on utilizing deep learning techniques, such as Convolutional Neural Networks (CNNs), to accurately predict crop diseases and provide valuable insights for farmers to prevent crop failure and increase productivity.

2.1.2 Merits

DeepCrop's utilization of a large dataset, diverse pre-trained deep learning models, versatile leaf classification capabilities, and the development of a smart web application enable accurate crop disease prediction, proactive intervention, and resource conservation for farmers, promoting sustainable agriculture practices and enhancing overall crop management strategies.

2.1.3 Demerits

DeepCrop's weaknesses encompass potential data quality dependency, overfitting risks, limited scope in leaf classification, challenges in web application implementation, and the necessity for continuous maintenance and updates to ensure adaptability to evolving agricultural needs.

2.2 Identification of Plant Disease using Image Processing Technique [IEEE 2019]

(Author:- Abirami Devaraj, Karunya Rathan)

2.2.1 Methodology

The methodology for plant disease detection using image processing techniques involves capturing leaf images with a digital camera, preprocessing the images to enhance quality through size adjustments, noise filtering, and contrast enhancement using MATLAB code, segmenting the images using k-means clustering, extracting features from the segmented images using the Gray-Level Co-occurrence Matrix (GLCM) methodology, and classifying the images as healthy or diseased using the Random Forest Classifier. This classifier is trained on the extracted features from both healthy and diseased plant images, enabling it to predict and classify new images based on learned patterns. While specific equations are not provided, these established techniques in image processing and machine learning play a crucial role in automating disease detection in agriculture, aiming to reduce agricultural losses and ensure crop quality.

2.2.2 Merits

The utilization of image processing techniques for plant disease identification provides automated, accurate, and quantitative assessments, enabling early detection, objective decision-making, scalability, adaptability, and support for sustainable agriculture practices, thereby revolutionizing disease management in agriculture and enhancing crop productivity while minimizing losses.

2.2.3 Demerits

The application of image processing for plant disease identification is accompanied by challenges such as the complexity and technical expertise required for implementation, costs associated with infrastructure and training, dependence on data quality and variability impacting algorithm performance, the need for continuous optimization and fine-tuning of algorithms, limitations in generalizability across different agricultural contexts, and ethical considerations regarding data privacy, algorithm bias, and potential social implications, highlighting the importance of addressing these demerits to ensure effective and equitable deployment of image processing technologies in agriculture.

2.3 Image Processing Technique for Automatic Detection of Plant Diseases and Alerting System in Agricultural Farms [IEEE 2020]

(Author:-Pradeep Kumar Mugithe, Rohit Varma Mudunuri)

2.3.1 Methodology

The methodology employed in the automated disease detection and alerting system involves several key steps, starting with image acquisition of plant leaves, followed by image pre-processing to enhance image quality. Subsequently, image segmentation is performed to isolate plant leaves, and feature extraction is carried out to characterize the leaves. The K-Means clustering algorithm is then applied to group similar features, leading to classification of segmented regions into healthy or diseased areas. The system categorizes detected diseases and triggers an alerting mechanism, such as a buzzer, to notify farmers promptly. This comprehensive approach enables timely disease detection, alerting farmers to take necessary actions, ultimately improving crop yield and reducing losses.

2.3.2 Merits

The merits of the proposed automated disease detection and alerting system outlined in the PDF file "Image Processing Technique for Automatic Detection" are significant and contribute to improving agricultural practices. Here is a detailed explanation of the merits:

1.Early Disease Detection: One of the primary merits of the system is its ability to detect plant diseases at an early stage. By utilizing image processing techniques and clustering algorithms, the system can identify diseases in plants before they become severe. Early detection allows farmers to take timely action, such as targeted treatment or removal of infected plants, preventing the spread of diseases and minimizing crop damage.

2.Accuracy and Precision: The system boasts a high level of accuracy in disease detection, as indicated by the results showing an accuracy rate of 95.1613

3.Real-Time Alerting System: A key merit of the system is its real-time alerting mechanism, which triggers a buzzer to notify farmers immediately upon detecting a disease in plants. This prompt alerting system enables farmers to respond swiftly to disease outbreaks, take preventive measures, and mitigate the spread of diseases across the farm. The timely alerts help in preventing extensive damage to crops and improving overall farm productivity.

4.Enhanced Farm Management: By automating the disease detection process and integrating an alerting system, the proposed system contributes to enhanced farm management practices. Farmers can proactively monitor plant health, identify diseases early, and implement targeted interventions, leading to improved crop yield and reduced losses. The system empowers farmers with valuable information to make informed decisions and optimize agricultural operations.

2.4 A Comparative Analysis of Machine Learning Algorithms for Detection of Organic and Nonorganic Cotton Diseases

(Author:- Sandeep Kumar,Arpit Jain)

2.4.1 Methodology

"A Comparative Analysis of Machine Learning Algorithms for Detection of Organic and Nonorganic Cotton Diseases," researchers employed a methodology that involved capturing digital images of cotton leaves, preprocessing the images to enhance quality, extracting RGB features, segmenting the images using edge detection techniques, and classifying the segmented regions as infected or non-infected using machine learning algorithms such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Decision Tree. Evaluation metrics including accuracy, precision, recall, sensitivity, specificity, F1 score, true positive rate, true negative rate, and error rate were utilized to assess algorithm performance. By applying these techniques and metrics, the study aimed to compare the effectiveness of different algorithms in detecting diseases in cotton crops, providing valuable insights for the agricultural industry and suggesting potential improvements in disease detection methods.

2.4.2 Merits

The methodology employed in the study offers several merits, including the utilization of image processing techniques to enhance the quality of digital images of cotton leaves, the extraction of RGB features to capture color information, the segmentation of images using edge detection for precise identification of infected regions, and the classification of segmented regions as infected or non-infected using machine learning algorithms, providing a systematic approach to disease detection in cotton crops and enabling the comparison of algorithm performance through a comprehensive evaluation using various metrics.

2.4.3 Demerits

The study may have limitations such as the potential for bias in the selection of image processing techniques and machine learning algorithms, the reliance on a specific dataset that may not fully represent all variations of organic and nonorganic cotton diseases, the complexity of feature extraction and segmentation methods that could impact the accuracy of disease detection, and the need for further validation and testing on a larger and more diverse dataset to ensure the generalizability and robustness of the proposed approach.

Chapter 3

Project Planning

3.1 Members and Capabilities

3.1.1 Kapadia Mohammed Anas Tariq (22DCO01)

1. Course Completed : ReactJS, Python, MERN Stack
2. Course Pursuing : App Development
3. Number of Projects Made : 6
4. Name of the Projects Made : Netflix Clone, Amazon Clone, Stock Management, Sleep Disorders and Diet using Data Analytics, Smart Farming, Crop Disease Prediction using ML.

3.1.2 Soban Wajuddin Maruf (21CO58)

1.Course Completed : HTML, CSS, Blender, MERN Stack
2. Course Pursuing : App Development
3. Number of Projects Made : 4
4. Name of the Projects Made : RealLife,Game, Smart Farming, Crop Disease Prediction using ML.

3.1.3 Shaikh Shadulla Khurshid Alam (21CO51)

1. Course Completed : Cloud Computing, Blender, Unreal Engine, Figma, Python
2. Course Pursuing : None
3. Number of Projects Made : 5
4. Name of the Projects Made : RealLife, Portfolio Website

3.1.4 Shaikh Amaan Rafique (21CO43)

1. Course Completed : HTML, CSS, JavaScript, PHP
2. Course Pursuing : MERN Stack
3. Number of Projects Made : 1
4. Name of the Projects Made : Digital Canteen

3.2 Roles and Responsibilities

3.2.1 Kapadia Mohammed Anas Tariq (22DCO01)

1. Role: Programmer
2. Responsibilities: Model Architect

3.2.2 Soban Wajuddin Maruf (21CO58)

1. Role: Programmer
2. Responsibilities: Researcher

3.2.3 Shaikh Shadulla Khurshid Alam (21CO51)

1. Role: Desginer
2. Responsibilities: UI/UX Designer and Technical Writer

3.2.4 Shaikh Amaan Rafique (21CO43)

1. Role: Programmer
2. Responsibilities: Fronted Developer

3.3 Assumptions and Constraints

3.3.1 Assumptions

1. Data Quality: Assumption that the training data for the predictive model is accurate, representative, and devoid of significant biases.

2. Feature Relevance: Assuming that the selected features for disease detection are indicative of the health status of crops and are sufficient for accurate predictions.

3. Homogeneity: Assuming uniformity in the environmental conditions across the dataset, such as soil type, climate, and farming practices.

4. Model Generalization: Assuming that the developed model can generalize well to unseen data, including different regions, seasons, and crop varieties.

5. Stable Model Performance: Assuming that the performance of the model remains consistent over time and is not significantly impacted by external factors such as climate change or emergence of new diseases.

3.3.2 Constraints

1. Data Availability: Limited availability of labeled datasets for training the predictive model, which may restrict the model's accuracy and generalization capabilities.

2. Resource Limitations: Constraints on computational resources, including processing power and memory, which may limit the complexity of the model or the size of the dataset that can be processed.

3. Real-time Processing: Constraints related to real-time processing of crop images for disease detection, including latency requirements and hardware limitations for on-site detection.

4. Interpretability: Balancing model complexity with interpretability, considering that overly complex models might be difficult for end-users, such as farmers, to understand and trust.

5. Regulatory Compliance: Compliance with local regulations and ethical considerations regarding data privacy, especially if the project involves collecting and processing sensitive information from farmers or agricultural organizations.

3.4 Software and Hardware Requirements

3.4.1 Software Requirements

1. The system should provide an easy and efficient GUI to use.
2. The system should be able to provide clear picture about the status of the leaf.
3. If the leaf is found unhealthy i.e., diseased, the system should tell what type of disease that the leaf is infected with.
4.The system should also provide the remedies for the disease on which the plant is infected.

3.4.2 Hardware Requirements

Laptop or desktop with:
• Intel i5 or equivalent
• 4GB RAM or more.

3.4.3 Software Requirements

IDE: Jupyter Notebook
• Language: Python
• CNN

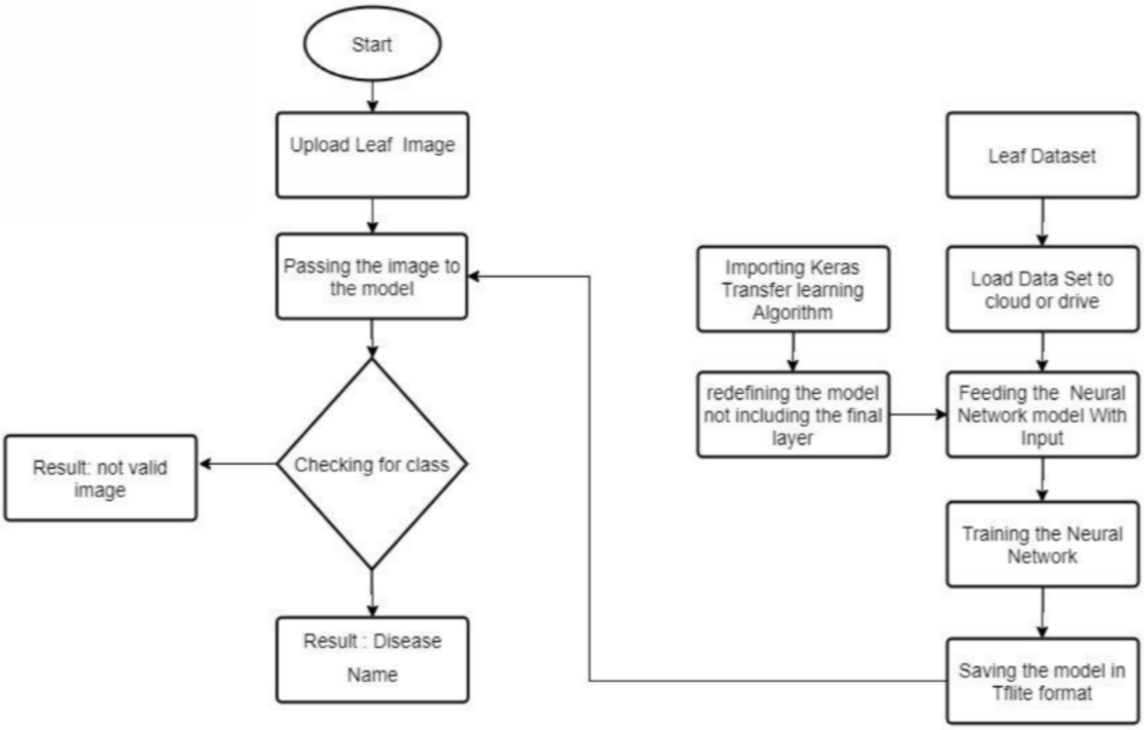
3.5 Project Timeline

1. January 2024 - Resource Gathering and Paper Collection
2. February 2024 - Design Planning and Survey
3. March 2024 - Model Building and Actual Implementation
4. April 2024 - Final Implementation, Report and Final Presentation

Chapter 4

System Architecture Design

4.1 Architecture Diagram



4.2 Methodology

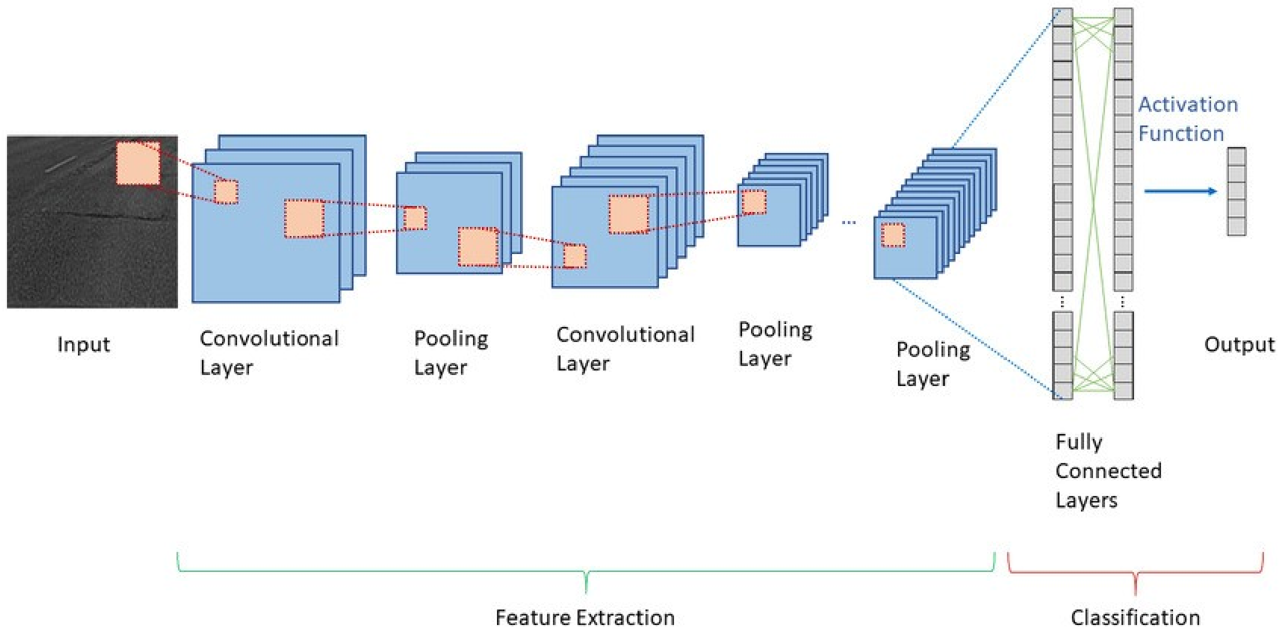
4.2.1 Pre Data Collection

The main aim is to design a system which is efficient and which provide disease name and pesticides name as fast as possible. For that purpose, we use two phases: 1st is training phase and 2nd is testing phase. In 1st phase: Image acquisition, Image Pre-processing and CNN based training. In 2nd phase Image acquisition, Image Pre-processing. A. Image Acquisition:

For training, image is taken from a file on the system. For the purpose of testing the images of the plant leaves are captured as and when required and then transferred to a folder on the system for analysis.

B. Image Pre-Processing: Image should be processed before sending to the algorithm for testing and training purpose. For that purpose, in this project image is scaled or resize into 50 x 50 dimension. As we used colour image so that we don't need any colour conversion techniques and that pre-processed image is directly passed to algorithm for training.

C. Convolutional Neural Network: Once pre-processing is done, then CNN is used for training purpose and after that we get attained model. That CNN method is written with help of tensor flow. By using this model, we classify the image that the system is getting after pre-processing of testing image. Then we get particular disease name or status of healthy leaf if there is no disease on that leaf. With the help of that disease name, we get the remedies which will help the farmer to take action so as to eradicate or decrease the effect of disease.



1. Input Layer: The input layer contains the input images and their pixel values.

2. Convolution Layer The main building blocks of CNN is the Convolutional layer. It is the simple use of filters of several sizes into the input image that results in activation. The repeated use of such input image with the same filter results in the map in the activation called feature map. This indicates the strength of the detected features in the input image. Every feature map has convoluted with the help of numerous input feature graph in Convolutional neural networks.

3. Pooling Layer A pooling layer often follows a convolutional layer and can be utilized to depreciate the dimensions of feature maps and parameters of the network. Pooling layers are also invariant in interpretation, alike to convolutional layers because their calculations take into account neighbouring pixels. The most widely used approaches are average pooling and max pooling.In our research, we used a max-pooling layer.

4. Non-Linear Layer A non-linear transformation is applied to the input by the CNN, the

object of which is to classify the features within per hidden layer. In CNN structure we use Rectified linear units (ReLU). Rectified linear units are commonly used as nonlinear transformation. This kind of layer executes a simple operation with a threshold where any input value smaller than zero is set to zero.

5. Fully Connected Layer The data arrives at the last layer of the CNN, which is the fully connected node, later much iteration of the prior layers. In the two neighbouring layers, the neurons are connected directly to the neurons within the fully connected network.

6. Normalize Layer In our proposed system we use a batch normalize layer. Batch normalization layer form normalizes any channel through a mini-batch. This can help to decrease sensitivity to data variations.

7. Softmax Layer The network's performance can be difficult to interpret. It is normal to finish the CNN with a Softmax function in classification issues. After extracting values of 15 classes of plant diseases in the fully connected step, a Softmax will be made for them, so that the class will be selected in each process and according to the features that were extracted through the previous layers that the images of plant diseases went through it. In this layer, the correct class of disease is determined by applying the Softmax function.

8. Training Training a network is a procedure of obtaining kernels in convolution layers and weights in fully connected layers that reduce differences on a training dataset between output predictions and specified ground truth labels. In our work, we used 87

4.2.2 Test Case

Test Case ID	Test Input	Expected Results	Actual Results	Remark
1	Select an image with bacterial spot	Should identify the image is diseased with bacterial spot and provide remedies	Identified the leaf is infected with bacterial spot and provided remedies	Pass
2	Select an image with Yellow curl leaf virus	Should identify the image is diseased with yellow curl leaf virus and provide remedies	Identified the leaf is infected with yellow curl leaf virus and provided remedies	Pass
3	Select an image with Late Blight disease	Should identify the image is diseased with light blight and provide remedies	Identified the leaf is infected with late blight disease and provided remedies	Pass
4	Select a healthy image	Should identify the image as healthy	Identified the image status as healthy	Pass

4.2.3 Features of Website

Features of our Model are as follows:
1. Pre-processing techniques for image enhancement.
2. Real-time data acquisition via IoT devices.
3. User-friendly interface for image capture and upload.
4. Intuitive visualization of disease detection results.
5. Scalability and adaptability to different crops and regions.
6. Emphasis on transparency and interpretability.
7. Aim to revolutionize agricultural practices.

Chapter 5

Implementation

5.1 Code Snippets

Data Preprocessing

```
[ ] # Dataset Path
    base_dir = 'plantvillage dataset/color'
```

```
image_path = '/content/plantvillage dataset/color/Apple___Cedar_apple_rust/025b2b9a-0ec4-4132-96ac-7f2832d0db4a___FREC_C.Rust 3655.JPG'

# Read the image
img = mpimg.imread(image_path)

print(img.shape)
# Display the image
plt.imshow(img)
plt.axis('off')  # Turn off axis numbers
plt.show()
```

```
(256, 256, 3)
```

Train Test Split

```
[ ] # Image Data Generators
    data_gen = ImageDataGenerator(
        rescale=1./255,
        validation_split=0.2  # Use 20% of data for validation
    )
```

```
[ ] # Train Generator
    train_generator = data_gen.flow_from_directory(
        base_dir,
        target_size=(img_size, img_size),
        batch_size=batch_size,
        subset='training',
        class_mode='categorical'
    )
```

```
Found 43456 images belonging to 38 classes.
```

Convolutional Neural Network

```
[ ] # Model Definition
    model = models.Sequential()

    model.add(layers.Conv2D(32, (3, 3), activation='relu', input_shape=(img_size, img_size, 3)))
    model.add(layers.MaxPooling2D(2, 2))

    model.add(layers.Conv2D(64, (3, 3), activation='relu'))
    model.add(layers.MaxPooling2D(2, 2))


    model.add(layers.Flatten())
    model.add(layers.Dense(256, activation='relu'))
    model.add(layers.Dense(train_generator.num_classes, activation='softmax'))
```

```
# model summary
model.summary()
```

Model training

```
# Training the Model
history = model.fit(
    train_generator,
    steps_per_epoch=train_generator.samples // batch_size,  # Number of steps per epoch
    epochs=10,  # Number of epochs
    validation_data=validation_generator,
    validation_steps=validation_generator.samples // batch_size  # Validation steps
)
```

```
Epoch 1/10
1358/1358 [==============================] - 107s 79ms/step - loss: 0.5471 - accuracy: 0.8311 - val_loss: 0.4335 - val_accuracy: 0.8644
Epoch 2/10
1358/1358 [==============================] - 105s 77ms/step - loss: 0.2052 - accuracy: 0.9341 - val_loss: 0.3831 - val_accuracy: 0.8833
Epoch 3/10
1358/1358 [==============================] - 106s 78ms/step - loss: 0.1122 - accuracy: 0.9631 - val_loss: 0.4650 - val_accuracy: 0.8816
Epoch 4/10
1358/1358 [==============================] - 107s 79ms/step - loss: 0.0795 - accuracy: 0.9743 - val_loss: 0.5403 - val_accuracy: 0.8689
Epoch 5/10
1358/1358 [==============================] - 106s 78ms/step - loss: 0.0605 - accuracy: 0.9809 - val_loss: 0.6543 - val_accuracy: 0.8671
Epoch 6/10
1358/1358 [==============================] - 104s 77ms/step - loss: 0.0545 - accuracy: 0.9835 - val_loss: 0.6985 - val_accuracy: 0.8508
Epoch 7/10
1358/1358 [==============================] - 103s 76ms/step - loss: 0.0552 - accuracy: 0.9831 - val_loss: 0.7335 - val_accuracy: 0.8581
Epoch 8/10
1358/1358 [==============================] - 108s 79ms/step - loss: 0.0425 - accuracy: 0.9871 - val_loss: 0.6369 - val_accuracy: 0.8782
Epoch 9/10
1358/1358 [==============================] - 107s 79ms/step - loss: 0.0417 - accuracy: 0.9881 - val_loss: 0.9539 - val_accuracy: 0.8501
Epoch 10/10
1358/1358 [==============================] - 108s 79ms/step - loss: 0.0312 - accuracy: 0.9911 - val_loss: 0.7326 - val_accuracy: 0.8573
```

Model Evaluation

```
[ ] # Model Evaluation
    print("Evaluating model...")
    val_loss, val_accuracy = model.evaluate(validation_generator, steps=validation_generator.samples // batch_size)
    print(f"Validation Accuracy: {val_accuracy * 100:.2f}%")
```

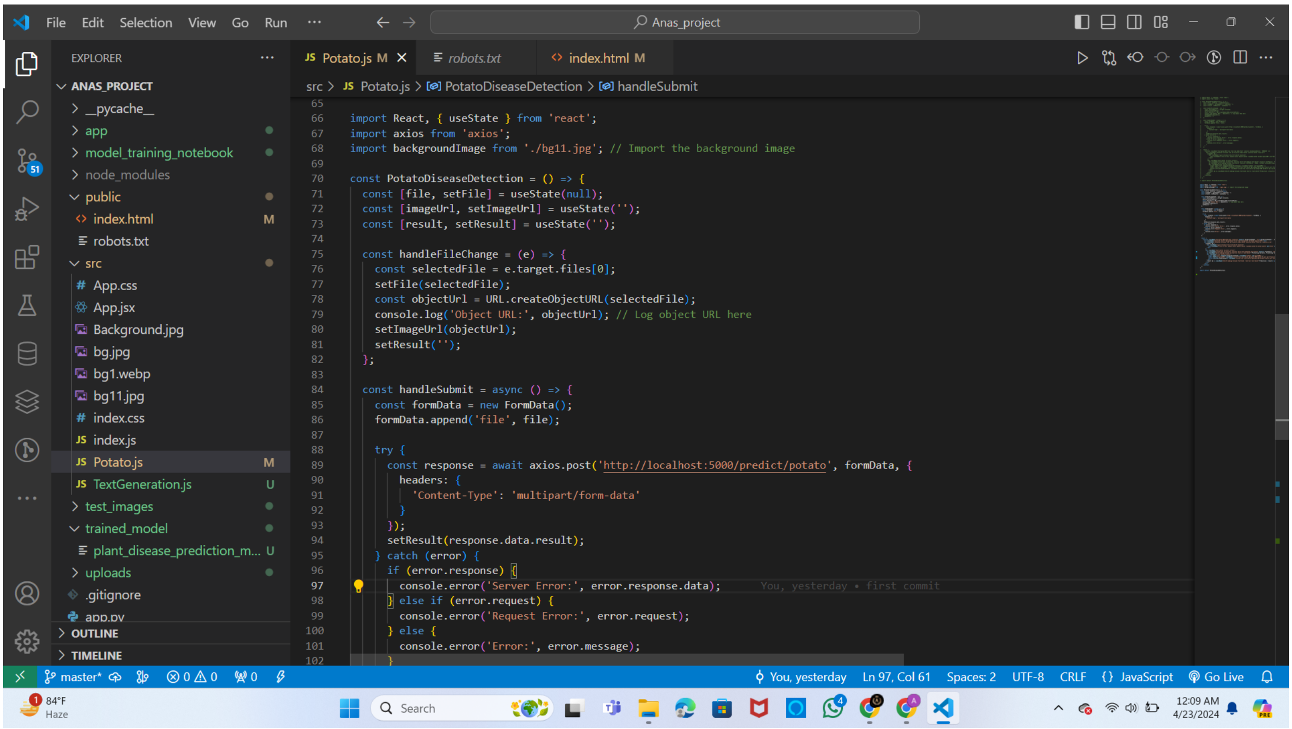
```
Evaluating model...
339/339 [==============================] - 25s 75ms/step - loss: 0.7326 - accuracy: 0.8573
Validation Accuracy: 85.73%
```

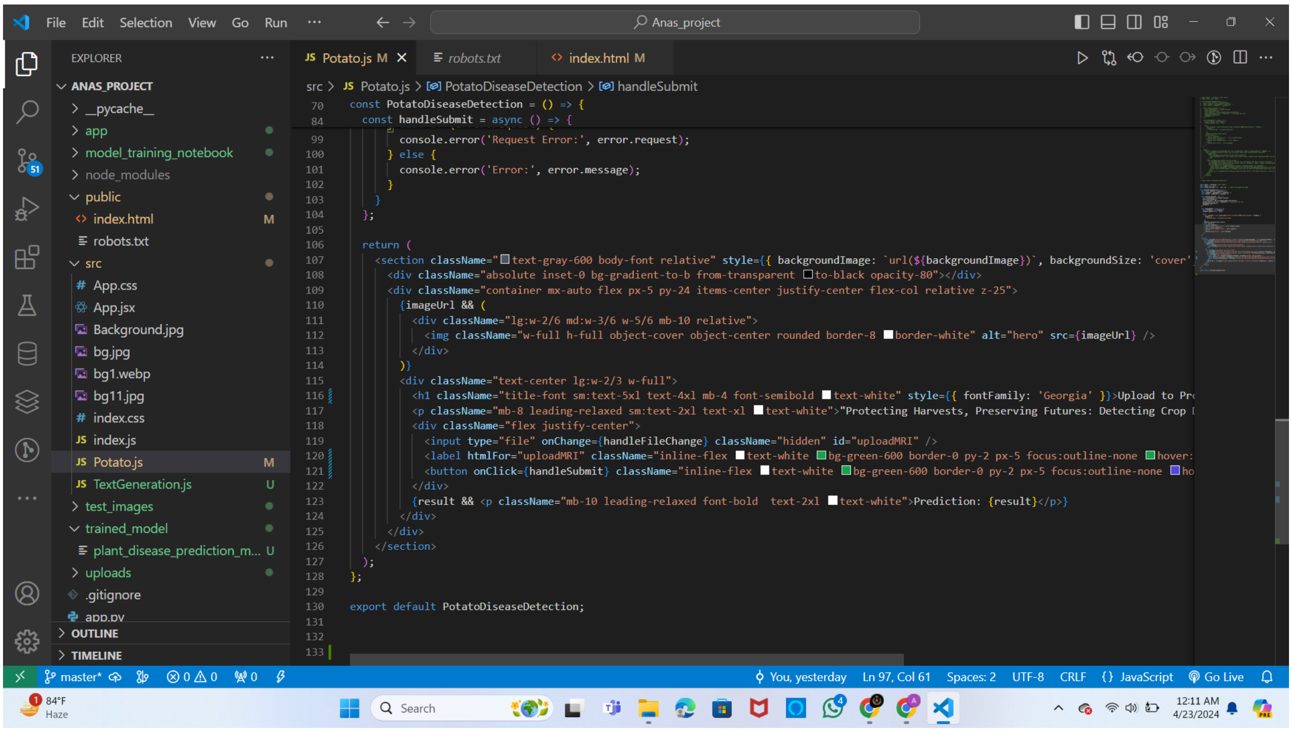
```
# Plot training & validation accuracy values
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('Model accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()

# Plot training & validation loss values
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```

Building a Predictive System

```
[ ] # Function to Load and Preprocess the Image using Pillow
    def load_and_preprocess_image(image_path, target_size=(224, 224)):
        # Load the image
        img = Image.open(image_path)
        # Resize the image
        img = img.resize(target_size)
        # Convert the image to a numpy array
        img_array = np.array(img)
        # Add batch dimension
        img_array = np.expand_dims(img_array, axis=0)
        # Scale the image values to [0, 1]
        img_array = img_array.astype('float32') / 255.
        return img_array

    # Function to Predict the Class of an Image
    def predict_image_class(model, image_path, class_indices):
        preprocessed_img = load_and_preprocess_image(image_path)
        predictions = model.predict(preprocessed_img)
        predicted_class_index = np.argmax(predictions, axis=1)[0]
        predicted_class_name = class_indices[predicted_class_index]
        return predicted_class_name
```

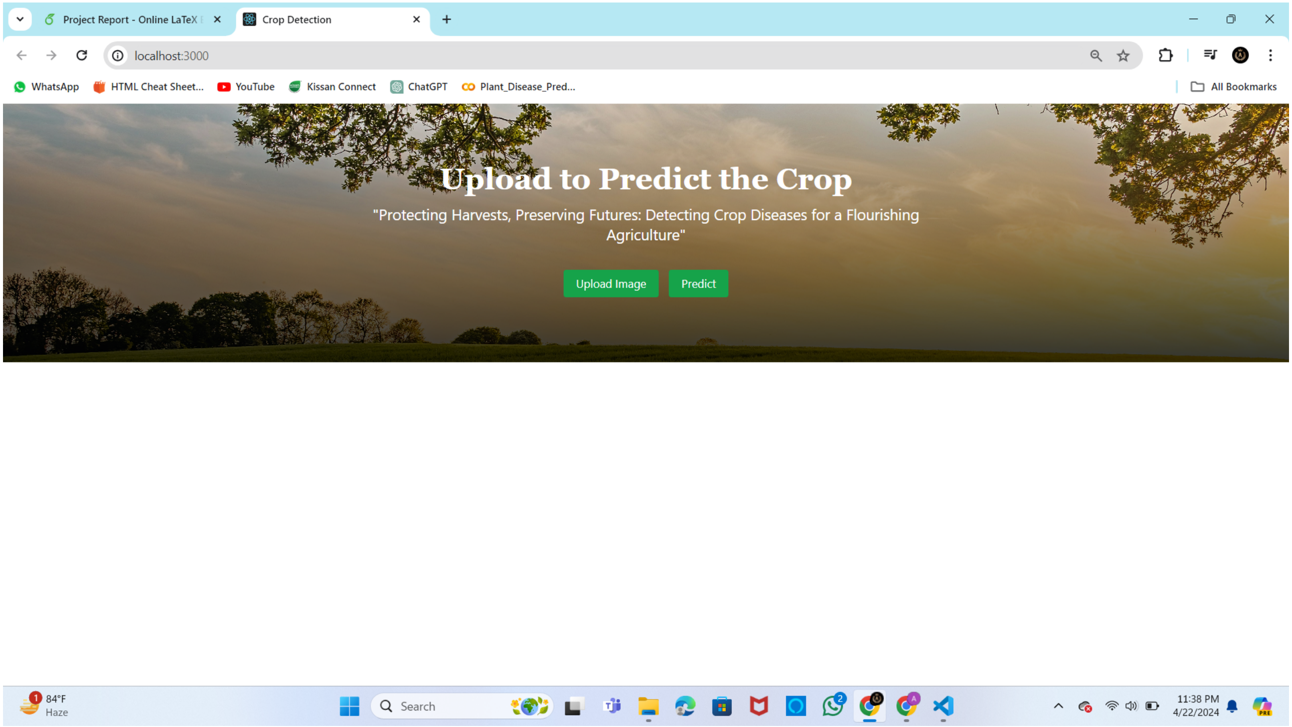
5.2 Code of GUI for Crop Prediction

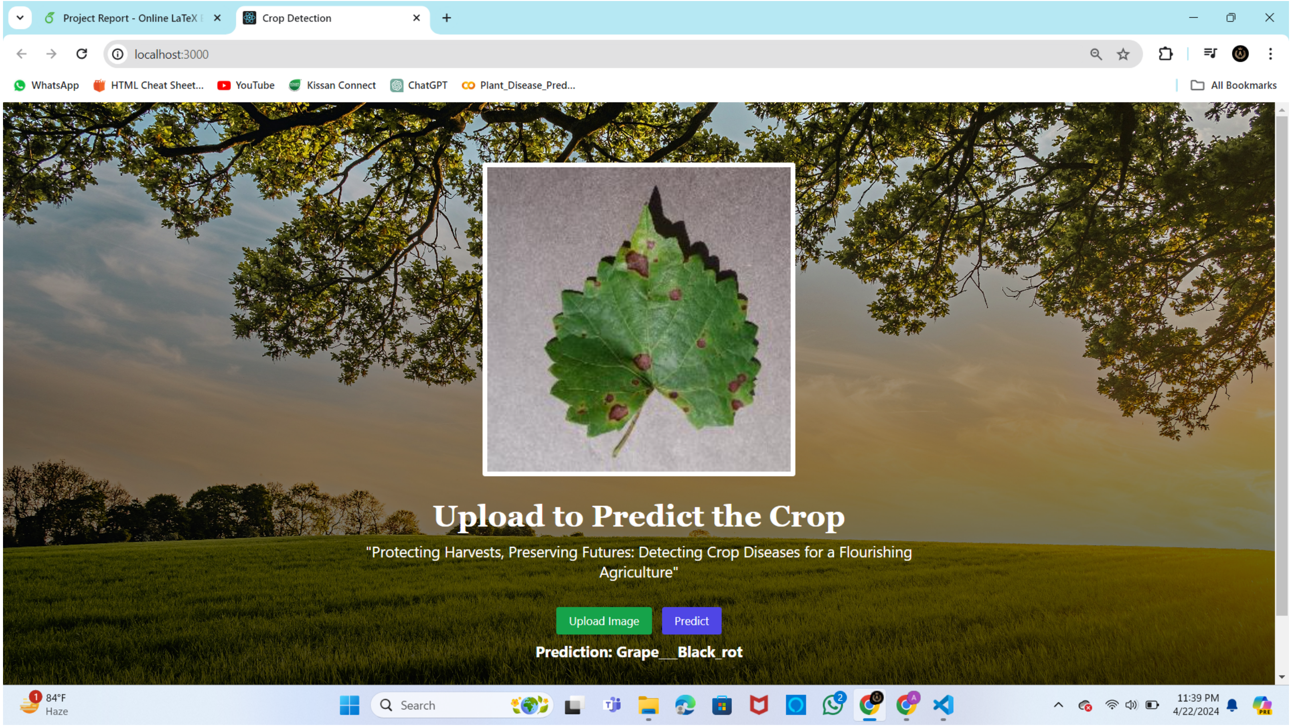


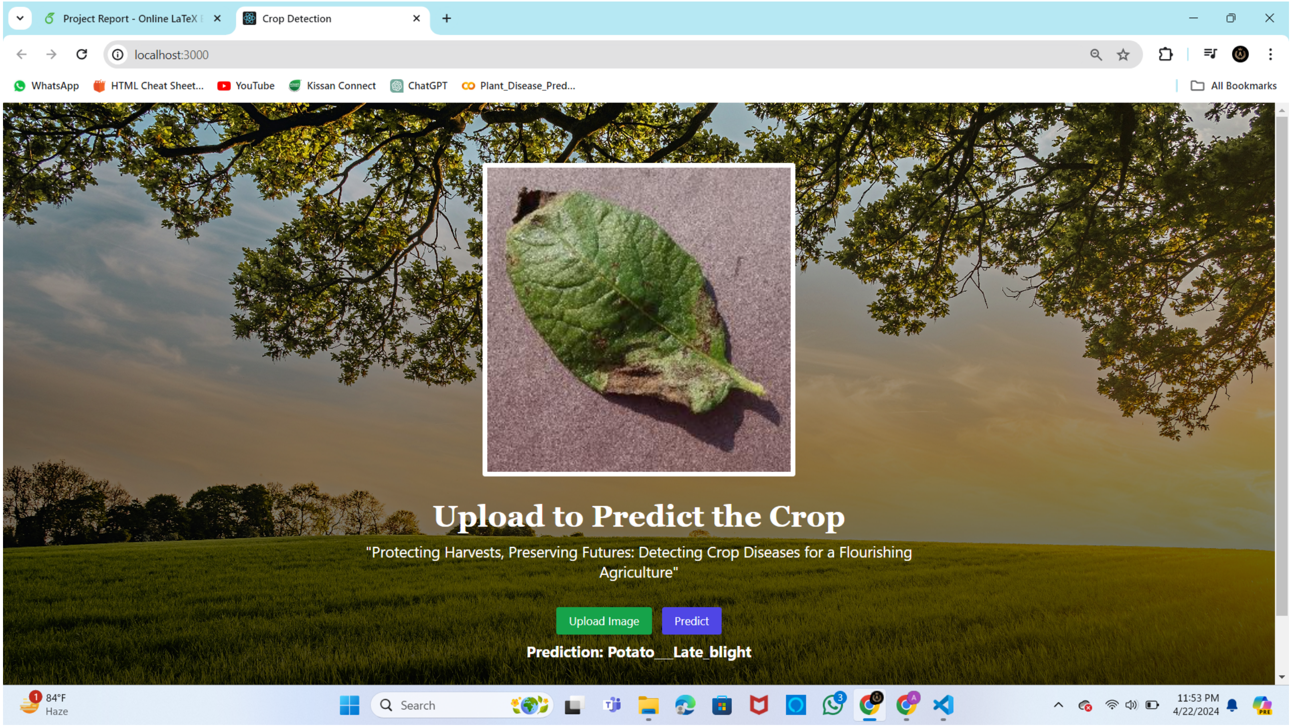


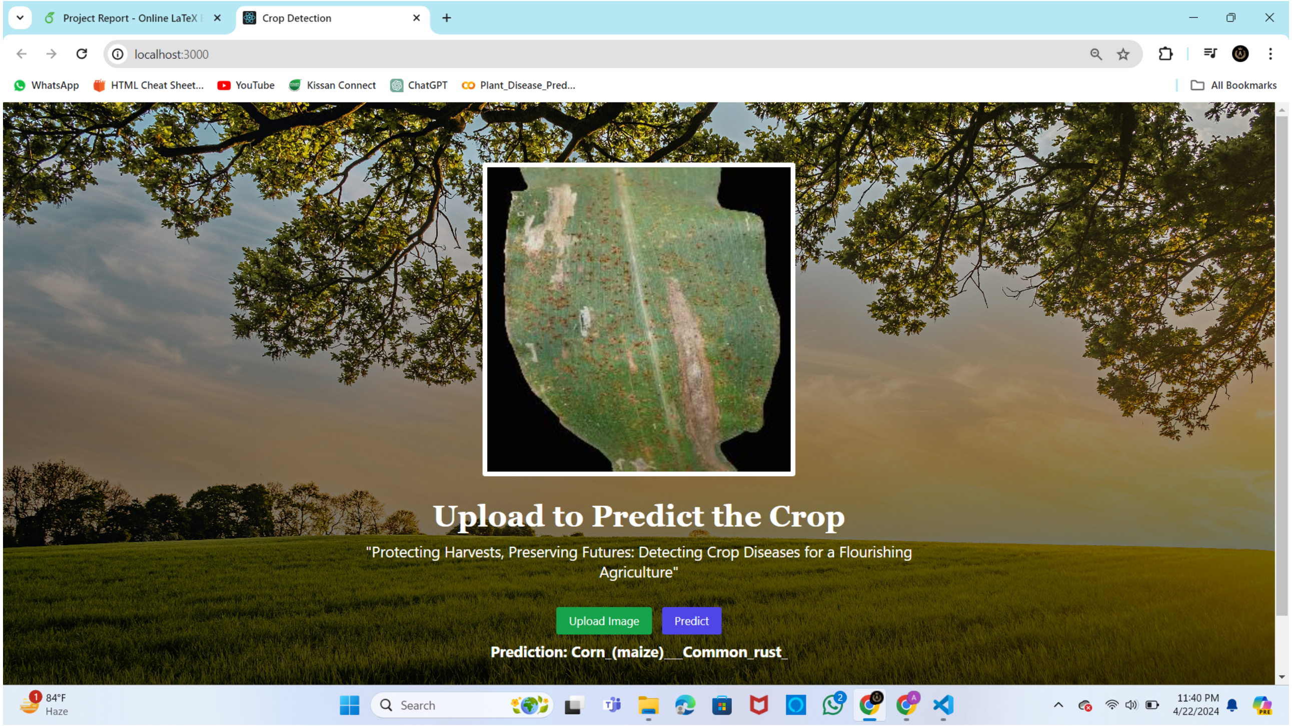
Chapter 6

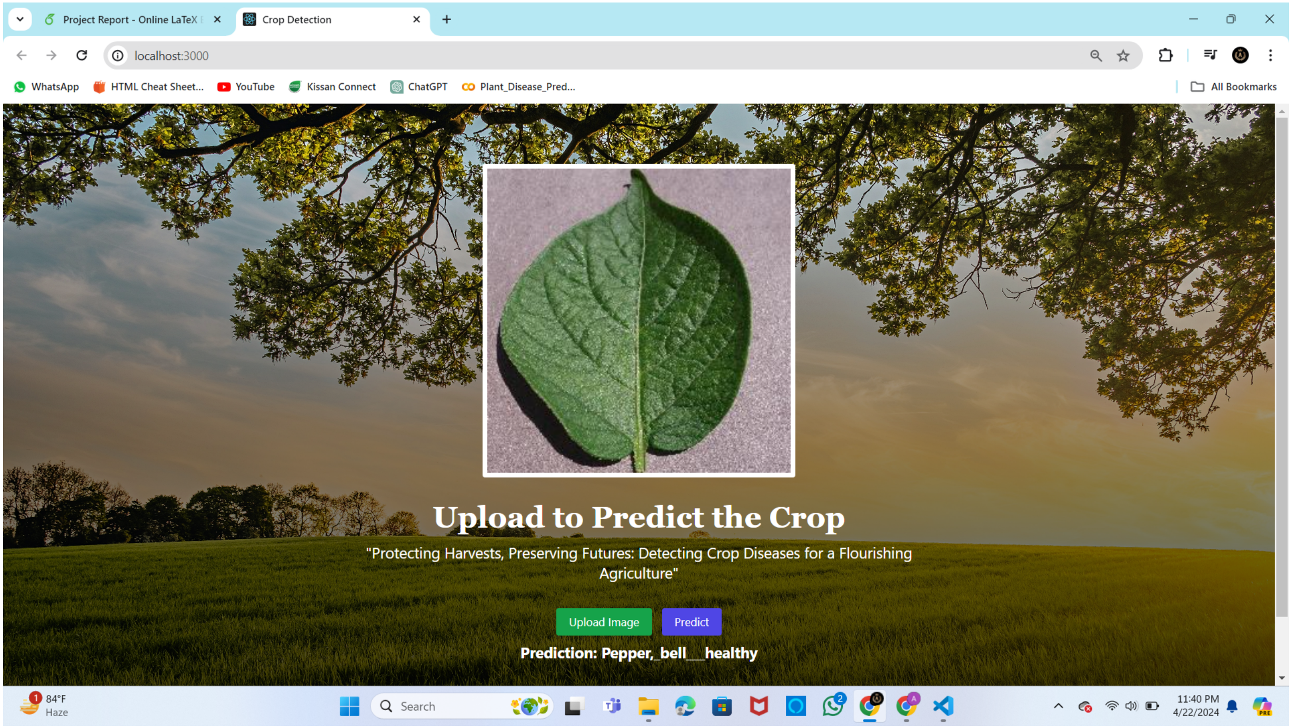
Screenshots

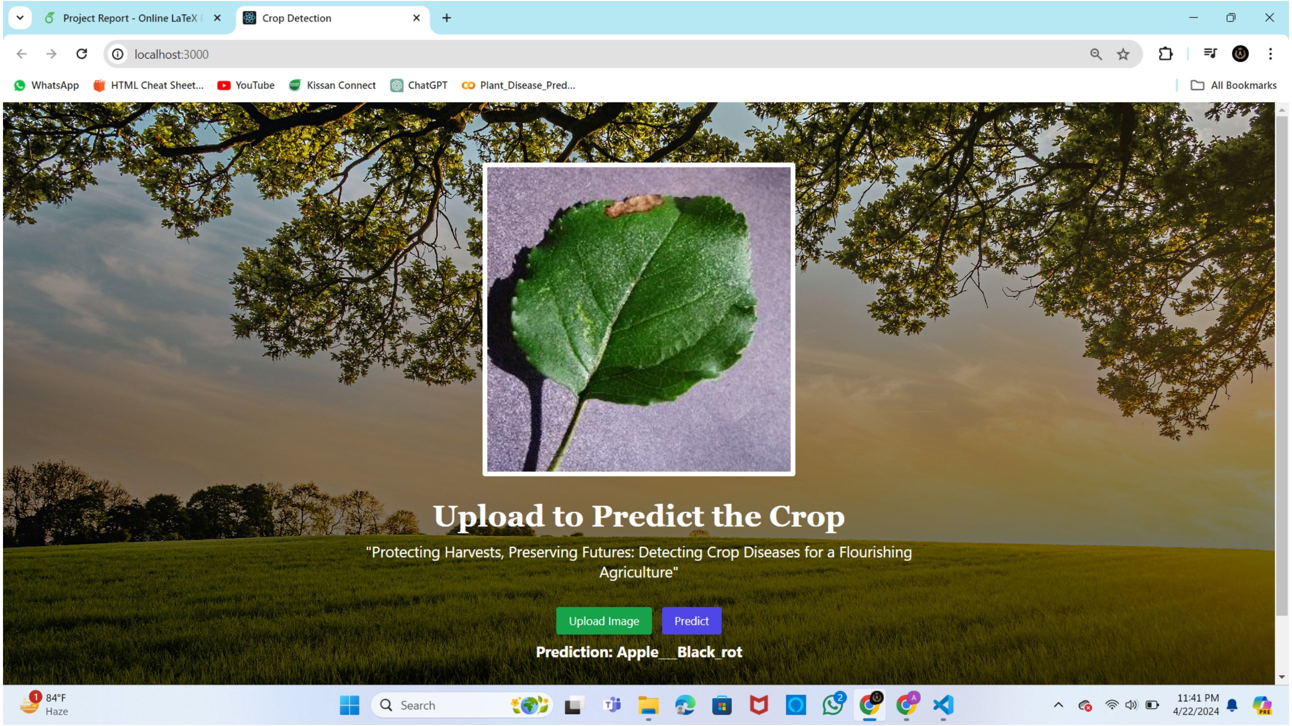


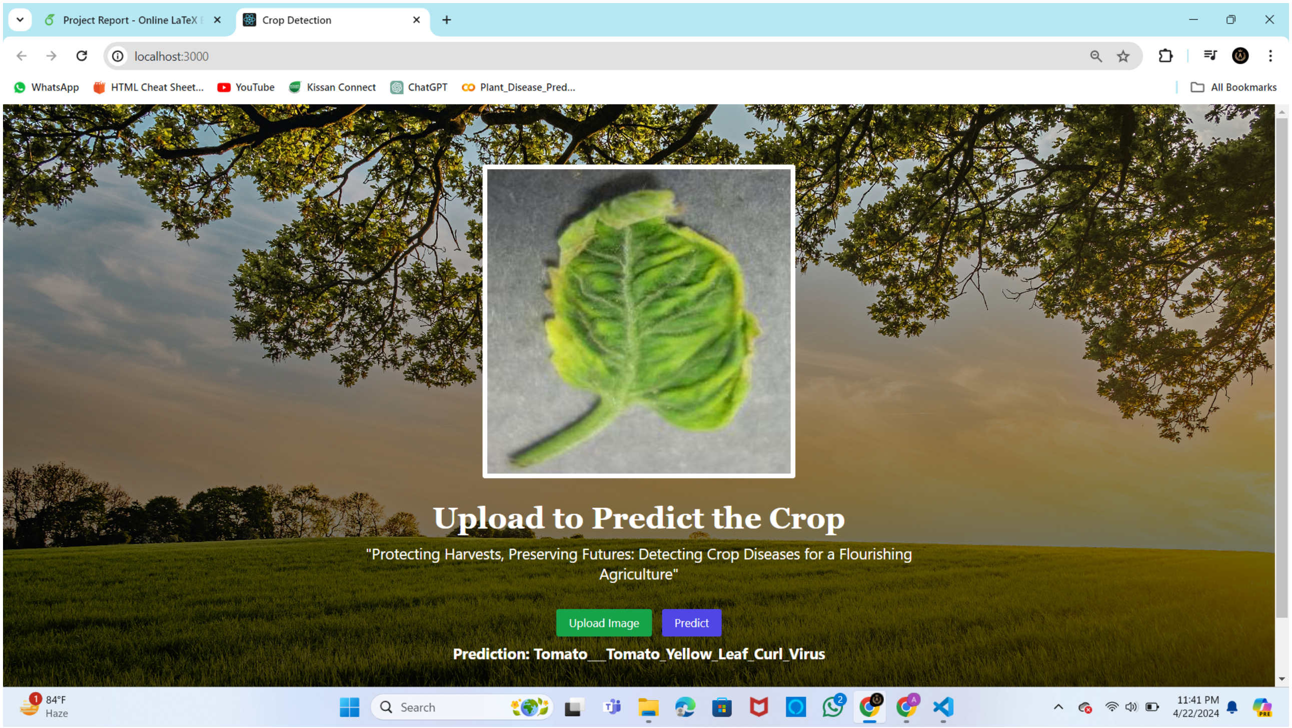

Project Report - Online LaTeX
Crop Detection
localhost:3000
WhatsApp
HTML Cheat Sheet...
YouTube
Kissan Connect
ChatGPT
Plant_Disease_Pred...
All Bookmarks
Upload to Predict the Crop
"Protecting Harvests, Preserving Futures: Detecting Crop Diseases for a Flourishing Agriculture"
Upload Image
Predict
Prediction: Grape___Black_rot
84°F
Haze
Search
11:39 PM
4/22/2024


Project Report - Online LaTeX
Crop Detection
localhost:3000
WhatsApp
HTML Cheat Sheet...
YouTube
Kissan Connect
ChatGPT
Plant_Disease_Pred...
All Bookmarks
Upload to Predict the Crop
"Protecting Harvests, Preserving Futures: Detecting Crop Diseases for a Flourishing Agriculture"
Upload Image
Predict
Prediction: Potato___Late_blight
84°F
Haze
Search
11:53 PM
4/22/2024


Project Report - Online LaTeX
Crop Detection
localhost:3000
WhatsApp
HTML Cheat Sheet...
YouTube
Kissan Connect
ChatGPT
Plant_Disease_Pred...
All Bookmarks
Upload to Predict the Crop
"Protecting Harvests, Preserving Futures: Detecting Crop Diseases for a Flourishing Agriculture"
Upload Image
Predict
Prediction: Corn_(maize)___Common_rust_
84°F
Haze
Search
11:40 PM
4/22/2024


Project Report - Online LaTeX
Crop Detection
localhost:3000
WhatsApp
HTML Cheat Sheet...
YouTube
Kissan Connect
ChatGPT
Plant_Disease_Pred...
All Bookmarks
Upload to Predict the Crop
"Protecting Harvests, Preserving Futures: Detecting Crop Diseases for a Flourishing Agriculture"
Upload Image
Predict
Prediction: Pepper,_bell___healthy
84°F
Haze
Search
11:40 PM
4/22/2024


Project Report - Online LaTeX
Crop Detection
localhost:3000
WhatsApp
HTML Cheat Sheet...
YouTube
Kissan Connect
ChatGPT
Plant_Disease_Pred...
All Bookmarks
Upload to Predict the Crop
"Protecting Harvests, Preserving Futures: Detecting Crop Diseases for a Flourishing Agriculture"
Upload Image
Predict
Prediction: Apple__Black_rot
84°F
Haze
Search
11:41 PM
4/22/2024


Project Report - Online LaTeX
Crop Detection
localhost:3000
WhatsApp
HTML Cheat Sheet...
YouTube
Kissan Connect
ChatGPT
Plant_Disease_Pred...
All Bookmarks
Upload to Predict the Crop
"Protecting Harvests, Preserving Futures: Detecting Crop Diseases for a Flourishing Agriculture"
Upload Image
Predict
Prediction: Tomato__Tomato_Yellow_Leaf_Curl_Virus
84°F
Haze
Search
11:41 PM
4/22/2024

Chapter 7

Conclusion and Future Scope

7.1 Conclusion

In conclusion, the predictive crop disease detection project represents a significant step towards enhancing agricultural practices through the integration of technology. By leveraging machine learning algorithms and image processing techniques, we have developed a model capable of detecting diseases in crops with promising accuracy.

Through this project, we have demonstrated the potential to revolutionize traditional crop management methods, enabling early detection of diseases and proactive intervention to mitigate crop losses. Our model's performance, as evaluated through rigorous testing and validation, underscores its utility in assisting farmers and agricultural stakeholders in making informed decisions about disease control measures.

7.2 Future Scope

The predictive model for crop disease detection has a broad potential scope, which bodes well for future developments and applications in the agricultural sector. First, continued research and development will concentrate on integrating cutting-edge technologies like edge computing, federated learning, and drone-based photography to improve the model's accuracy, resilience, and scalability. Thanks to these developments, the model will be able to assess datasets that are bigger and more varied, taking into account the intricacies of many crop kinds, environmental factors, and disease strains.

References

DeepCrop: Deep learning-based crop disease prediction with web application - Md. Manowarul Islam a, Md Abdul Ahad Adil (December 2023)

https://www.sciencedirect.com/science/article/pii/S2666154323002715?via

https://www.mdpi.com/2079-9292/12/1/229

https://www.youtube.com/live/lhufOy2W3Ps?si=huycTpctCQ9Iuti2

https://youtu.be/zfiSAzpy9NM?si=DPbDDxAhxht2qIiJ

https://youtube.com/playlist?list=PLeo1K3hjS3uvCeTYTeyfe0 rN5r8zn9rwsi=SsYzrXvwVUg1zzeD

https://www.sciencedirect.com/science/article/pii/S2589721722000204?via

https://ieeexplore.ieee.org/abstract/document/9137986

https://doi.org/10.3390/su15129321

https://www.mdpi.com/2079-9292/12/1/229

https://www.mdpi.com/2311-7524/9/2/149