© MAY 2026 | IRE Journals | Volume 9 Issue 11 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I11-1717463

Image-Based Analysis for Identification of Plant Leaf Pathologics Using Deep Learning

R. DEEPAK KESAV[1], DR. S. PRAKASAM[2]

[1]*PG Student, Department of Computer Application, SCSVMV deemed to be University, Kanchipuram, Tamilnadu, India*

[2]*Associate Professor, Department of Computer Application, SCSVMV deemed to be University, Kanchipuram, Tamilnadu, India*

Abstract- This project introduces a Convolutional Neural Network (CNN) as the proposed system for plant disease prediction, with a comparative analysis conducted against two existing models: Recurrent Neural Networks (RNN_GRU) and Artificial Neural Networks (ANN_MLP). The proposed CNN model is specifically designed to address the limitations of traditional approaches, such as lower accuracy and slower prediction times, particularly when handling complex image data. The system allows users to upload images of plant leaves, select the type of plant (e.g., potato, tomato, grape), and choose between RNN_GRU, ANN_MLP, or the newly developed CNN model for disease prediction. Additionally, users can run all three models simultaneously to compare their outputs, enabling a comprehensive evaluation of performance. Predictions are securely stored in a SQLite database, along with metadata such as confidence scores, prediction times, timestamps, and a unique group ID for efficient retrieval and management. Built using Flask, the application provides a professional-grade user interface with features like secure authentication, prediction history tracking, and deletion of past predictions. Comparative analysis demonstrates that the proposed CNN model significantly outperforms RNN_GRU and ANN_MLP in terms of accuracy, prediction speed, and overall reliability, making it a more effective tool for real-time agricultural applications. This advancement highlights the potential of CNNs in transforming agricultural practices by providing faster, more accurate, and reliable disease predictions, thereby contributing to improved crop health, reduced losses, and increased agricultural productivity.

Keywords: Plant Disease Detection, Recurrent Neural Network (RNN), Artificial Neural Network (ANN), Convolutional Neural Network (CNN), Flask, SQLite, Image Processing, Machine Learning, Prediction Pipeline.

I. INTRODUCTION

This project focuses on developing a plant disease prediction system aimed at addressing the critical need for early detection of diseases in crops. The system integrates three machine learning models—Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN_GRU), and Artificial Neural Networks (ANN_MLP)—to analyze images of plant leaves and predict potential diseases. Among these, the CNN model is introduced as the primary predictive mechanism due to its superior performance in handling complex image data compared to the existing RNN_GRU and ANN_MLP models. Users can upload leaf images, select the plant type (e.g., potato, tomato, grape), and choose between individual models or run all three simultaneously for comparative analysis. The system stores predictions in a SQLite database, grouped by unique IDs for efficient retrieval, and provides a secure, user-friendly Flask-based interface with features like user authentication, prediction history tracking, and deletion of past predictions. By comparing the performance of CNN, RNN_GRU, and ANN_MLP, the project highlights the advantages of the CNN model in terms of accuracy, efficiency, and dependability, offering a practical solution for improving agricultural productivity and crop health. This system is designed to assist farmers, researchers, and agricultural professionals in making informed decisions, ultimately contributing to sustainable agricultural practices and enhanced food security.

II. LITERATURE REVIEW

The detection and classification of plant diseases using machine learning and deep learning techniques have gained significant attention in recent years due to their potential to improve agricultural productivity and reduce crop losses. Traditional methods of disease identification relied heavily on manual inspection by experts, which is time-consuming, costly, and prone to human error. With the advancement of artificial intelligence, automated systems have been developed to provide faster and more accurate disease detection.

Early research in this domain focused on Artificial Neural Networks (ANN) for classification tasks. ANN models, particularly Multi-Layer Perceptrons (MLP), were used to identify patterns in plant leaf images. Although these models demonstrated moderate success, they lacked the ability to effectively capture spatial features in images, resulting in limited accuracy for complex datasets.

Subsequently, Recurrent Neural Networks (RNN), especially GRU (Gated Recurrent Unit) models, were introduced to improve prediction performance. RNNs are designed to handle sequential data and have been applied in various domains. However, their application in image-based plant disease detection is not optimal because they do not inherently capture spatial relationships within images. This limitation reduces their effectiveness compared to other deep learning approaches.

With the evolution of deep learning, Convolutional Neural Networks (CNN) have emerged as the most effective technique for image-based classification tasks. CNNs are specifically designed to process visual data and automatically extract relevant features such as edges, textures, and patterns from images. Research studies have shown that CNN models significantly outperform traditional machine learning and ANN-based approaches in terms of accuracy and efficiency.

Recent literature also emphasizes the integration of multiple models for comparative analysis. Hybrid systems that combine CNN, ANN, and RNN models provide a better understanding of model performance and reliability. Such systems allow users to compare prediction accuracy, confidence scores, and processing time, enabling more informed decision-making.

Moreover, web-based implementations using frameworks like Flask have been explored to make these models accessible to end users. These systems allow users to upload images, process them using trained models, and store results in databases for future reference. The inclusion of features such as user authentication, prediction history, and real-time analysis enhances usability and practicality in real-world agricultural applications.

Despite these advancements, challenges still exist, including the need for large and diverse datasets, handling variations in lighting and image quality, and improving model generalization. Ongoing research focuses on optimizing deep learning architectures, integrating transfer learning, and deploying systems on mobile and cloud platforms for wider accessibility.

III. PROBLEM STATEMENT

Agriculture plays a vital role in ensuring food security and economic stability, but plant diseases significantly affect crop yield and quality. Early and accurate detection of plant diseases is essential to minimize losses and improve productivity. However, traditional methods of disease identification rely on manual observation by farmers or experts, which is time-consuming, inefficient, and often inaccurate due to human error and lack of expertise.

Existing automated systems use machine learning techniques such as Artificial Neural Networks (ANN) and Recurrent Neural Networks (RNN), but these approaches face limitations when dealing with complex image data. They are not highly effective in capturing spatial features of plant leaf images, resulting in lower accuracy and slower prediction performance. Additionally, many existing systems do not provide a mechanism to compare multiple models, making it difficult to determine the most reliable prediction method.

There is a need for an intelligent, efficient, and user-friendly system that can accurately detect plant diseases from leaf images using advanced deep learning techniques. The system should support multiple models, enable comparative analysis, and provide fast and reliable predictions. It should also allow users to upload images easily, store prediction history, and access results through a web-based interface.

Therefore, the problem is to design and develop a robust plant disease prediction system that overcomes the limitations of traditional and existing methods by improving accuracy, speed, and usability while supporting real-time decision-making in agriculture.

IV. OBJECTIVE

The main objective of this project is to develop an intelligent and efficient plant disease prediction system using advanced machine learning and deep learning techniques. The system aims to accurately identify plant diseases from leaf images and provide reliable results to support agricultural decision-making. To achieve this, the project integrates multiple models such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN_GRU), and Artificial Neural Networks (ANN_MLP), with a focus on improving prediction accuracy and speed. A user-friendly web-based interface is developed using Flask, allowing users to upload images, select models, and view results easily. The system also includes features such as secure user authentication, prediction history tracking, and database management using SQLite. Additionally, the project enables comparative analysis of different models based on performance metrics like confidence score and prediction time, helping users identify the most effective model. Overall, the objective is to build a scalable, reliable, and practical solution that enhances crop health monitoring and contributes to increased agricultural productivity.

V. SYSTEM ANALYSIS

System analysis is a critical phase in software development that focuses on understanding the system requirements, identifying problems in the existing system, and proposing effective solutions. It helps in defining system functionalities, improving performance, and ensuring that the developed system meets user needs efficiently.

In this project, the system analysis focuses on designing a plant disease prediction system that uses machine learning models to analyze plant leaf images and predict diseases accurately.

A. Existing System

The existing system utilizes traditional machine learning models, specifically Recurrent Neural Networks (RNN_GRU) and Artificial Neural Networks (ANN_MLP), for plant disease prediction. These models analyse images of plant leaves to identify potential diseases based on patterns learned during training. While functional, these models face challenges in terms of accuracy, efficiency, and robustness when dealing with complex image data. RNN_GRUs are primarily designed for sequential data and may struggle with spatial features in images, while ANN_MLPs lack the specialized architecture needed for high-dimensional image analysis. Additionally, the existing system does not provide a comparative analysis of multiple models, limiting its ability to offer comprehensive insights into prediction reliability.

B. System Architecture

The system architecture of the Plant Disease Prediction System is designed using a modular and layered approach to ensure efficiency, scalability, and ease of maintenance. The system consists of four main components: the presentation layer, application layer, machine learning layer, and database layer. The presentation layer, developed using HTML and CSS, provides a user-friendly interface that allows users to register, log in, upload plant leaf images, select the type of plant and prediction model, and view results. The application layer is built using Python with the Flask framework and acts as the core of the system, handling user requests, authentication, image validation, and communication between different components. The machine learning layer includes three models—Convolutional Neural Network (CNN), Recurrent Neural Network (RNN_GRU), and Artificial Neural Network (ANN_MLP)—which process the uploaded images and generate predictions such as disease type, confidence score, and prediction time, with CNN being the primary model due to its higher accuracy in image processing. The database layer uses SQLite to store user credentials and prediction history, ensuring efficient data management. Additionally, the system includes a file validation module to ensure secure and proper handling of uploaded images. Overall, the architecture enables smooth interaction between components, providing fast, accurate, and reliable plant disease prediction.

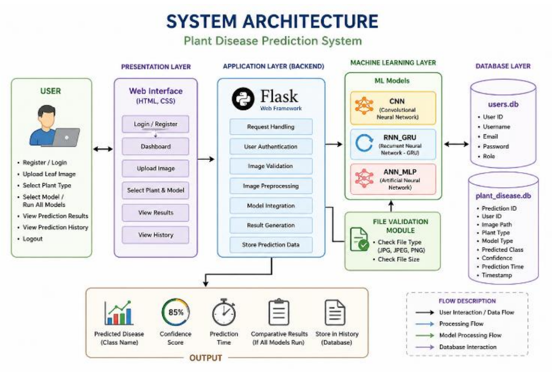

Fig. 1. System Architecture Diagram

C. Data Flow

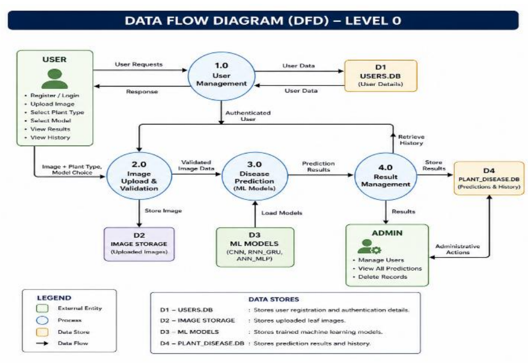

Fig. 2. Data Flow Diagram

The Data Flow Diagram (DFD) of the Plant Disease Prediction System illustrates how data moves through the system and how different components interact to process user inputs and generate outputs. Initially, the user interacts with the system by registering or logging in and then uploading a plant leaf image along with selecting the plant type and prediction model. The system processes this input through the user management module, which handles authentication and stores user details in the database. The uploaded image is then passed to the image upload and validation module, where it is checked for correct format and size before being stored. Once validated, the image is forwarded to the disease prediction module, where machine learning models such as CNN, RNN_GRU, and ANN_MLP analyze the image to detect the disease and generate prediction results including the disease name, confidence score, and prediction time. These results are then sent to the result management module, which stores the data in the database and displays it to the user. Additionally, the system allows users to view their prediction history, while the admin can manage users and monitor stored data. Overall, the DFD provides a clear representation of how data flows efficiently through the system to ensure accurate and timely plant disease prediction.

D. System Database Design

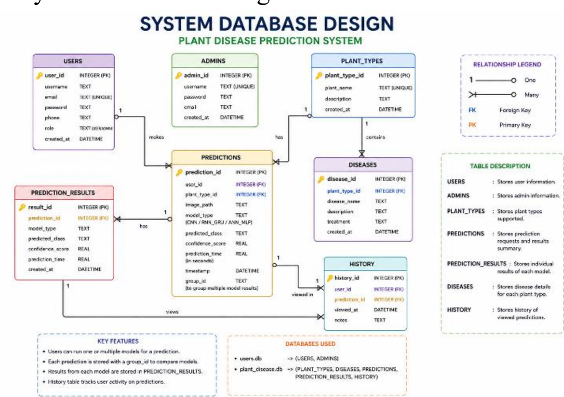

Fig. 3. System Database Diagram

The database design of the Plant Disease Prediction System is structured to efficiently store, manage, and retrieve all relevant data required for system operations. The system follows a relational database model where data is organized into multiple interconnected tables. Each table is assigned a primary key to uniquely identify records, while foreign keys are used to establish relationships between tables, ensuring data integrity and reducing redundancy. This structured design improves performance, consistency, and scalability of the system.

The system primarily uses two databases: users.db and plant_disease.db. The users.db database stores user and administrator information, including login credentials and roles, while the plant_disease.db database manages data related to plant types, diseases, predictions, results, and user activity history.

The USERS table stores details of registered users such as user ID, username, email, password, and role.

© MAY 2026 | IRE Journals | Volume 9 Issue 11 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I11-1717463

Each user can perform multiple predictions, establishing a one-to-many relationship with the predictions table. The ADMINS table maintains administrator information, allowing authorized access to manage users and system data.

The PLANT_TYPES table contains information about different plant categories such as potato, tomato, and grape. Each plant type can have multiple associated diseases. The DISEASES table stores disease-related information including disease name, description, and treatment, and is linked to the plant types table using a foreign key.

The PREDICTIONS table acts as the central table in the system, storing details of each prediction request such as user ID, plant type, image path, selected model, predicted class, confidence score, prediction time, and timestamp. A group ID is used to group predictions generated by multiple models for the same image, enabling comparison between models.

The PREDICTION_RESULTS table stores detailed outputs generated by each model, allowing the system to compare results from CNN, RNN_GRU, and ANN_MLP models. The HISTORY table keeps track of user activities, including viewed predictions and timestamps, enabling users to revisit past results.
Overall, the database design ensures efficient data storage, quick retrieval, and proper relationship management between entities. By using primary keys, foreign keys, and well-structured tables, the system maintains data consistency and supports all core functionalities such as user management, prediction processing, result comparison, and history tracking.

VI. METHODOLOGY

The methodology of the Plant Disease Prediction System follows a structured approach that integrates image processing, machine learning, and web application development to achieve accurate disease detection. The process begins with data collection, where a dataset of plant leaf images representing different diseases and healthy conditions is gathered. These images are preprocessed through resizing, normalization, and noise removal to ensure consistency and improve model performance.

User Authentication: Manages user registration, login, and session handling using a SQLite database (users.db). Ensures secure access with unique usernames/emails and password validation. Session management is implemented via Flask's session mechanism.

Image Upload and Validation: Handles file uploads, ensuring only valid formats (PNG, JPG, JPEG) and sizes (≤5MB) are accepted. Files are securely processed using secure_filename, and invalid uploads trigger error messages.

Prediction Processing: Processes uploaded images using CNN, RNN_GRU, or ANN_MLP models. Users can select one model or run all three for comparison. Results, including predicted class, confidence score, and prediction time, are stored in the plant_disease.db database with a unique group_id.

Database Management: Uses two SQLite databases: users.db for authentication and plant_disease.db for predictions. Predictions include metadata like leaf type, model type, confidence, and timestamps. Supports deletion of individual predictions or clearing the entire history.
Result Display: Presents prediction results in an organized format. For single models, it shows the predicted class, confidence, and time. For "All Models," outputs are compared side-by-side. Uploaded images are displayed alongside results with timezone-aware timestamps. • History and Management: Allows users to view their prediction history, grouped by group_id when multiple models are used. Metadata like leaf type, model type, and timestamps are displayed. Users can delete individual predictions or clear the entire history.

VII. RESULTS AND DISCUSSION

The provided code is a Flask-based web application designed to predict plant leaf diseases using machine learning models (CNN, RNN_GRU, ANN_MLP) and provide a seamless user experience through features like user authentication, file upload validation, and prediction history tracking. The application allows users to upload images of plant leaves (potato, tomato, grape), select a model or use all models for

© MAY 2026 | IRE Journals | Volume 9 Issue 11 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I11-1717463

predictions, and view detailed results, including predicted class, confidence, and prediction time. It integrates SQLite databases to store user credentials and prediction records, with a unique `group_id` mechanism to group predictions from "All Models" cases. The system enforces strict file validation rules, such as allowed file types (PNG, JPG, JPEG) and a maximum size limit of 5MB, ensuring secure and error-free uploads. Additionally, it provides a history page where users can view past predictions, delete specific entries, or clear all records, with timestamps converted to the local timezone (Asia/Kolkata) for better usability. Error handling is robust, ensuring invalid inputs, unexpected scenarios, and database errors are managed gracefully with clear feedback to users.

A. Customer Dashboard

the user registration (Sign Up) interface of the Plant Disease Prediction System. The interface is designed with a visually appealing background consisting of green plant leaves, which aligns with the agricultural theme of the application. At the center of the screen, a semi-transparent form is displayed to ensure readability while maintaining the aesthetic background. The form includes essential input fields such as Username, Email, and Password, allowing users to create a new account securely. Each input field is clearly labeled, and placeholder text is provided to guide users on what information to enter. A prominent "Sign Up" button is placed below the input fields, enabling users to submit their registration details. Additionally, there is a navigation option for existing users, displayed as "Already have an account? Log In", which redirects them to the login page.

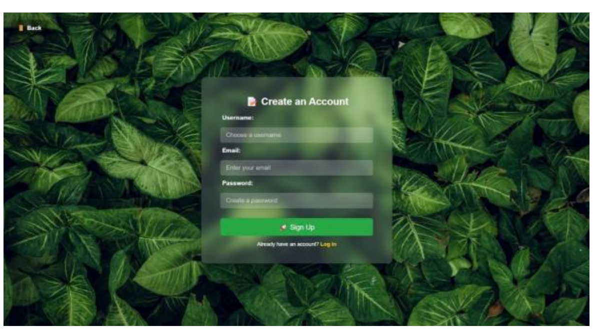

Fig. 4. Customer Dashboard

B. Admin Dashboard

The given image represents the Admin Registration (Add Admin) interface of the Plant Disease Prediction System. The design follows a consistent visual theme with a green leafy background, symbolizing the agricultural nature of the application. At the center of the screen, a semi-transparent form is displayed, ensuring both readability and aesthetic appeal.

The form contains input fields such as Username, Email, and Password, which are required to create a new admin account. Each field is clearly labeled and includes placeholder text to guide the user in entering appropriate information. A prominent "Sign Up" button is provided to submit the entered details and complete the admin registration process

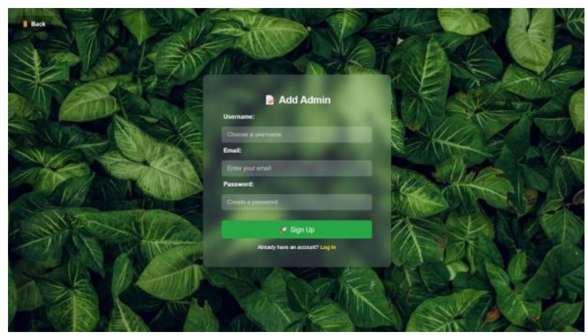

Fig. 5. Admin Dashboard

C. prediction History

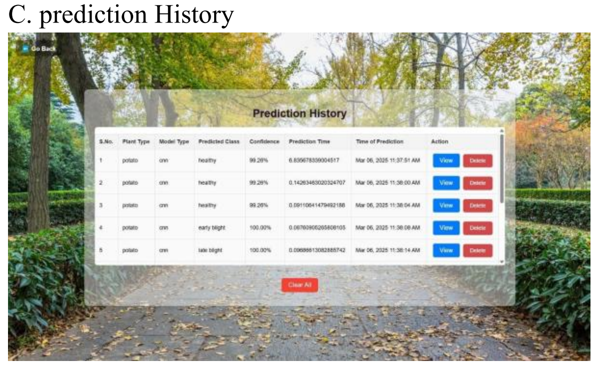

Fig. 6. Prediction History

the Prediction History interface of the Plant Disease Prediction System. This page is designed to display previously generated prediction results in a structured and user-friendly tabular format. The background features a natural outdoor scene with trees and greenery, maintaining consistency with the agricultural theme of the application, while a semi-transparent panel is placed in the center to clearly present the data. Each row represents a prediction made by the user, showing details like the type of plant (e.g., potato), the model used (CNN), the predicted disease (such as healthy, early blight, or late blight), and the confidence percentage. The prediction time indicates how long the model took to generate the result, while the timestamp shows when the prediction was made.

VIII. CONCLUSION

The Plant Disease Prediction System developed in this project successfully demonstrates the effective use of machine learning and deep learning techniques in the field of agriculture. The system provides an efficient, accurate, and user-friendly solution for identifying plant diseases using leaf images. By integrating advanced models such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN_GRU), and Artificial Neural Networks (ANN_MLP), the system enables comparative analysis and highlights the superiority of CNN in handling image-based classification tasks.

The implementation of a web-based platform using Flask enhances accessibility and usability, allowing users to easily upload images, select models, and view prediction results in real time. Features such as secure user authentication, prediction history tracking, and database management further improve the practicality and reliability of the system. The ability to store and retrieve prediction data ensures that users can analyze past results and make informed decisions regarding crop management.

One of the key achievements of this project is the improvement in prediction accuracy and processing speed, especially with the use of CNN models, which are well-suited for extracting spatial features from images. The inclusion of multiple models also provides valuable insights into their performance, enabling users to compare results based on confidence scores and prediction time. This comparative approach adds flexibility and transparency to the system.

Furthermore, the system addresses the limitations of traditional manual disease detection methods, which are often time-consuming and dependent on expert knowledge. By automating the process, the system reduces human effort, minimizes errors, and provides quick and reliable results. This makes it a valuable tool for farmers, researchers, and agricultural professionals.

IX. LIMITATIONS

The Plant Disease Prediction System, while effective and innovative, has certain limitations that need to be considered. One of the primary limitations is the dependency on the quality of input images. If the uploaded leaf images are blurry, poorly lit, or taken from complex backgrounds, the accuracy of the prediction may decrease. The system performs best when images are clear and properly captured.

Another limitation is the reliance on a limited dataset. The models are trained on specific plant types such as potato, tomato, and grape, which restricts the system from identifying diseases in other crops. Additionally, the model may not generalize well to unseen diseases or variations that were not included in the training dataset.

The system currently uses a local SQLite database, which is suitable for small-scale applications but may not perform efficiently when handling large amounts of data or multiple concurrent users. This limits scalability and may require migration to more robust database systems in the future.

X. FUTURE ENHANCEMENTS

The Plant Disease Prediction System can be further improved and extended in several ways to enhance its performance, scalability, and real-world applicability. One of the major enhancements is the expansion of the dataset to include a wider variety of crops and diseases. By training the models on larger and more diverse datasets, the system can improve its accuracy and generalization capabilities, enabling it to detect diseases across multiple plant species.

Another important enhancement is the integration of advanced deep learning techniques such as transfer learning and pre-trained models (e.g., ResNet, EfficientNet), which can significantly improve prediction accuracy while reducing training time. The system can also be optimized for mobile platforms by developing a mobile application, allowing farmers to capture images directly from their smartphones and receive instant predictions.

To improve scalability, the system can be migrated from SQLite to more powerful databases like MySQL or PostgreSQL and deployed on cloud platforms such as AWS or Google Cloud. This will enable the system to handle a larger number of users and real-time data efficiently.

REFERENCES

[1] J. Kolli, D. M. Vamsi and V. M. Manikandan, "Plant Disease Detection using Convolutional Neural Network," 2021 IEEE Bombay Section Signature Conference (IBSSC), Gwalior, India,2021, pp. 1-6, Doi: 10.1109/IBSSC53889.2021.9673493.

[2] G. Shrestha, Deepsikha, M. Das and N. Dey, "Plant Disease Detection Using CNN," 2020IEEE Applied Signal Processing Conference (ASPCON), Kolkata, India, 2020, pp. 109-113,doi: 10.1109/ASPCON49795.2020.9276722

[3] G. Madhulatha and O. Ramadevi, "Recognition of Plant Diseases using Convolutional Neural Network," 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 2020, pp. 738-743, Doi: 10.1109/I- SMAC49090.2020.9243422.

[4] Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. Frontiers in Plant Science, 7, 1419. https://doi.org/10.3389/fpls.2016.01419

[5] Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. Computers and Electronics https://doi.org/10.1016/j.compag.2018.01.009in Agriculture, 145, 311-318.

[6] Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016). Deep neural networks based recognition of plant diseases by leaf image classification. Computational Intelligence and Neuroscience, 2016, 1-11. https://doi.org/10.1155/2016/3289801

[7] Too, E. C., Yujian, L., Njuki, S., & Yingchun, L. (2019). A comparative study of fine-tuning deep learning models for plant disease identification. Computers and Electronics in Agriculture, 161, 272-279. https://doi.org/10.1016/j.compag.2018.03.032

[8] Amara, J., Bouaziz, B., & Algergawy, A. (2017). A deep learning-based approach for banana leaf diseases classification. Proceedings of the Datenbanksysteme für Business, Technologie und Web (BTW), 79-88.

[9] Zhang, S., Huang, W., & Zhang, C. (2019). Three-chANN_MLPel convolutional neural networks for vegetable leaf disease recognition. Cognitive Systems Research, 53, 31-41. https://doi.org/10.1016/j.cogsys.2018.04.006 81

[10] Fuentes, A., Yoon, S., Kim, S. C., & Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. Sensors, 17(9), 2022. https://doi.org/10.3390/s17092022

[11] Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. Computers and Electronics https://doi.org/10.1016/j.compag.2018.02.016 in Agriculture, 147, 70-90.

[12] Picon, A., Alvarez-Gila, A., Seitz, M., Ortiz-Barredo, A., Echazarra, J., & JohANN_MLPes, A. (2019). Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. Computers and Electronics in Agriculture, 161, 280-290. https://doi.org/10.1016/j.compag.2018.04.010

[13] Lu, Y., Yi, S., Zeng, N., Liu, Y., & Zhang, Y. (2017). Identification of rice diseases using deep convolutional neural networks. https://doi.org/10.1016/j.neucom.2017.06.023 Neurocomputing, 267, 378-384.

[14] Wiesner-Hanks, T., & Stewart, E. L. (2020). Remote sensing of plant diseases. ANN_MLPual Review of Phytopathology, 58, 89-107. https://doi.org/10.1146/ANN_MLPurev-phyto-010820- 012855

[15] Brahimi, M., Boukhalfa, K., & Moussaoui, A. (2017). Deep learning for tomato diseases: Classification and symptoms visualization. Applied Artificial Intelligence, 31(4), 299-315. https://doi.org/10.1080/08839514.2017.