Deep Learning-Based Plant Stress Detection Using Leaf Image Analysis with Enhanced Feature Extraction

Dr. N. Mythili
Associate Professor
Computer Science and Engineering
St. Joseph's Institute of Technology
Chennai, Tamil Nadu, India
mythilin@stjosephstechnology.ac.in

Ranjitha Prabha P
Computer Science and Engineering
St. Joseph's Institute of Technology
Chennai, Tamil Nadu, India
ranjithaprabhap11pais@gmail.com

Monisha J R
Computer Science and Engineering
St. Joseph's Institute of Technology
Chennai, Tamil Nadu, India
monishajr2212@gmail.com

The stress on the plant due to diseases, nutrient deficiency, or environmental conditions is a significant factor in the growth of the plant. It is necessary to understand the stress factors at an early stage so that the growth of the plant is not adversely affected. In this paper, a system is developed to analyze the images of the plant leaves and check whether the plant is healthy or stressed. The system will analyze the image and check the characteristics of the plant leaf based on the input image. The system is made efficient and fast so that the output is generated within a short time. The experimental results show that the system is highly accurate with a prediction time of less than two seconds. This method is useful for the farmers or the scientist.

Keywords : Plant Stress Detection, Leaf Image Analysis, Convolutional Neural Network, Image Classification, Agriculture, Crop Monitoring

I. INTRODUCTION

The agriculture industry plays a major role in the economic success of countries. However, the health of plants may be compromised through many types of stresses such as diseases, pests, mineral deficiencies, and environmental changes. All these stresses negatively impact the production of crops as well as food safety.

Traditionally, there have been many challenges involved in identifying stresses in plants because this process was done by humans either by the farmer himself or by experts. Not only is this process tedious, but it also requires a vast amount of knowledge regarding agriculture. Moreover, the effectiveness of the results may be influenced by biases of human beings. Therefore, an automated plant stress detector is needed.

Today, image processing techniques and machine learning have advanced to an extent whereby the classification of plant leaf images is possible. Machine learning algorithms like CNN demonstrate accurate classifications (Mohanty et al., 2016).

In this paper, a plant stress detector is proposed which will classify plant leaves based on images. The device can make basic recommendations on how to handle plant stress.

II. LITERATURE REVIEW

Recently, there has been research into the applicability of deep learning in plant disease identification through the use of leaf images. The study by Mohanty et al. (2016) demonstrated that deep learning-based solutions could be used to develop image-based plant disease detection tools, achieving high accuracy results on the PlantVillage dataset. This paper highlighted the ability of CNNs in applications related to agriculture.

In addition, Ramcharan et al. (2017) explored deep learning solutions for cassava disease detection, concluding that deep learning approaches performed satisfactorily when applied to real images. It was stressed that deep learning algorithms should be validated with relevant datasets for more efficient performance.

Sandler et al. (2018) introduced MobileNetV2 as a lightweight architecture of deep learning models that can be implemented in real-time processing. The proposed MobileNetV2 has proven efficient in tasks of image classification thanks to its lower computational complexity.

Hughes and Salathé (2015) established an open-access repository of leaf images of plants for developing plant disease detection solutions. In particular, the developed dataset allowed building efficient plant disease detection tools.

In turn, Dhaka et al. (2021) conducted a review of various deep convolutional neural networks applied for plant leaf disease prediction. Another literature review concerning deep learning techniques applied in the field of plant disease detection has been carried out by Shoaib et al. (2023).

Balaji et al. (2022) discussed deep transfer learning based plant disease classification by exploring multiple modalities. Mazumder et al. (2021) suggested a lightweight transfer learning-based solution for accurate plant disease detection requiring fewer images for training.

Thus, the proposed solution uses a CNN-based architecture of MobileNetV2.

III. EXISTING SYSTEM

The detection of plant stress was conventionally done using manual techniques and traditional image processing methods. Manual techniques involve human visual inspections of leaf images for any abnormalities in plant health, such as discolored leaves, spots, and texture changes. Even though the technique is straightforward, it is domain-specific and susceptible to human error, particularly when symptoms are hard to detect.

Apart from manual techniques, traditional techniques include basic image processing methods like segmentation, thresholding, and feature extraction. In traditional image processing methods, handcrafted features such as colors, shapes, and textures are used to distinguish different plant states. Traditional image processing is highly dependent on factors such as image quality and lighting that can influence their accuracy.

On the other hand, machine learning techniques use handcrafted features that are then classified using models such as SVM and decision trees. Although such techniques perform better than manual techniques, they are highly dependent on hand-crafted features, making them inefficient in generalization across different datasets.

Lastly, most current techniques do not incorporate user-friendliness and cannot process images in real time.

As such, the current methods become inefficient, time consuming, and inaccurate in their functioning. Hence, there is a need for the development of a more automated, accurate, and simple way of detecting plant stresses.

IV. PROPOSED SYSTEM

The system seeks to address the problem of plant stress detection through leaf image analysis. While previous attempts relied on human effort, the system employs machine learning algorithms, specifically, deep learning models to classify plants efficiently.

To start the process, a user will upload a picture of a plant leaf through the web application. After that, the image is subjected to processes such as resizing, normalization, and noise reduction to enhance its quality. Such processes can help increase the efficiency of the classification algorithm.

Once the image has been processed, it will be fed into the CNN model trained on the MobileNetV2 framework. This process helps extract crucial information, including leaf colors, shapes, and textures. The model will use the information obtained during the extraction phase to classify the plant conditions into different categories, such as whether the plant is stressed or healthy.

Transfer learning is applied in the system to increase accuracy and reduce training time. The system generates the output alongside a confidence score once the classification process is complete. In addition, the system suggests measures to users based on the plant's conditions.

Generally, the proposed model will be precise, speedy, and easy to use; therefore, it can be adopted for real-time monitoring of plants in agriculture.

V. SYSTEM ARCHITECTURE

The design of the system architecture is aimed at ensuring efficient detection of plant stress through the use of images of plant leaves. The architecture ensures that the process of front-end user interaction, image processing, classification using deep learning algorithms, and recommendation generation is well-structured. All the components in the architecture are organized in a manner that facilitates effective flow of information from the start of the process until its completion.

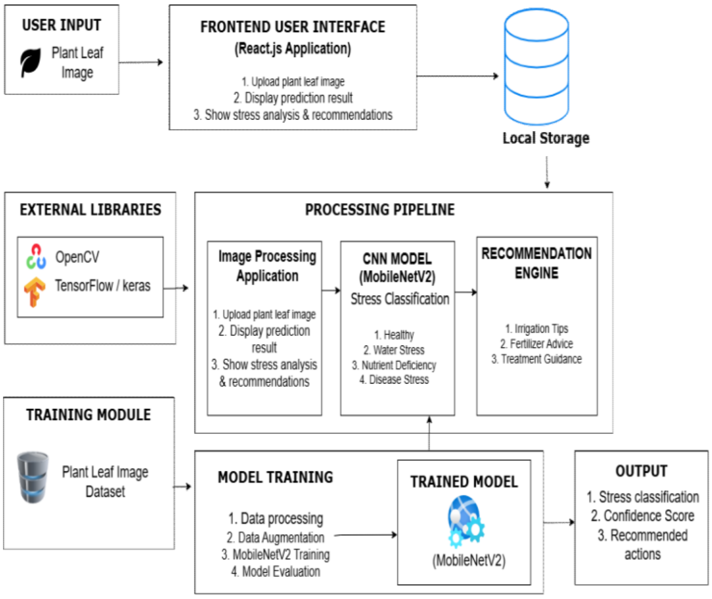

Fig.1 Architecture diagram of Plant Stress Detection Using Leaf Image Analysis

Fig.1 shows us, Firstly, User Input Module is present in which the user uploads the image of plant leaf through web application. The user input becomes the starting point in the processing of input data.

Secondly, there is the presence of Frontend User Interface (React.js Application). It enables users to upload images and view the results of the prediction as well as stress analysis. The recommendations are provided accordingly.

There is also an element of Local Storage in which the images as well as predictions will be stored for future analysis and study purposes.

Thirdly, the External Libraries Module contains tools such as OpenCV and TensorFlow/Keras. Such external libraries are helpful in the processing of images.

Finally, there is the presence of Processing Pipeline that consists of different stages. At the initial stage, the images are processed and analyzed at the Image Processing Application. They are resized and normalized before being classified. Then, they are forwarded to the CNN Model (MobileNetV2). The CNN Model categorizes the plant condition based on various conditions. For instance, the plant may have water stress, nutrient deficiency, disease stress, or healthy leaves.

After classification, the Recommendation Engine gives the relevant suggestions according to the classified disease like irrigation, fertilization, and remedies.

There is another module called Training Module, wherein the data set of the images of the plants' leaves is fed into the process of training the model. Data preprocessing, data augmentation, MobileNetV2 training, and evaluation are done here.

After training the model, it is kept in the module called Trained Model.

The Output Module provides the user with the information about the stress classified in the plant and the corresponding confidence level along with the suggested action to take by the Recommendation Engine.

All these together make sure that the stress in plants is detected efficiently and effectively.

V. SYSTEM DESIGN

The proposed system follows a systematic approach where deep learning techniques will be used to analyze the plant stress status through leaf images. The system is meant to receive data, pre-process them, and obtain the classification results.

A. Data Acquisition

Plant leaves images will be obtained from existing public datasets like PlantVillage alongside real-time images. The dataset consists of images of plants in different statuses; images of healthy leaves and diseased, nutrient-deficient, and water-stressed leaves.

B. Data Pre-processing

Data pre-processing is meant to make images clear and uniform. Images are resized into a particular size that can be accepted by the model. This will also make sure images are of the same size. Image normalization is done to aid model convergence. Noises present in the images are removed. Various data augmentation techniques such as rotations, flips, zooming, and scaling are implemented.

C. Feature Extraction

Images are automatically analyzed and important features extracted. These include the texture, color, and shape of the images.

D. Model Selection and Architecture

In this system, the architecture used is that of MobileNetV2, and it is efficient. The architecture utilizes inverted residual layers and linear bottlenecks to optimize computation while achieving high levels of accuracy. Pre-trained weights are used for transfer learning to ensure performance and faster training process.

E. Model Training

The dataset is divided into training, validation, and testing parts. The model training involves use of an optimizer and appropriate loss functions. In training, the algorithm learns how to classify images based on their features.

F. Prediction and Output Generation

The prediction step entails classification of new input images after training. The output produced includes class labels as well as confidence values. Recommendations are made based on predictions generated by the system.

G. Mathematical Formulation

1. Accuracy

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

Where:
TP = True Positives
TN = True Negatives
FP = False Positives
FN = False Negatives

2. Cross-Entropy Loss

$$L = -\sum_{i=1} y_i \log(\hat{y}_i)$$

Where:

y_i = actual label

$\hat{y}_i$ = predicted probability

3. Softmax Function

$$y_i = \frac{e^{zi}}{\sum_{j=1} e^{(zj)}}$$

Where:

Zi = input value

y_i = output probability

4. ReLU Activation Function

f(x) = max(0,x)

This function is used in hidden layers to introduce non-linearity.

VI. SYSTEM IMPLEMENTATION

A. User Interface and Image Input Module

The initial process involves a web-based client interface created by React.js allowing users to input their images. As a rule, it is rather simple and intuitive which makes it easy to interact with the software and perform necessary actions. An uploaded picture works as an input in further processes.

B. Image Preprocessing Module

As a next step, the system requires to carry out some operations on the uploaded image, for example, such as its resizing, normalization and cleaning from noise. These processes aim to prepare data properly for processing.

C. Deep Learning Frameworks Implementation

The most crucial part of the project can be realized with the help of Python-based frameworks including TensorFlow and Keras. In particular, CNN classification technique based on MobileNetV2 architecture is used. Transfer learning is utilized in this case.

D. Model Training and Evaluation

Dataset will be split into three parts, namely, training, validation, and testing. During training and validation stages, the optimizer and cross entropy will be utilized. Besides, different metrics like accuracy and loss will be calculated to evaluate model performance.

E. Prediction & Recommendation Engine

The model, once built, is used for predicting real-time data. The system will perform analysis of the uploaded picture of the leaf and classify it in terms of whether it is healthy or sick. Recommendations, such as irrigation advice, fertilizers, or disease treatment, will be provided accordingly.

F. Data Storage & Backend Processes

The system will employ either local storage or a lightweight database for storing the uploaded picture and the results of its prediction. Efficient backend processes will be implemented to ensure the speed of responses of the system.

G. Deployment & System Performance

The system is a web application and is accessible via browsers. The average inference time per picture does not exceed two seconds.

H. Output Visualization

The output of the suggested model is displayed using the web interface for various plants. The outputs obtained from the proposed system for the detection of pepper, potato, and tomato leaves are given below.

1) Potato Leaf Analysis

The user starts the diagnostic process by uploading a clear image of a potato leaf to the PlantSense AI interface. This step lets the system prepare the visual data for processing.

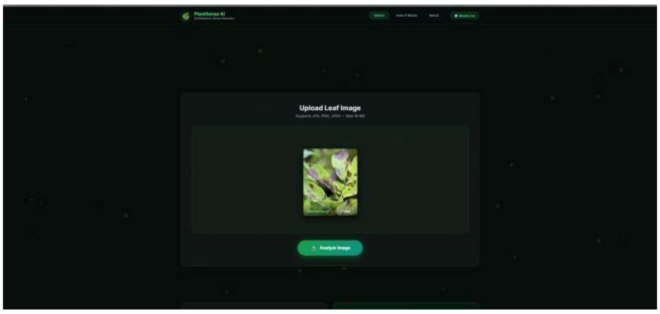

Fig.2 : Input image - potato leaf

The dashboard shows that the potato leaf uploaded successfully and provides a clear preview of the leaf's current condition before detailed analysis starts.

After processing, the system detects Late Blight with a 98% confidence rating, backed by specific spectral indices. The analysis shows significant necrotic lesions and gives immediate suggestions for chemical and cultural control.

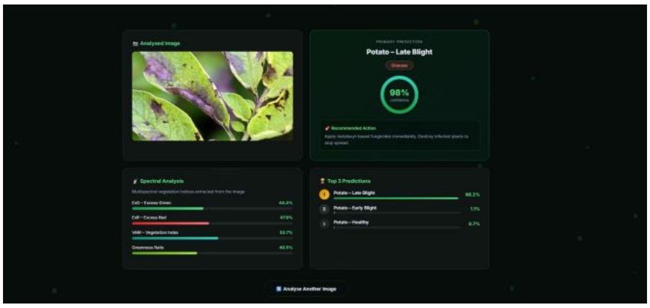

Fig.3 : Late Blight Disease - potato leaf

This report includes multispectral vegetation indices and a ranked list of predictions, ensuring the grower has a data-supported plan for crop intervention.

2) Pepper Leaf Analysis

For the pepper plant assessment, a sample leaf with circular lesions is sent to the multispectral stress detection model. The interface can handle various file formats to make it easy for field researchers.

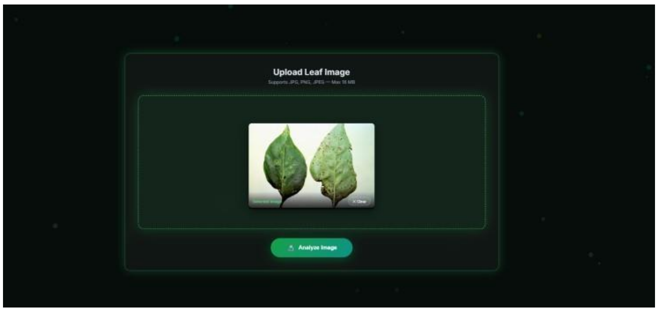

Fig.4 : Input image - pepper leaf

The input image shows the affected pepper leaf, which serves as the main source for the digital pathology analysis.

The analysis confirms Bacterial Spot disease with 97% confidence and highlights the specific spectral signatures linked to this pathogen. The system recommends using copper-based bactericides and better irrigation management to stop further spread.

Fig.5 : Bacterial Spot Disease - pepper leaf

By comparing the spectral analysis to known healthy ratios, the dashboard offers a clear visual of the plant's physiological stress levels.

3) Tomato Leaf Analysis

The final analysis looks at a tomato leaf with small, dark spots that suggest fungal or bacterial infections. The user starts the "Analyze Image" function to begin the deep learning classification.

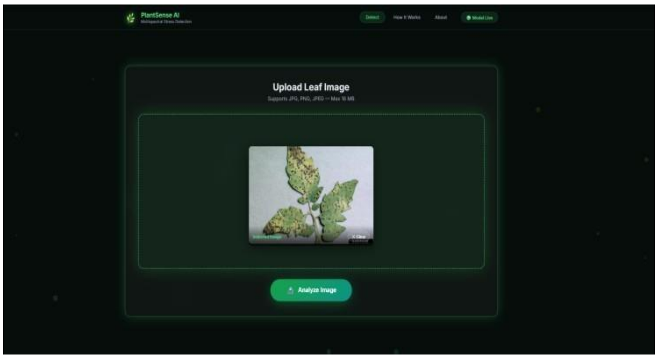

Fig.6 : Input image - tomato leaf

The uploaded tomato leaf image is queued for analysis, showing the characteristic spots that the AI model will quantify.

PlantSense AI diagnoses Septoria Leaf Spot with a very high confidence level of 99%. The analysis ends with a recommendation to remove infected leaves right away and apply mancozeb fungicide.

Fig.7 : Setoria leaf Spot Disease - tomato leaf

From the analysis above, it can be concluded that the proposed system successfully identifies plant stress in various crops like peppers, potatoes, and tomatoes. The proposed model can also correctly classify particular diseases like bacterial spots for peppers, late blight for potatoes, and septoria leaf spots for tomatoes. The system produces an output of predicted classes together with confidence scores and recommendations.

VII. RESULTS AND GRAPHS

The effectiveness of the suggested plant stress detection model is analyzed through several criteria, including accuracy, loss function, confusion matrix, and classification performance. The learning model is trained and tested using a data set comprising images of plant leaves in various states.

The results show that the learning algorithm has a training accuracy of about 97%, a validation accuracy of about 95%, and a testing accuracy of between 93% and 95%.

The performance of the proposed plant stress detection system is evaluated graphically by the help of parameters like accuracy, loss, confusion matrix, and other classification metrics.

A. Accuracy Analysis

Accuracy of the model during the training process across different epochs is illustrated in Fig.8 below.

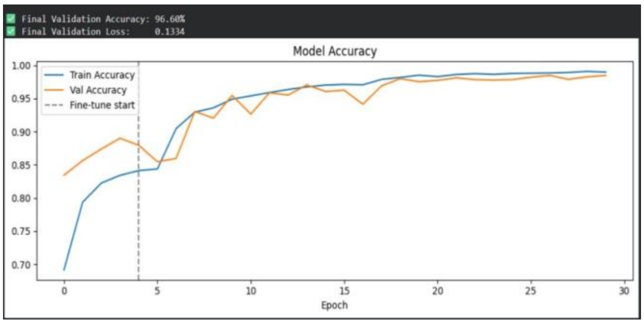

Fig.8 : Training and Validation Accuracy

From the graph, it is clear that the accuracy of the training continues to rise for each epoch. Similarly, the validation accuracy also continues to improve for each epoch. The slight difference between both values demonstrates that the model has no overfitting issues.

B. Loss Analysis

The loss during training and validation of the model is depicted in Fig.9

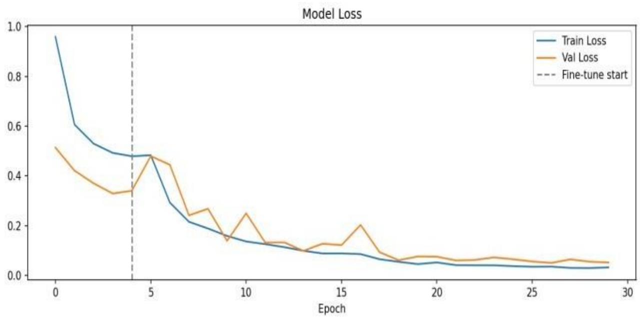

Fig.9 : Training and Validation Loss

From the above figure, it can be seen that the loss decreases continuously with increasing training. It implies that the model is performing well in reducing the errors in its predictions.

C. Confusion Matrix

The confusion matrix for the model is presented in Fig.10

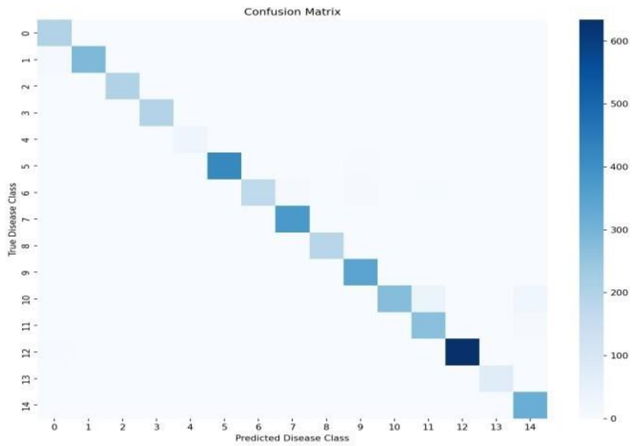

Fig.10 : Confusion Matrix for Classification Output

The confusion matrix presents the results of classification by the model. It is evident that the entries are mostly around the main diagonal line, which means that most of the samples are classified accurately by the model. The small number of off-diagonal entries indicates few errors.

D. Performance Metrics

The performance metrics such as precision, recall, and F1-score are shown in **Fig.11**

Fig.11 : Precision, Recall, and F1-Score

From the graph, it can be seen that the model provides high values of precision and recall for each category. Moreover, the F1 score depicts an equilibrium between precision and recall, proving that the model is effective in identifying plant stress conditions.

G. System Performance

The performance of the entire system is assessed based on accuracy and speed. The system is capable of predicting with an average prediction speed of less than 2 seconds per image, thereby allowing the system to be used in real-time scenarios.

The findings indicate that the developed system is efficient, accurate, and dependable for plant stress detection.

The graph shows that the model achieves high precision and recall values across different classes. The F1-score indicates a good balance between precision and recall, confirming the effectiveness of the model in detecting plant stress conditions.

H. Performance Evaluation

Table I. Performance Evaluation of Proposed Model

Metric	Value
Training Accuracy	97%
Validation Accuracy	95%
Test Accuracy	93–95%
Precision	94%
Recall	93%
F1-score	94%
Inference Time	< 2 seconds

VIII. CONCLUSION & FUTURE ENHANCEMENT

The proposed plant stress detection system based on deep learning provides an efficient solution for identifying plant diseases using leaf images. By utilizing a lightweight architecture such as MobileNetV2, the system achieves high accuracy with low computational complexity.

The experimental results, including accuracy, loss, confusion matrix, and performance metrics, demonstrate that the model performs reliably with minimal misclassification. The system successfully detects diseases such as **bacterial spot in pepper**, **late blight in potato**, and **septoria leaf spot in tomato**.

Furthermore, the system provides predictions with low inference time, making it suitable for real-time agricultural applications. Overall, the proposed approach offers a practical and scalable solution for early plant stress detection.

FUTURE ENHANCEMENTS :

The proposed system can be further enhanced by incorporating a larger and more diverse dataset to improve model generalization across different plant species and environmental conditions. Future improvements may include integrating the system into a mobile application to enable real-time usage by farmers in field conditions.Additionally, features such as disease severity estimation, automated treatment recommendations, and multilingual support can be included to make the system more user- friendly and impactful in real-world agricultural scenarios.

REFERENCES

[1] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," Frontiers in Plant Science, 2016.

[2] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in Proc. IEEE CVPR, 2018.

[3] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, "Deep learning for image-based cassava disease detection," Frontiers in Plant Science, 2017.

[4] D. P. Hughes and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," arXiv, 2015.

[5] J. J. Walsh, E. Mangina, and S. Negrão, "Advancements in imaging sensors and AI for plant stress detection: A systematic literature review," 2021.

[6] V. S. Dhaka et al., "A survey of deep convolutional neural networks applied for prediction of plant leaf diseases," Sensors, 2021.

[7] G. ElMasry et al., "Recent applications of multispectral imaging in seed phenotyping and quality monitoring—An overview," Sensors, 2019.

[8] M. Shoaib et al., "An advanced deep learning models-based plant disease detection: A review of recent research," 2022.

[9] J. J. Walsh, E. Mangina, and S. Negrão, "Advancements in imaging sensors and AI for plant stress detection: A systematic literature review," 2021.

[10] V. Balaji et al., "Deep transfer learning technique for multimodal disease classification in plant images," 2022.

[11] M. K. A. Mazumder et al., "A robust and light-weight transfer learning-based architecture for accurate detection of leaf diseases across multiple plants using less amount of images," 2021.

[12] M. S. Anari, "A hybrid model for leaf diseases classification based on modified deep transfer learning and ensemble approach for agricultural AIoT-based monitoring," 2022.

[13] H. N. Ngugi, A. E. Ezugwu, A. A. Akinyelu, and L. Abualigah, "Revolutionizing crop disease detection with computational deep learning: A comprehensive review," 2024.

[14] M. U. I. Tamim, S. A. Hamim, S. Malik, M. F. Mridha, and S. Mahmood, "InsightNet: A deep learning framework for enhanced plant disease detection and explainable insights," 2025.

[15] A. Sharma et al., "Artificial intelligence and internet of things oriented sustainable precision farming: Towards modern agriculture," 2022.

[16] G. Saha et al., "Smart IoT-driven precision agriculture: Land mapping, crop prediction, and irrigation system," 2022.

[17] S. Mishra, "Internet of things enabled deep learning methods using unmanned aerial vehicles for integrated farm management," 2022.

[18] S. Vallabhajosyula, V. Sistla, and V. K. Kishore, "A novel hierarchical framework for plant leaf disease detection using residual vision transformer," 2023.