Plant Disease Detection System For Sustainable Agriculture

A Project Report

submitted in partial fulfillment of the requirements

of

AICTE Internship on AI: Transformative Learning
with
TechSaksham – A joint CSR initiative of Microsoft & SAP

by

Arekante Uppara Vinod Kumar, uvinodkumar176@gmail.com

Under the Guidance of

P.Raja, Master Trainer, Edunet Foundation

ACKNOWLEDGEMENT

I would like to express my heartful gratitude to my mentor Dr.P.Raja , for his unwavering guidance , encouragement, and valuable insights throughout this project , Plant Disease Detection System For Sustainable Agriculture. His in depth knowledge and constant support have been instrumental in overcoming challenges and achieving the desired outcomes.

I am deeply appreciative of his patience and willing to clarify even the smallest doubts which helped me enhance my technical understanding and practical skills. His constructive feedback motivated me to strive for excellence and ensure the project met high standards.

I extend my sincere thank to the Edunet Foundation AICTE internship for providing this platform to apply theoretical knowledge to real world problems. This experience has greatly enriched my learning and prepared me for future challenges.

ABSTRACT

The **Plant Disease Detection System for Sustainable Agriculture** aims to address the critical challenge of timely and accurate detection of plant diseases, which significantly impact crop yield and quality. The project leverages advancements in deep learning to assist farmers in identifying diseases at an early stage, thereby promoting sustainable agricultural practices.

The primary objective of this system is to develop a user-friendly tool capable of identifying multiple plant diseases from leaf images with high accuracy. A convolutional neural network (CNN) model was trained on a comprehensive dataset of plant disease images. The system was implemented using TensorFlow and deployed via a Streamlit based web application for accessibility.

The methodology involved preprocessing the dataset, optimizing the model architecture, and evaluating its performance using metrics such as accuracy, precision, and recall. The trained model achieved an accuracy of over 90%, demonstrating its effectiveness in identifying a wide range of plant diseases, including but not limited to apple scab, bacterial spots, and tomato mosaic virus.

The web application allows users to upload leaf images for real-time predictions, providing instant feedback on the disease and recommended actions. This system empowers farmers with a reliable tool to prevent crop losses, reduce the need for excessive chemical usage, and support sustainable farming practices.

In conclusion, the project successfully integrates artificial intelligence and agriculture to promote food security and environmental sustainability. Future work will involve expanding the dataset, enhancing the model's robustness, and integrating additional features such as disease treatment suggestions.

TABLE OF CONTENT

LIST OF FIGURES

CHAPTER 1

Introduction

1.1Problem Statement:

Plant diseases significantly reduce crop yield and quality, threatening food security and farmers' livelihoods. Traditional detection methods are time-consuming, costly, and inaccessible to many, especially in rural areas. This project addresses the need for an efficient, AI-based solution to detect plant diseases early through leaf images. By providing accurate, real-time results, it promotes sustainable farming, reduces chemical usage, and minimizes crop losses.

1.2Motivation:

This project was chosen to address the growing challenge of plant disease management in agriculture, which directly impacts global food security and farmers' incomes. The lack of accessible and reliable diagnostic tools leads to delayed or inaccurate treatment, resulting in significant crop losses. By leveraging AI and deep learning, this project offers a practical solution for early disease detection, empowering farmers to take timely and informed actions.

The potential applications of this system include use in agricultural advisory services, integration into smart farming technologies, and deployment as a mobile or web application for farmers worldwide. Its impact lies in promoting sustainable farming, reducing dependency on harmful chemicals, and improving crop yields, ultimately supporting the livelihoods of farmers and contributing to environmental conservation.

1.3Objective:

- To develop an AI-based system capable of accurately detecting plant diseases from leaf images.
- To design a user-friendly web application for real-time disease prediction and feedback.
- To assist farmers in identifying plant diseases early, reducing crop losses and improving yield.
- To promote sustainable agricultural practices by minimizing the use of chemical treatments through precise disease diagnosis.
- To enhance food security by supporting effective and timely disease management strategies.

1.4Scope of the Project:

- **Disease Detection**: The system is trained to identify a wide range of plant diseases affecting crops like apples, tomatoes, corn, and others.
- **User Accessibility**: A web-based interface allows farmers to upload images and receive real-time disease predictions.
- **Sustainability Focus**: The project encourages sustainable agriculture by providing accurate disease diagnostics, reducing the need for chemical treatments.
- **Future Expansion**: The scope includes the potential to expand the dataset, improve model accuracy, and incorporate additional features, such as disease treatment suggestions.

CHAPTER 2

Literature Survey

The article discusses plant disease detection using deep convolutional neural networks (CNNs), focusing on their effectiveness in identifying various plant diseases from images. Existing models primarily use CNN-based architectures, but challenges remain regarding the scalability, dataset limitations, and real-time application performance. Gaps identified in current research include insufficient robustness across diverse plant species and environmental conditions. This project addresses these limitations by employing a more generalized and efficient CNN-based model to enhance the accuracy and applicability of plant disease detection systems in agriculture. [1]

Existing methods for plant disease detection often rely on deep learning techniques, with CNN-based models being the most common. PlantRefineDet is one such methodology that modifies the RefineDet model by integrating ResNet50 as its backbone for better accuracy. This model is trained on agricultural images containing 38 types of plant diseases to detect crop abnormalities. However, a gap remains in real-time, scalable applications and in handling diverse environmental factors.
This project addresses these gaps by improving model generalization and enabling real-time disease prediction.[2]

The article discusses machine learning applications in plant disease detection, with a focus on various techniques used for accurate classification and recognition. It highlights challenges like dataset diversity, real-time prediction capabilities, and model scalability. The paper emphasizes the need for models that can generalize across different plant species and environmental conditions. This project aims to address these challenges by improving dataset diversity and using deep learning for more robust, scalable, and real-time plant disease detection.[3]

The article from MDPI discusses the application of deep learning (DL) techniques for automatic and reliable leaf disease detection. The proposed methodologies focus on convolutional neural networks (CNNs) to improve the accuracy of plant disease classification. Existing models like ResNet and Inception networks are applied to large agricultural datasets for plant recognition. However, challenges like model generalization and dataset limitations still exist. Your project can address these gaps by employing a custom-trained model specific to plant diseases with fine-tuned parameters to enhance prediction accuracy.

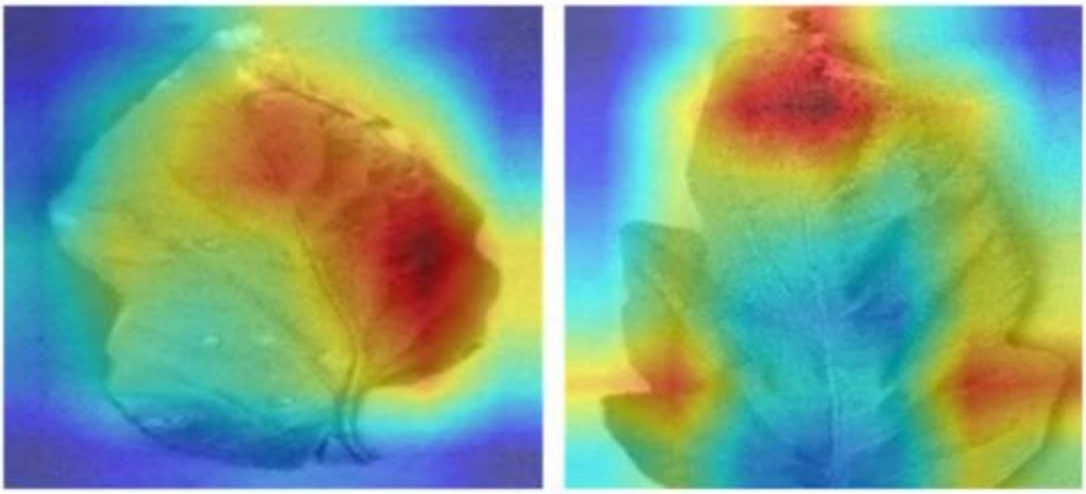

Figure 1. Score-CAM visualization of tomato leaf images, to show where the CNN model is mainly taking the decision.[4]

The paper explores deep learning models like VGG16, ResNet50, and GoogleNet for plant disease detection, using feature extraction methods like SVM and KNN for classification. The study investigates the application of transfer learning and feature extraction for plant disease identification, comparing the models' performance using various classifiers. The findings suggest that deep learning methods, particularly CNNs, offer high accuracy in plant disease detection, though computational complexity remains a challenge. The research aims to refine these models for practical, mobile-based solutions in agriculture.

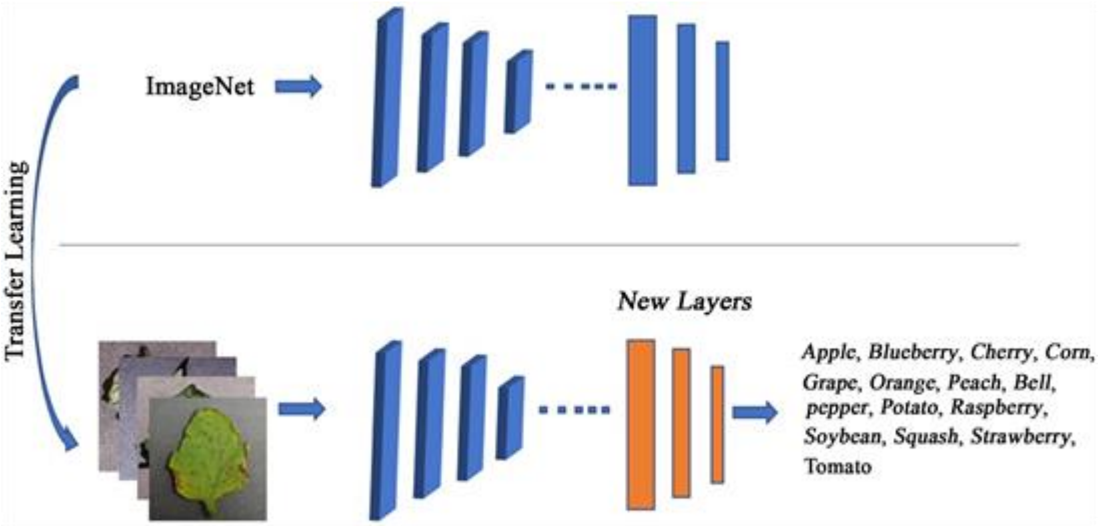

Figure 2. A brief representation of transfer learning with the data set image.

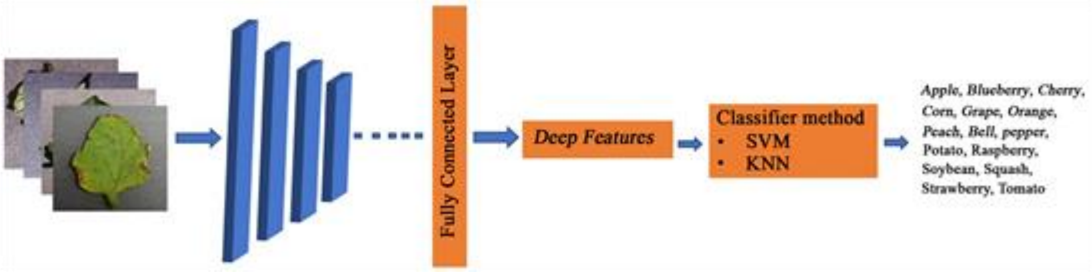

Figure 3. **A brief representation of feature extraction with the data set image**.[5].

The paper discusses the use of deep learning techniques, particularly CNNs, for detecting tomato leaf diseases. It outlines the importance of utilizing large datasets for training models and presents the effectiveness of CNN-based approaches in accurately identifying various leaf diseases. The paper also highlights challenges related to dataset diversity, model complexity, and real-time prediction. These challenges can be addressed by enhancing model efficiency and dataset quality, which this project aims to achieve for broader applicability in agricultural disease management.[6]

In the proposed methodology, machine learning (ML) algorithms are utilized for various tasks such as classification, clustering, and outlier detection. Deep learning, an advanced form of ML, is especially effective for image recognition tasks using Convolutional Neural Networks (CNN). CNNs excel in processing and identifying patterns in image datasets, which is crucial for applications like plant disease detection. This method enhances the model's ability to handle complex, high-dimensional data, providing efficient and accurate predictions.[7]

CHAPTER 3

Proposed Methodology

3.1 System Design

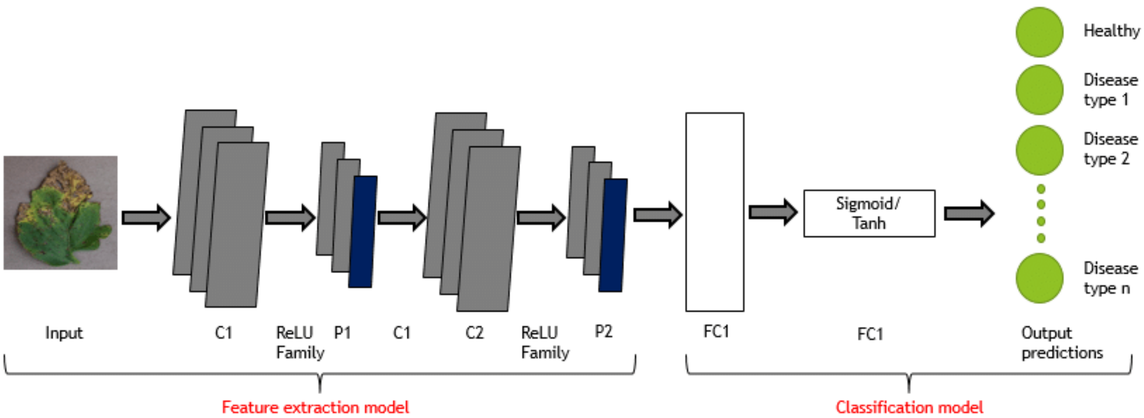

Figure 4: CNN Architecture

1. **Input Layer**: The image of the plant leaf is passed into the system.
2. **Convolutional Layers**: These layers extract features from the input image, identifying textures, edges, and patterns.
3. **Pooling Layers**: Reduce the dimensionality of the data, helping to minimize computation while retaining important features.
4. **Fully Connected Layers**: Process the extracted features for classification.
5. **Output Layer**: The model outputs the disease prediction based on the training data.

3.2 Requirement Specification

Tools and Technologies:

1. **Programming Language**: Python
2. **Web Framework**: Streamlit (for UI)
3. **Machine Learning Library**: TensorFlow (for CNN model)
4. **Image Processing**: OpenCV, Pillow
5. **Data Storage**: SQLite (for storing images and predictions)
6. **Model Deployment**: TensorFlow.js (if required for web-based prediction)
7. **Visualization**: Matplotlib, Plotly (optional for visual representation)
8. **Database**: MySQL or SQLite for managing data related to predictions

3.2.1 Hardware Requirements:

Device Requirements for this project execution

Processor: Intel i5/i7 or AMD Ryzen 5/7 (for model training and prediction)
RAM: Minimum 8 GB (preferably 16 GB for handling large datasets)
Graphics Card (GPU): NVIDIA GTX 1060 or higher (for efficient deep learning model training)
Storage: SSD with at least 100 GB (for data and model storage)
Webcam (optional): For real-time disease detection through live feed or upload from drive.These specifications will ensure smooth processing of both model training and real-time predictions.

3.2.2 Software Requirements:

Operating System: Windows, Linux, or macOS
Programming Language: Python 3.x
Libraries/Frameworks:
TensorFlow or Keras (for model development)
Streamlit (for web-based application)
OpenCV, Pillow (for image processing)
Scikit-learn (for machine learning algorithms, if needed)
SQLite or MySQL (for database management)
IDE/Editor: VS Code, PyCharm, or Jupyter Notebook
Web Browser: Chrome or Firefox (for accessing the application)

CHAPTER 4

Implementation and Result

4.1 Snap Shots of Result:

```
'alidation_set = tf.keras.utils.image_dataset_from_directory(
    r'C:\Users\vinnu\Desktop\aicte internship project\Plant Disease Detection System for Sustainable Agriculture\Dataset\valid
    labels="inferred",
    label_mode="categorical",
    class_names=None,
    color_mode="rgb",
    batch_size=32,
    image_size=(128, 128),
    shuffle=True,
    seed=None,
    validation_split=None,
    subset=None,
    interpolation="bilinear",
    follow_links=False,
    crop_to_aspect_ratio=False
]
:lass_name = validation_set.class_names
'rint(class_name)
```

✓ 1.3s Python

```
Found 17572 files belonging to 38 classes.
['Apple___Apple_scab', 'Apple___Black_rot', 'Apple___Cedar_apple_rust', 'Apple___healthy', 'Blueberry___healthy', 'Cherry_(includ
```

Figure 5 :loading all image files

```python
#Test Image Visualization
import cv2
image_path = 'test/TomatoYellowCurlVirus6.JPG'
# Reading an image in default mode
img = cv2.imread(image_path)
img = cv2.cvtColor(img,cv2.COLOR_BGR2RGB) #Converting BGR to RGB
# Displaying the image
plt.imshow(img)
plt.title('Test Image')
plt.xticks([])
plt.yticks([])
plt.show()
```

✓ 1.5s

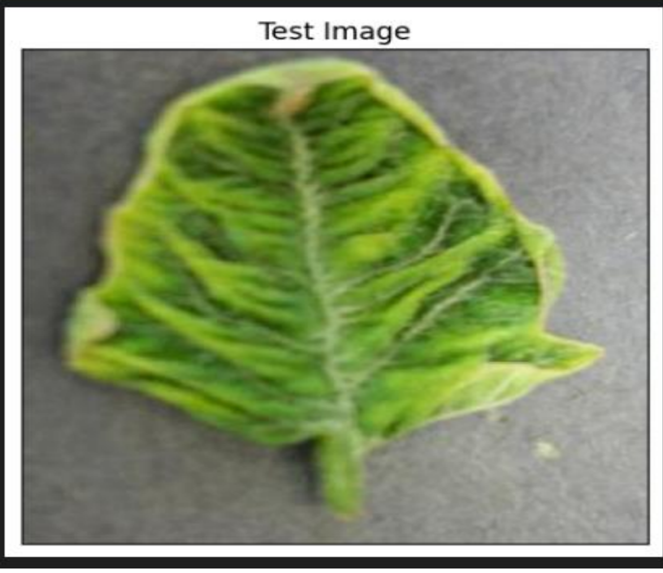

Figure 5: Passing test Image

```python
print(predictions)
```

[6] ✓ 0.0s

```
[[8.4503556e-23 2.3018767e-23 9.3245738e-18 9.5689173e-19 1.3770097e-18
  4.7074018e-17 8.5684178e-22 1.7727106e-23 1.9865234e-23 1.3441291e-22
  6.6417300e-21 4.3285778e-18 3.4048929e-20 2.7505210e-17 4.8590451e-21
  8.7032778e-16 9.2940487e-22 5.9445895e-24 5.6777995e-17 3.2420183e-17
  2.5853845e-24 3.5388826e-19 1.2271234e-17 1.0794362e-15 6.1457658e-19
  6.2441734e-17 1.6250220e-19 1.4235588e-17 4.8949587e-14 3.0498251e-14
  3.6615318e-16 5.9607049e-19 3.9399862e-16 5.1077537e-16 3.8462492e-17
  1.0000000e+00 3.7863733e-17 6.0154605e-18]]
```

Figure 7: Prediction Values

```python
# Displaying the disease prediction
model_prediction = class_name[result_index]
plt.imshow(img)
plt.title(f"Disease Name: {model_prediction}")
plt.xticks([])
plt.yticks([])
plt.show()
```

✓ 0.0s

Figue 8: Predicted Image Disease Type

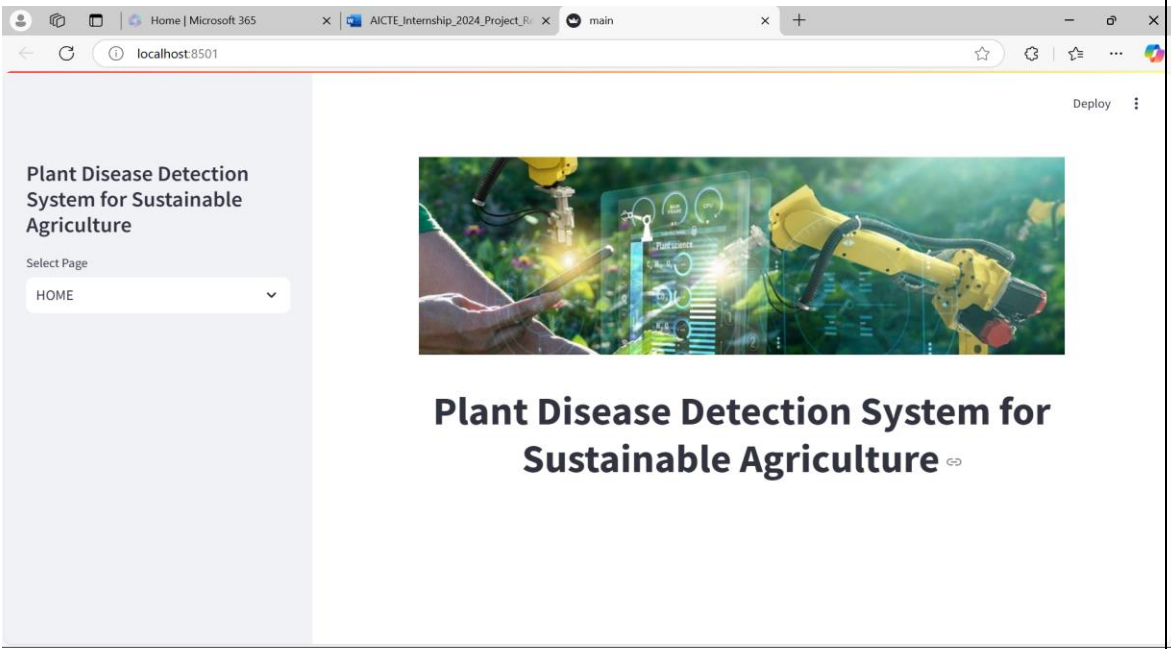

Figure 9: streamlit web home page of Plant Disease Detection System for Sustainable Agriculture

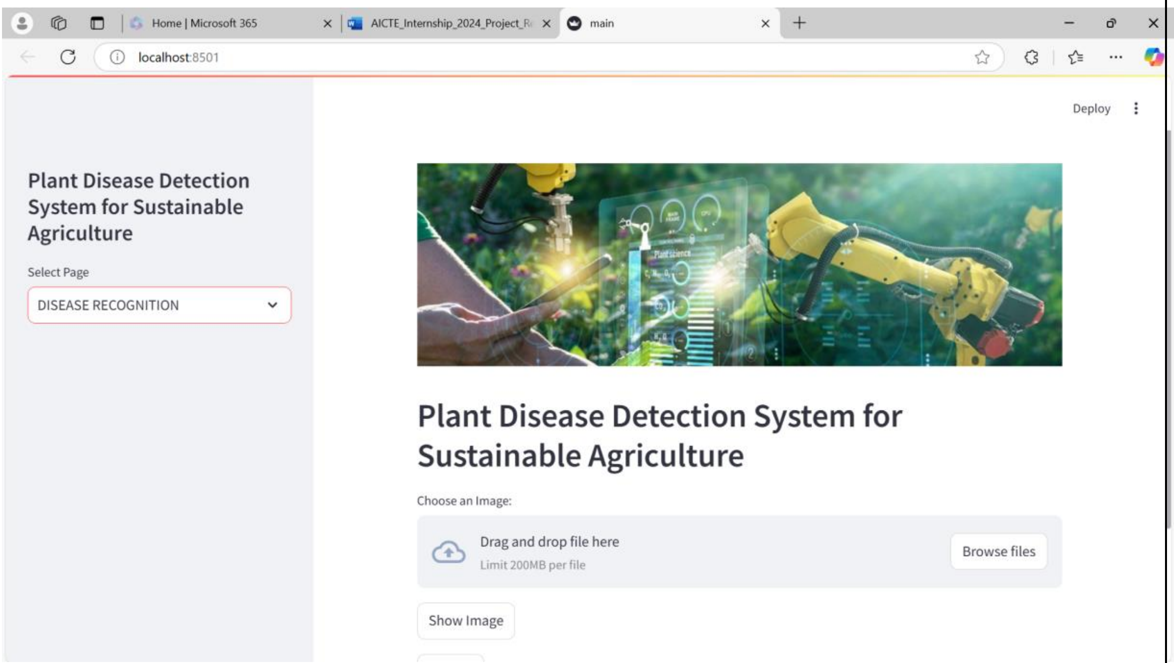

Figure 10: Disease Recognition Page and Browse test image

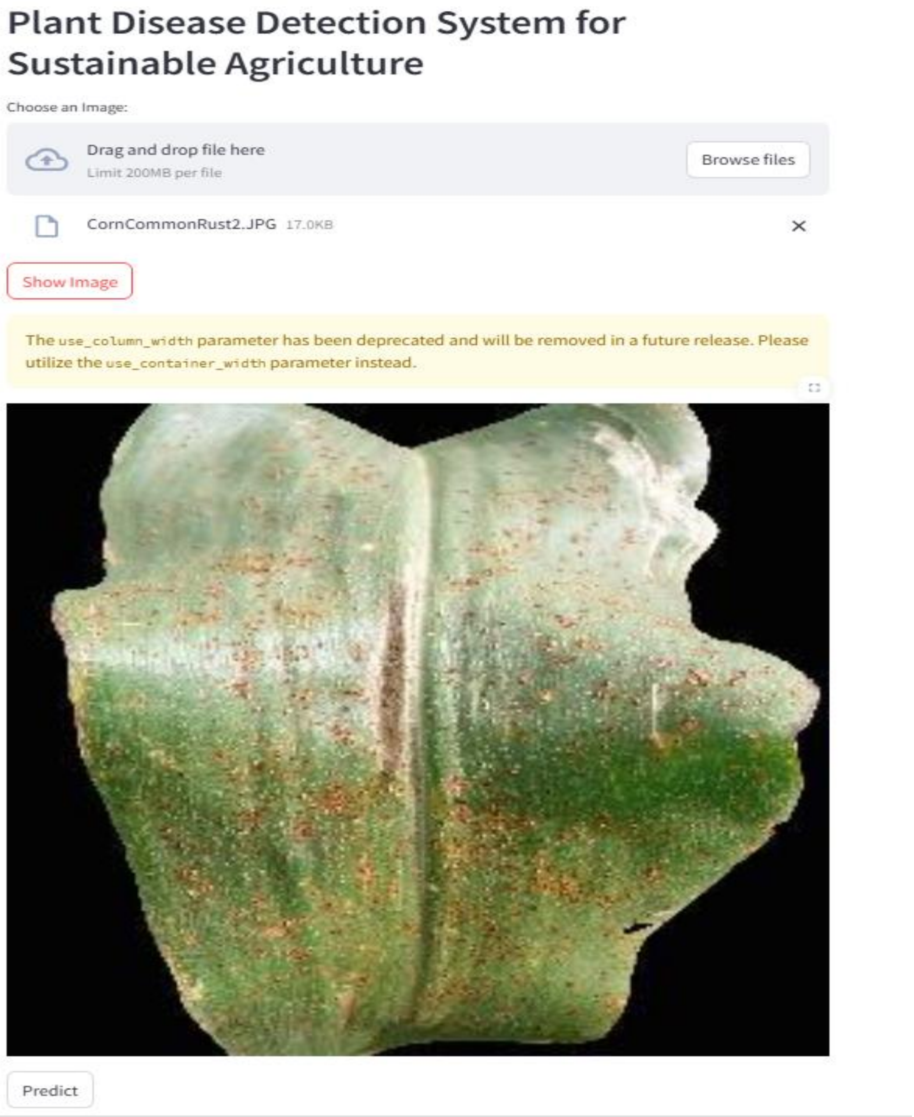

Figure 11: Browsed Image

Figure 12 : Displaying Predicted disease name

4.2 GitHub Link for Code:

https://github.com/Vinod9676/AICTE-Internship_Project

CHAPTER 5

Discussion and Conclusion

5.1 Future Work:

- **Enhance Model Accuracy**: Experiment with advanced architectures like EfficientNet or Vision Transformers for better performance.
- **Real-Time Detection**: Implement live detection using mobile or edge devices for practical use in fields.
- **Expand Dataset**: Incorporate more diverse plant diseases to improve model generalization.
- **Explainable AI**: Develop methods to explain the model's predictions, aiding farmers in understanding the results.
- **Integrate IoT**: Combine with IoT devices to monitor plant health continuously and automate disease prevention actions.

5.2 Conclusion:

The "Plant Disease Detection System for Sustainable Agriculture" successfully addresses the challenge of identifying plant diseases early, aiding in reducing crop losses and improving agricultural productivity. By leveraging deep learning and image recognition, the system provides an efficient, user-friendly solution for farmers and agricultural experts. This project contributes to sustainable farming practices and highlights the potential of AI in revolutionizing agriculture. Future enhancements can further expand its scope, making it a vital tool for global food security and sustainable development.

REFERENCES

[1] Pandian, J. A., Kumar, V. D., Geman, O., Hnatiuc, M., Arif, M., & Kanchanadevi, K. (2022). Plant disease detection using deep convolutional neural network. *Applied Sciences*, *12*(14), 6982.

[2] Alqahtani, Y., Nawaz, M., Nazir, T., Javed, A., Jeribi, F., & Tahir, A. (2023). An improved deep learning approach for localization and recognition of plant leaf diseases. *Expert Systems with Applications*, *230*, 120717.

[3] Ramesh, S., Hebbar, R., Niveditha, M., Pooja, R., Shashank, N., & Vinod, P. V. (2018, April). Plant disease detection using machine learning. In *2018 International conference on design innovations for 3Cs compute communicate control (ICDI3C)* (pp. 41-45). IEEE.

[4] Chowdhury, M. E., Rahman, T., Khandakar, A., Ayari, M. A., Khan, A. U., Khan, M. S., ... & Ali, S. H. M. (2021). Automatic and reliable leaf disease detection using deep learning techniques. *AgriEngineering*, *3*(2), 294-312.

[5] Mohameth, F., Bingcai, C., & Sada, K. A. (2020). Plant disease detection with deep learning and feature extraction using plant village. *Journal of Computer and Communications*, *8*(6), 10-22.

[6] Ashok, S., Kishore, G., Rajesh, V., Suchitra, S., Sophia, S. G., & Pavithra, B. (2020, June). Tomato leaf disease detection using deep learning techniques. In *2020 5th International Conference on Communication and Electronics Systems (ICCES)* (pp. 979-983). IEEE.

[7] Sujatha, R., Chatterjee, J. M., Jhanjhi, N. Z., & Brohi, S. N. (2021). Performance of deep learning vs machine learning in plant leaf disease detection. *Microprocessors and Microsystems*, *80*, 103615.