**CSE 499A (Section 4)**

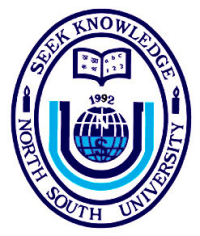
**Final Report (CO5)**

**Project Title: Plant Diseases Detection Using Image Processing**

**Submitted To**

**Dr. Shazzad Hosain (SZZ)**

**Date: 05/12/2024**



**Semester: Summer 2024**

**Group No: G-3**

**Group Members**

|  |  |
| --- | --- |
| **ID** | **Name** |
| 2012464042 | Abroor Zahin Niloy |
| 2011892042 | Akib Hasan |
| 2012959042 | Naima Homaira Khan |
| 2011566042 | Tania Akter Lima |

Chapter 1 – Introduction

Bangladesh is a country that predominantly relies on agriculture, with a significant portion of its population engaged in farming. Despite its fertile lands and favorable climate, the agricultural sector faces numerous challenges, including plant diseases that threaten crop yield and food security. Plant diseases significantly affect agricultural productivity, leading to substantial economic losses. Early detection of plant diseases is crucial for mitigating these losses. In this project, we develop an automated plant disease detection system based on deep learning, using a convolutional neural network (CNN). The system is integrated into a web application, allowing users to upload images of plant leaves, receive predictions of possible diseases, and access treatment recommendations.

The manual identification of plant diseases by farmers and experts is time-consuming, error-prone, and often requires expert knowledge. This project addresses these issues by automating the disease detection process, providing an accessible and efficient tool for anyone with a smartphone or computer. We have a selected certain objectives for this project. The objectives of this project are:

* To design and implement a CNN model capable of classifying plant diseases based on leaf images.
* To develop a web-based application with user authentication (login and registration).
* To integrate the trained CNN model with the web interface, allowing users to upload images and receive predictions.
* To analyze the system's performance and identify areas for improvement.

This project focuses on the classification of plant diseases from images of leaves. The system predicts from 38 disease categories and displays the prediction results along with suggestions for treatment.

Chapter 2 – System Design

The system is composed of two main components: the backend, which contains the CNN model and handles data processing, and the frontend, which is the user interface where users interact with the system. The architecture is as follows:

**Frontend (User Interface):**

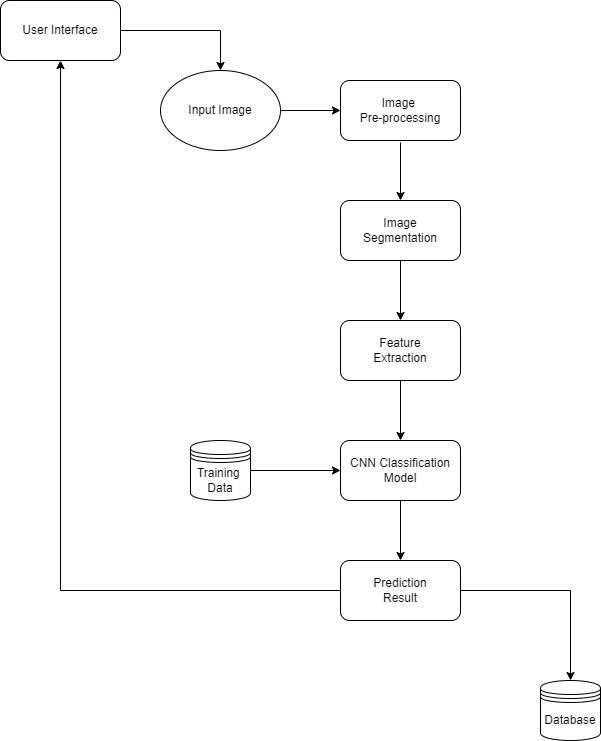
* **Login Page**: Users can log in to access the disease detection system. New users can register by providing their email, name, and password.
* **Registration Page**: This page allows users to create a new account.
* **Home Page**: Displays an overview of the platform and provides access to disease detection functionality.
* **Disease Detection Page**: Allows users to upload images of plant leaves for disease classification. The page displays the results after processing the image.

**Backend (Model and Processing):**

* **Image Preprocessing**: The uploaded image is resized to 128x128 pixels and normalized before being passed to the CNN model.
* **CNN Model**: A deep learning model that processes the image and classifies it into one of 38 possible disease categories.
* **Prediction Output**: The model returns the predicted disease along with a confidence score, which is displayed to the user on the web interface.

**Database:**

* **Users**: Stores user credentials, including email and hashed password.
* **Images**: Stores the paths and metadata of uploaded images.
* **Predictions**: Stores the results of disease predictions, including the predicted class and confidence score.



System Design Diagram

Chapter 3 – Impacts and Constraints

**Impacts:** Building a disease detection system can have both negative and positive impact.

* **Positive Impacts**:
  + **Early Disease Detection**: The system allows for quicker and more accurate disease detection, helping farmers take preventive measures before the disease spreads.
  + **Accessibility**: The web-based interface ensures that farmers can easily access the tool on both desktops and mobile devices.
  + **Cost Efficiency**: By automating the disease detection process, the need for expert consultations is reduced, lowering operational costs.
* **Negative Impacts**:
  + **Data Dependency**: The accuracy of the model depends heavily on the quality and diversity of the dataset. Limited data or poorly labeled images may result in less accurate predictions.
  + **Internet Access**: The system requires an internet connection to access the web-based platform, which may not be feasible in rural or remote areas.

### Constraints

* **Model Limitations**: The model is trained on a fixed dataset, and its performance may degrade if exposed to new or unseen disease types.
* **Data Quality**: The system’s accuracy depends on the quality of the input images. Low-quality images may lead to inaccurate predictions.
* **Computational Resources**: Running deep learning models requires significant computational power, which may be challenging for resource-limited environments.
* **Internet Connectivity**: The web-based platform requires a stable internet connection to function properly, limiting access in rural or remote areas.

Chapter 4 – Methodologies

**Data Collection and Preprocessing**

The dataset used consists of images of plant leaves, each labeled with one of 38 disease categories. The images were collected from publicly available sources such as Kaggle.

* **Image Preprocessing**: The images are resized to 128x128 pixels and normalized to values between 0 and 1 to make them suitable for model input.

### Model Development

The model is built using a convolutional neural network (CNN), a powerful deep learning architecture for image classification. The CNN model used in this project has the following layers:

1. **Convolutional Layers**: Extracts features from images using filters.
2. **Max-Pooling Layers**: Reduces the spatial dimensions of feature maps, improving efficiency and reducing the risk of overfitting.
3. **Dropout Layers**: Prevents overfitting by randomly disabling some neurons during training.
4. **Fully Connected Layers**: After flattening the features, fully connected layers are used to classify the image into one of 38 disease categories.

### Evaluation Metrics

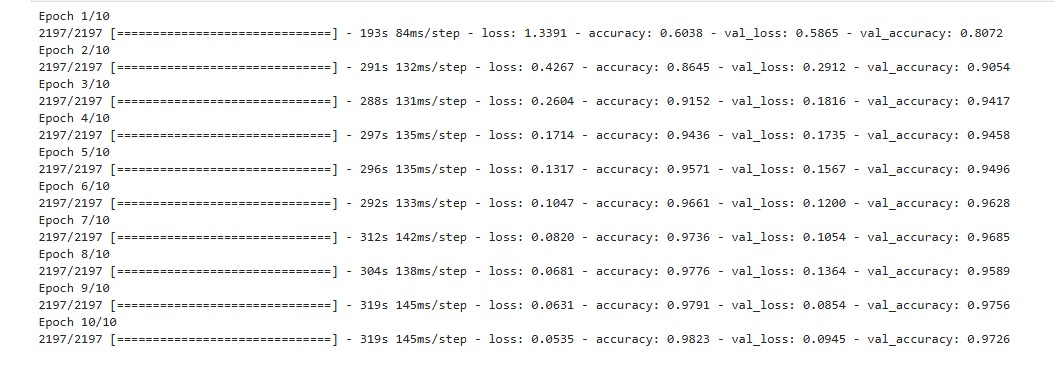
The model is evaluated using several metrics:

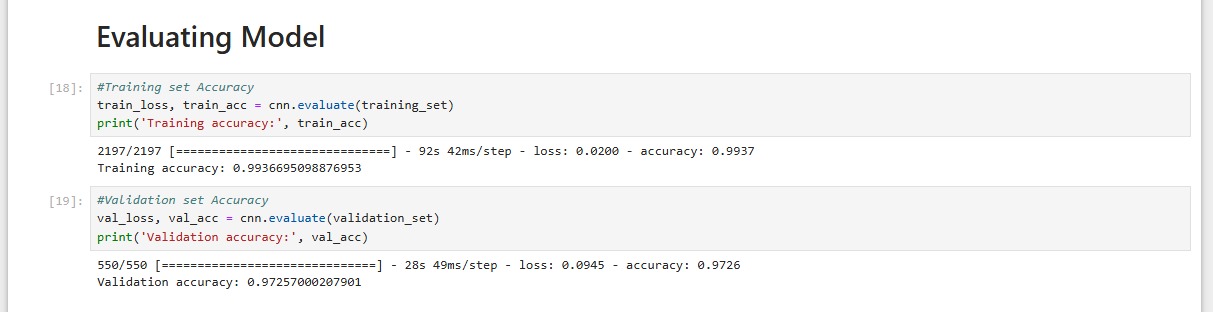
1. **Accuracy**: Measures the percentage of correct predictions.
2. **Precision, Recall, and F1-Score**: Provide insights into the model’s performance for each disease class.
3. **Confusion Matrix**: Helps visualize the performance of the model across all disease categories.

Chapter 5 – Result

**Training and Validation Results**

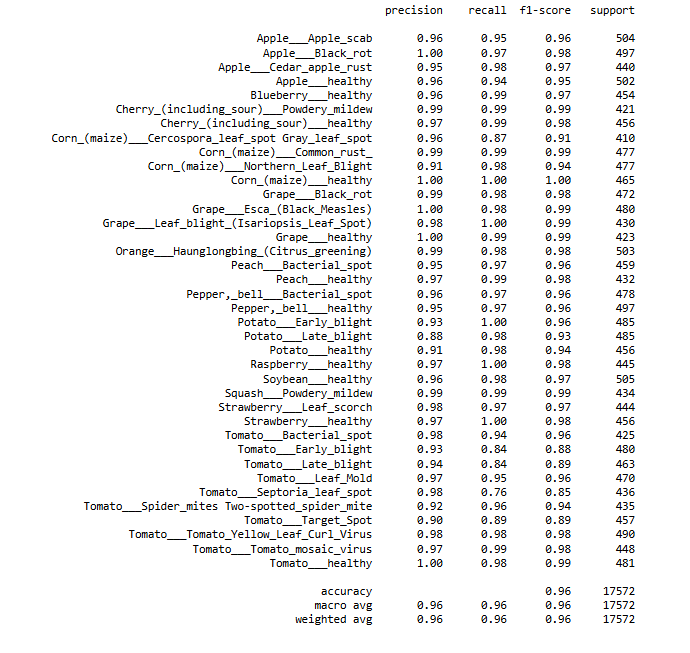
The model was trained for 10 epochs. The final training accuracy was 99.37% and the validation accuracy was 97.26% .The model’s loss decreased steadily throughout the training process, indicating that it was learning effectively.





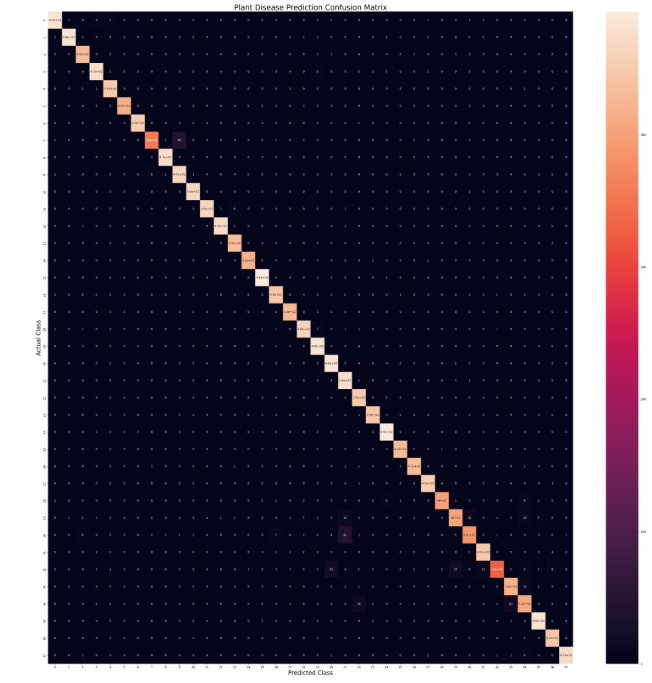
**Evaluation Metrics**

* **Precision**: The model achieved a precision of approximately 0.96 for most classes.
* **Recall**: The recall was also high, particularly for classes with a larger number of samples.
* **F1-Score**: The F1-scores were balanced, indicating that the model was both accurate and reliable.



**Model Confusion Matrix**

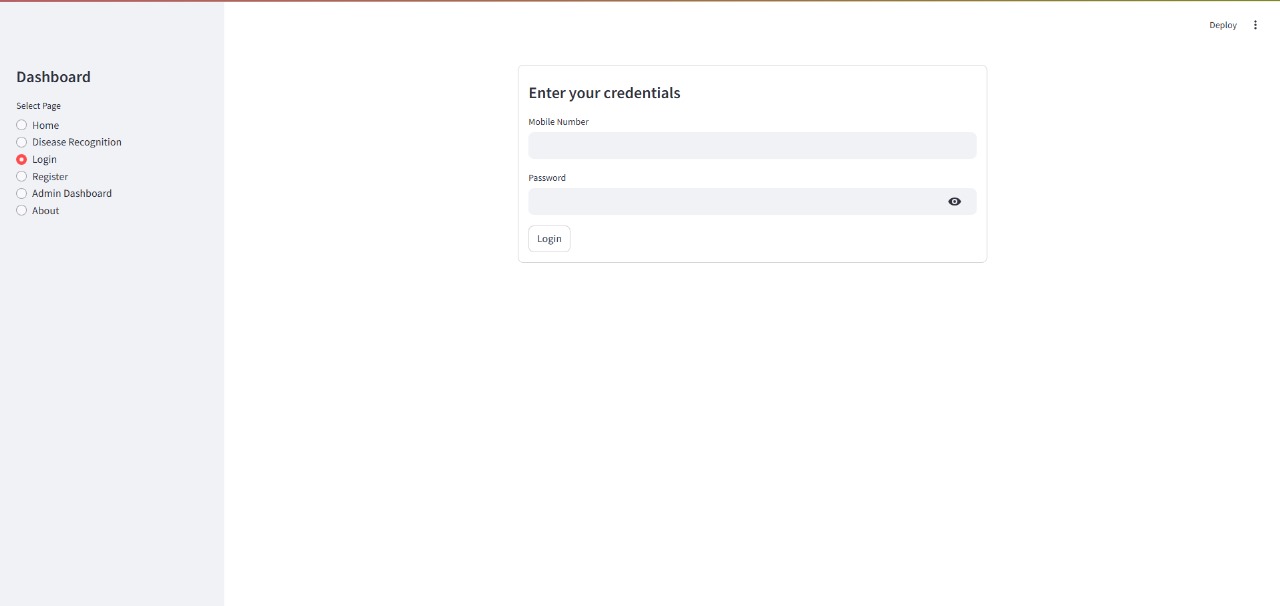
The confusion matrix was used to evaluate the performance of the model across all 38 classes. The heatmap revealed that some disease classes were misclassified more frequently than others, particularly those with fewer samples in the dataset.



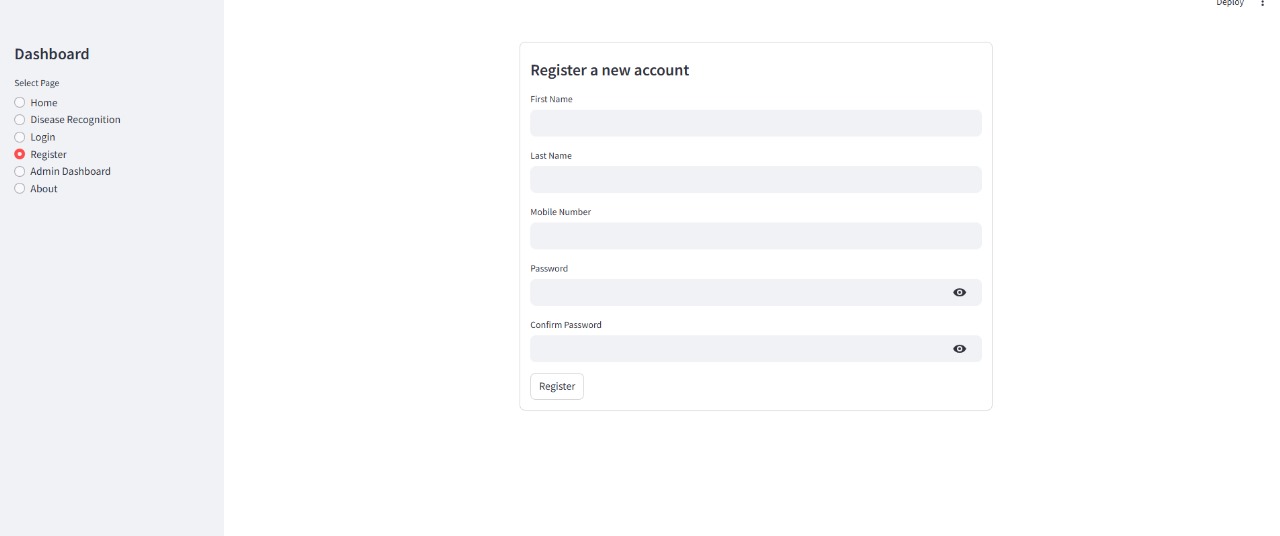
**Web Interface Performance**

The web application worked as intended. Users could successfully log in, register, upload images, and receive predictions. The system was responsive and provided results within a few

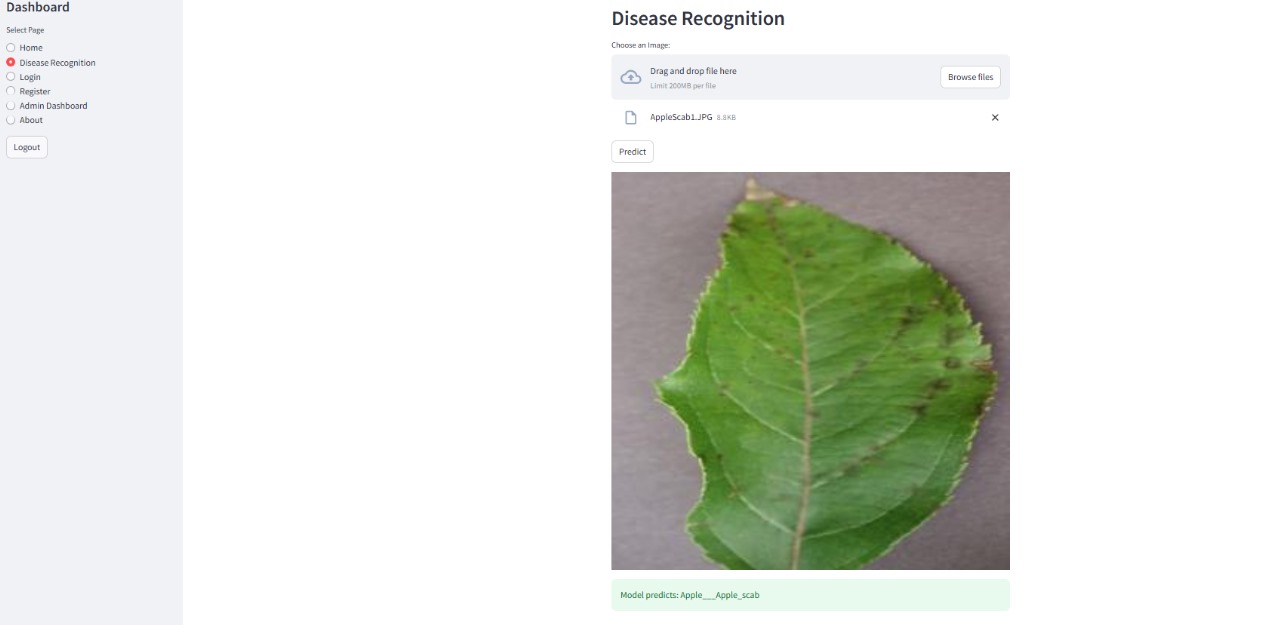
seconds.



Login Page



Registration Page



Disease Recognition Page

Chapter 6 – Conclusion

In conclusion, this project demonstrates the potential of machine learning in addressing critical challenges in agriculture, particularly plant disease detection. By leveraging advanced algorithms and image processing techniques, the proposed system provides an efficient, accurate, and scalable solution for identifying plant diseases at an early stage. This not only helps reduce economic losses but also contributes to sustainable agricultural practices by enabling timely intervention and resource optimization.

This project successfully developed a plant disease detection system using deep learning, integrated with a user-friendly web interface. The CNN model achieved high accuracy in both training and validation, making it a reliable tool for detecting plant diseases from leaf images.

Now if we want to talk about contributions we’ve made a few. For example:

* Developed a deep learning model capable of identifying 38 different plant diseases.
* Created a fully functional web interface that allows users to upload images and view predictions.
* Evaluated the system’s performance and identified areas for improvement.

This project highlights the intersection of technology and agriculture, paving the way for innovative solutions to empower farmers and ensure food security in the future.