Diyala Journal of Engineering Sciences

Journal homepage: https://djes.info/index.php/djes

ISSN: 1999-8716 (Print); 2616-6909 (Online)

A Portable AI-Driven Edge Solution for Automated Plant Disease Detection

Adel Jalal Yousif [1*], Fadhil Kadhem Zaidan[2], Nebras Jalel Ibrahim[1]

[1]Department of Computer Engineering, College of Engineering, University of Diyala, 32001 Diyala, Iraq
[2] Department of Computer Science, College of Science, University of Diyala, 32001 Diyala, Iraq

ARTICLE INFO

Article history:
Received March 10, 2025
Revised July 01, 2025
Accepted July 22, 2025
Available online September 01, 2025

Keywords:
Plant Disease Detection
Jetson Nano
Edge Computing
Deep Learning
PlantVillage
Smart Agricultural
Leaf Disease Classification

ABSTRACT

Plant diseases can cause severe damage to crops and lead to food shortages and financial losses for both farmers and the agricultural sector. Detecting these diseases early is essential for protecting crops, increasing agricultural productivity, and ensuring food security. This paper introduces a new intelligent edge computing framework that provides a cost-effective, portable, and energy-efficient solution for deep learning-based automated plant disease detection. Unlike cloud-dependent systems, the proposed framework operates independently of an internet connection, making it ideal for real-time field deployment. It employs the NVIDIA Jetson Nano as an edge computing device and incorporates an Android-based interface for user interaction. The system utilizes a convolutional neural network (CNN) for feature extraction, followed by a deep classification network to identify plant diseases. Plant images are captured by a smartphone and transmitted to the Jetson Nano over a local WiFi network using the KDE Connect application for processing. After classification, the image with the predicted disease category is sent back to the smartphone using FTP for user display. The framework was trained and evaluated on the PlantVillage dataset involving 38 disease categories for 14 different types of healthy and diseased crop leaves, achieving a maximum accuracy of 99.1%. This efficient and practical system demonstrates the potential of edge AI in precision agriculture by enabling on-device disease diagnosis without relying on cloud computing infrastructure.

1. Introduction

Farming is essential to every aspect of human life, including food production, clothing, medicine, antimicrobial treatments, and environmental sustainability. Plants are a fundamental food source for both humans and animals, and agriculture is closely linked to the economic structures of various nations. One of the factors contributing to reduced productivity is plant disease [1], which causes significant economic losses worldwide each year. For the past few decades, farmers have used a lot of herbicides and insecticides to get rid of weeds and pests. If applied sparingly, they might be a way to increase crop productivity. However, excessive use negatively impacts both human health and the overall quality of agricultural land. Pesticide and herbicide overuse contaminates groundwater and agricultural land. Numerous studies show that human ailments like diabetes, cancer, asthma, and neurological and reproductive issues are on the rise [2].

The primary factor influencing crops both qualitatively and quantitatively is climate change. Heatwaves, hailstorms, tornadoes, and droughts can all cause partial or complete crop

[*] Corresponding author.
E-mail address: adil.jalal.yousif@uodiyala.edu.iq
DOI: 10.24237/djes.2025.18308

loss and may provide a suitable environment for pests to develop. Plant diseases are becoming a major threat to global food security and are causing a significant decrease in food output as human society develops [3]. Thus, early detection of plant diseases can significantly lower food losses and reduce planting expenses. Infected plants typically exhibit visible lesions or marks on their leaves, stems, flowers, or fruits. Generally speaking, each disease or pest condition has a distinct visible pattern that can be used to diagnose abnormalities in a unique way. Typically, the leaves of plants serve as the main source for identifying plant diseases, and the majority of disease symptoms may start on the leaves [4]. The majority of plant disease research relies on photographs of plant leaves since they are typically used to visually identify different plant categories [5].

However, Artificial Intelligence (AI) and Deep Learning (DL) have achieved noteworthy development for many techniques and applications like medical imaging, pattern recognition, biometric systems, and smart surveillance. For instance, deep learning has enabled advanced classification in EEG-based motor imagery tasks [6], real-time respiratory disease detection [7], handwriting recognition [8], smart surveillance system [9], and biometric recognition systems [10].

The old methods of plant disease image detection mainly depend on manually extracting image features [11]. These methods were mostly dependent on human skill and frequently resulted in high recognition error rates when dealing with diverse imaging environments. With the express progress in deep learning techniques [12], these traditional methods are progressively replaced by automated systems that have the ability to learn complex patterns and forms from large datasets specified for this purpose [13]. DL algorithms, like Convolutional Neural Networks (CNNs), have reformed the technique of detecting and analyzing plant diseases. These models are capable of automatically extracting hierarchical features from raw image data without manual intervention [14]. Integrating DL approaches into crop health monitoring allows farmers to enhance productivity in addition to reducing the time and resources that are needed for input management.

However, the farmers still face several challenges in diagnosing and managing plant diseases especially in remote areas and resource-limited environments. Traditional detection methods depend on expert help that may not always be available, or manual inspection by hand that is time-consuming and inaccurate. Additionally, many AI-based methods depend on cloud connectivity [15], which is not practical for regions with limited or no internet access. The high cost of advanced diagnostic tools limits their use by small-scale farmers. These challenges emphasize the need for a cost-effective, portable, and internet-independent solution for low-latency disease detection.

To solve these problems, this paper proposes a smart system for automated plant disease detection. The suggested approach enables fast and cost-effective diagnosis without internet dependency to make it accessible and practical across various agricultural environments. The main contributions of this study are summarized as follows:

1- Proposes an edge computing framework using Jetson Nano for performing plant disease classification without the need for cloud connectivity.
2- Develop a deep learning model for efficient feature extraction and disease classification for ensuring a balance between accuracy and computational efficiency for resource-constrained environments.
3- Integrate an Android-based interface for capturing plant images to make the system user-friendly and portable.
4- Implements a simple prototype of the proposed system for providing a low-cost and standalone diagnostic tool that reduces the need for expensive hardware and internet access.

2. Related Work

The field of plant disease detection has increased significant attention from researchers in recent years with various approaches utilizing DL techniques. Early methods mainly depended on traditional image processing and manually feature extraction techniques, like color, texture, and shape analysis [16]. However, these methods must be adapted to specific conditions which make it not suitable for most natural environments. To overcome these limitations, researchers have investigated various techniques, including machine learning, deep learning, fuzzy logic, and genetic algorithms [17].

Based on the machine learning approach, R. Kumar et al. [18] introduced a cotton identification method using hybrid techniques. The authors evaluated several machine learning methods involving Support Vector Machine (SVM), Random Forest (RF), Decision Tree (DT), and Ensemble Learning (EL). The obtained accuracy was 94.5% on the Kaggle Cotton Disease. Another approach that used machine learning was developed by S. Ramesh et al. [19], where a Histogram of an Oriented Gradient (HOG) is used for feature extraction, while the Random Forest is used as the classifier for identifying between healthy and diseased crops.

M. Rajagopal et al. [20] proposed plant disease recognition by integrating fuzzy C-Means with machine learning. They used Legion Kernels and a parallel support vector machine combined with Fuzzy C-Means for image segmentation. The reported accuracy was 96.1% using the PlantVillage dataset. In [21], a corn rust disease detection method is suggested based on combining fuzzy logic with deep convolutional network VGG16. A threshold segmentation followed by fuzzy rules was used to detect the diseased leaf area of the processed image. The accuracy obtained was 89% using the four classes of corns in the PlantVillage dataset. The study in [22] presented an automatic classification technique for medicinal plant leaves using Particle Swarm Optimization (PSO)–based fuzzy C-means (PSO-FCM) and Gaussian Mixture Model (GMM) for segmentation. Vein, shape, edge-based, and texture features were extracted, and classification was performed using a multiple kernel parallel SVM.

The study in [23] introduced an automated apple disease detection method by integrating the genetic algorithm with machine learning. It integrates filtering techniques, correlation-based segmentation, and genetic algorithm optimization with multi-class support vector machine (M-SVM classification. Tested on the PlantVillage dataset, the approach achieved high accuracy, highlighting the impact of effective preprocessing.

Focusing on the genetic algorithm, D. Angayarkanni and L. Jayasimman [24] proposed a hybrid image recognition approach for early plant disease detection using CNN-based denoising, pixel-wise classification, and genetic algorithm-based feature selection. This method obtained 97.7% accuracy on the PlantVillage dataset. Using deep learning, V. Monigari et al. [25] presented a plant disease prediction method by exploring various pretrained models like VGG18, AlexNet, MobileNet, and ResNet50 in addition to building a custom CNN from scratch. They found that the pretrained CNNs outperformed the custom model in their experiment using the PlantVillage dataset. Similarly, M. Belmir et al. [26] offered a custom deep convolutional neural network model for plant disease classification using the PlantVillage dataset, using 14 crop types with 38 classes. The model got 98.01% training accuracy and 94.33% test accuracy.

The study in [27] suggested a lightweight CNN framework for automatic plant disease detection for enabling fast identification. The model was trained on 57,000 tomato leaf images belonging to nine classes that were captured in a natural setting without subtracting the background. It got 97.04% accuracy with an error ratio below 0.2, which demonstrates a high precision in disease detection. The study in [28] introduced an AI-based plant disease detection model using an ensemble method from VGG16, VGG19, ResNet101, and Inception V3, which got over 90% accuracy on 38 classes of the PlantVillage dataset. It includes an explainable AI method to provide interpretable visual explanations for further highlighting key image regions influencing predictions. A. Ashurov et al [29] presented a modified depth-wise CNN enhanced with squeeze-and-excitation blocks and optimized residual skip connections for plant disease detection. The reported accuracy was 98.0% using the PlantVillage dataset.

However, most of the existing studies on plant disease detection lack end-to-end hardware implementation and focus on software simulations. To address this challenge, the study in [30] implements an Arabica coffee plant disease detection system using CNN models and deploys it as a web application. MobileNetV2 was selected for deployment due to its lightweight architecture, achieving 99.0% accuracy when tested on five classes of the dataset. The trained model is stored in an hdf5 file and loaded onto a Streamlit-based web platform to process user-uploaded images for disease classification and treatment recommendations. A. Ahmed and G. Reddy [15] developed a mobile-based plant disease detection system using a CNN model trained on the PlantVillage dataset. The application is implemented with Kotlin Multiplatform and deployed on Android to enable farmers to capture or upload images of infected plant leaves to a cloud-based server for classification. However, the approaches in [30, 15] require an Internet connection. Table 1 summarizes the main results of the plant disease detection methods mentioned above.

After reviewing the relevant studies, it is clear that although many investigations introduce various solutions to specific challenges, some limitations still remain, particularly in remote agricultural areas, leaving significant gaps that remain unaddressed. The main challenges tackled by the proposed work include low power consumption and portability, both of which are essential for real-world applications.

Table 1: Summary of literature on plant disease detection

Study	Approach	Algorithms and techniques	Dataset	Target crops	Reported accuracy (%)	Hardware application
[18]	Machine learning	SVM, RF, DT, and EL	Kaggle Cotton Disease	Cotton	94.5	N/A
[19]	Machine learning	HOG and RF	Author-created	Papaya	Max 70.1	N/A
[20]	Machine learning and fuzzy C-Means	SVM and Legion Kernels	PlantVillage	Apple	96.1	N/A
[21]	Deep learning and fuzzy logic	Threshold segmentation and VGG16	PlantVillage	Corns	89.0	N/A
[22]	Machine learning, PSO, and fuzzy C-Means	GMM and SVM	PlantVillage	Tomato	85.5	N/A
[23]	Machine learning and genetic algorithm	SVM, filtering, and segmentation techniques	PlantVillage	Apple	Max 97.2	N/A
[24]	Deep learning, Machine learning, and genetic algorithm	CNN, SVM, and pixel-wise classification	PlantVillage	Potato and tomato	97.7	N/A
[25]	Deep learning	Various CNNs	PlantVillage	Diverse	Max 98.2	N/A
[26]	Deep learning	Custom CNN	PlantVillage	Diverse	94.3	N/A
[27]	Deep learning	Custom CNN	PlantVillage	Tomato	97.0	N/A
[28]	Deep learning and explainable AI	Various CNNs	PlantVillage	Diverse	92.3	N/A
[29]	Deep learning	Custom CNN	PlantVillage	Diverse	98.0	N/A
[30]	Deep learning	MobileNetV2	Arabica coffee leaf disease	Coffee	99.0	Smartphone and web-based application
[15]	Deep learning	Custom CNN	PlantVillage	Diverse	94.0	Smartphone and cloud server

3. Methodology

This section describes the proposed approach for detecting plant disease in the leaves of crops including dataset description, proposed deep learning model, and system design.

3.1 Dataset Description

The PlantVillage dataset [31] is a widely used benchmark for plant disease detection that provides a diverse collection of images of healthy and diseased leaves across various crop species. It consists of approximately 54,303 images categorized into 38 classes that belong to different plant species and disease conditions as presented in Table 2. The dataset contains crops like tomatoes, potatoes, apples, and more, with specific diseases such as bacterial spot, early blight, and late blight. The images in this dataset are in RGB format and vary in resolution, but they can be resized (e.g., 224×224) for compatibility issues with deep learning models. This dataset is considered a valuable resource to train the machine learning models for classifying plant diseases and enabling automated disease diagnosis in agricultural field. Figure 1 shows different types of diseased and healthy plant leaves from this dataset. The dataset is publicly available at the following link. https://data.mendeley.com/datasets/tywbtsjrjv/1.

The PlantVillage dataset is selected in this study due to its comprehensiveness, quality, and wide acceptance in the plant disease detection research community. This dataset covers various crop species and disease types, which makes it a valuable benchmark for training and evaluating deep learning models in a controlled setting. The large number of labeled images ensures that the model can learn different visual patterns effectively. Additionally, the consistency of image quality helps the model obtain high accuracy.

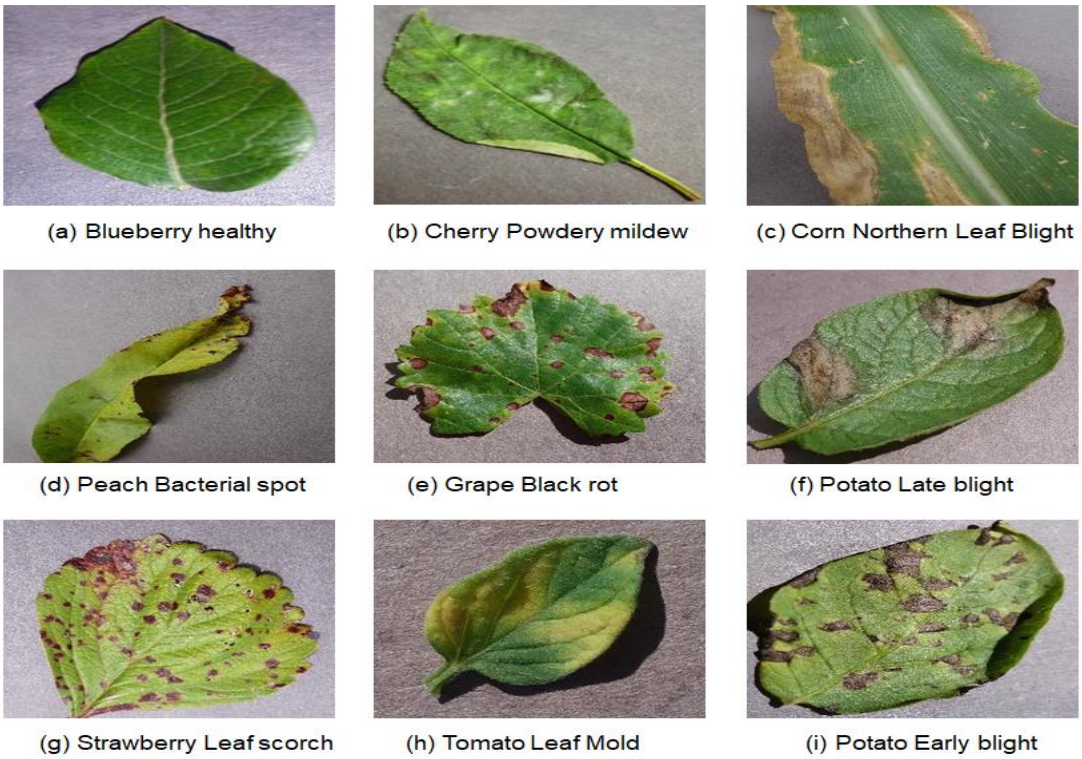

Figure 1. Different samples of the PlantVillage dataset

Table 2: Number of images per disease and crop category in the PlanetVillage dataset

Class	No. of images	Class	No. of images
Apple Apple scab	630	Pepper bell healthy	1478
Apple Black rot	621	Potato Early blight	1000
Apple Cedar apple rust	275	Potato healthy	152
Apple healthy	1645	Potato Late blight	1000
Blueberry healthy	1502	Raspberry healthy	371
Cherry healthy	850	Soybean healthy	5090
Cherry Powdery mildew	1034	Squash Powdery mildew	1835
Corn Cercospora leaf spot Gray leaf spot	513	Strawberry healthy	456
Corn Common rust	1192	Strawberry Leaf scorch	909
Corn healthy	1162	Tomato Bacterial spot	2130
Corn Northern Leaf Blight	985	Tomato Early blight	1000
Grape Black rot	1180	Tomato healthy	1591
Grape Esca (Black Measles)	1383	Tomato Late blight	1909
Grape healthy	423	Tomato Leaf Mold	952
Grape Leaf blight (Isariopsis Leaf Spot)	1076	Tomato Septoria leaf spot	1771
Orange Haunglongbing (Citrus greening)	4507	Tomato Spider mites Two-spotted spider mite	1676
Peach Bacterial spot	1797	Tomato Target Spot	1404
Peach healthy	360	Tomato Tomato mosaic virus	373
Pepper bell Bacterial spot	997	Tomato Tomato Yellow Leaf Curl Virus	5357

3.2 Proposed Deep Learning Model

The aim of the suggested DL method is to effectively detect leaf diseases in plants based on integrating a feature extraction unit with a classification network. This approach employs a pretrained CNN to extract important features from plant leaf images in addition to effective classification for low-latency applications on resource-constrained devices. The pipeline of the proposed model involves data preprocessing, data splitting, feature extraction, classification, and final prediction. The process starts with a dataset containing images of plant leaves for different diseases or plants remaining in a healthy form. Before training, these images go through a preprocessing stage involving resizing and normalization. After preprocessing, the dataset is split into two subsets: 80% is allocated for training, and the remaining data 20% is used for testing to evaluate the generalization ability of the presented model on unseen data. Feature extraction is implemented using a pretrained CNN due to its effectivity in image recognition field. Several CNN architectures including AlexNet, GoogleNet, and DenseNet, are investigated for this task. The architecture of these models is adapted by replacing its original 1000-class classification layer with a 2D global average pooling layer for reducing the spatial dimensions and obtaining 1D image feature vectors that represent the most important information of the input images.

Then, the extracted features are passed through a classification network, which consists of two fully connected layers with ReLU activation function. A dropout layer is placed between these layers for preventing overfitting and ensure better generalization through different plant species and disease types. The final output layer consists of a softmax function for assigning probability scores to each class in the dataset and obtaining the predicted label for the input leaf image. After training and evaluating the model, it can be deployed for plant disease classification for allowing precise analysis in agricultural applications. Figure 2 presents an overview of the proposed deep-learning framework.

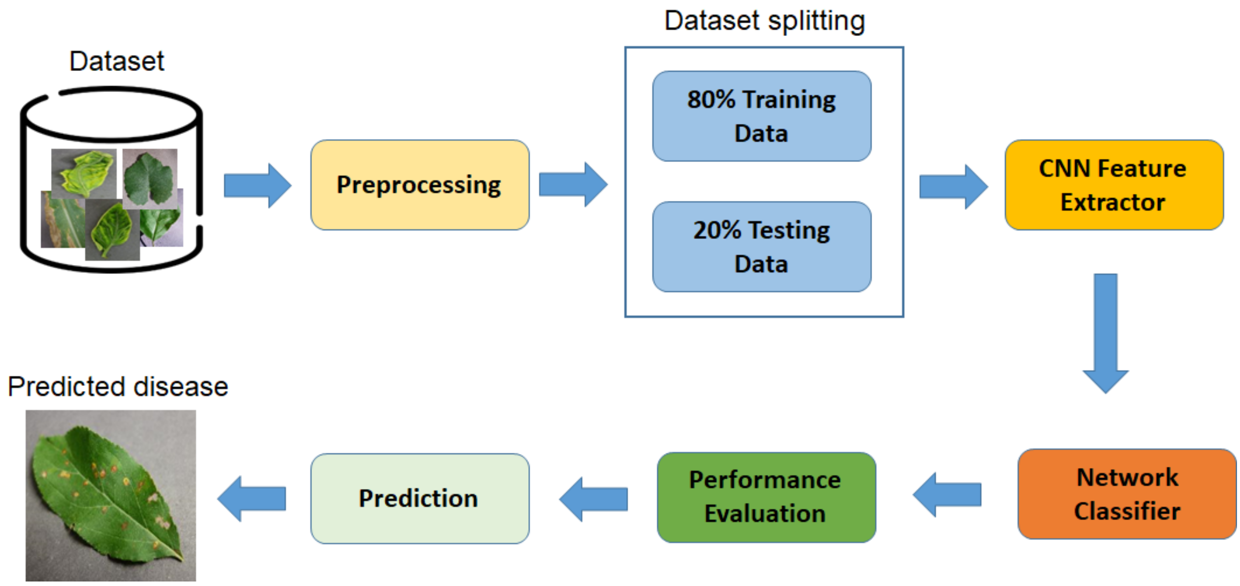

Figure 2. Workflow of the proposed deep learning framework for plant disease detection

3.3 System Design

The proposed plant disease detection system is designed for efficient and low-latency classification of plant leaf diseases using an embedded AI approach. The system consists of three main components: Jetson Nano as the processing unit, an Android smartphone as the user interface, and a wireless access point for WiFi communication between the devices as shown in Figure 3. The smartphone captures images of plant leaves and transmits them to the Jetson Nano via KDE Connect software (stands for K Desktop Environment), which enables seamless wireless communication between Android and other operating systems without requiring Internet connectivity. This tool allows secure file transfer over a local network. In this system, the KDE application is configured on both the Android device and the Jetson Nano (running a Linux-based OS) to enable automatic and encrypted file sharing via its TCP/IP over the WiFi channel. Once the user selects or captures an image on the phone, it is shared directly to a designated directory on the Jetson Nano through KDE Connect's "Send File" feature. At the core of the detection system is a deep neural model, designed for efficient execution on the Jetson Nano. The received image undergoes preprocessing, including resizing and normalization, before being passed through the deep learning model for disease classification. The model predicts the disease category, and the predicted label is texted on the image to provide a visual diagnosis. For result transmission, an FTP (File Transfer Protocol) server running on Jetson Nano facilitates the transfer of the processed image back to the smartphone. Finally, the image is displayed to the user for providing an intuitive interface for real-time plant disease assessment. After inference, the annotated image (with prediction) is saved to the shared FTP directory, which the smartphone accesses using a standard FTP client to retrieve the result. This approach ensures fast and direct communication over the local network.

Figure 4 shows the hardware components of the proposed system involving a Jetson Nano, a 5V battery, a wireless access point, and a smartphone. This hardware setup ensures an efficient and portable plant disease detection system, capable of operating in both indoor and outdoor agricultural environments with minimal power consumption and reliable wireless connectivity.

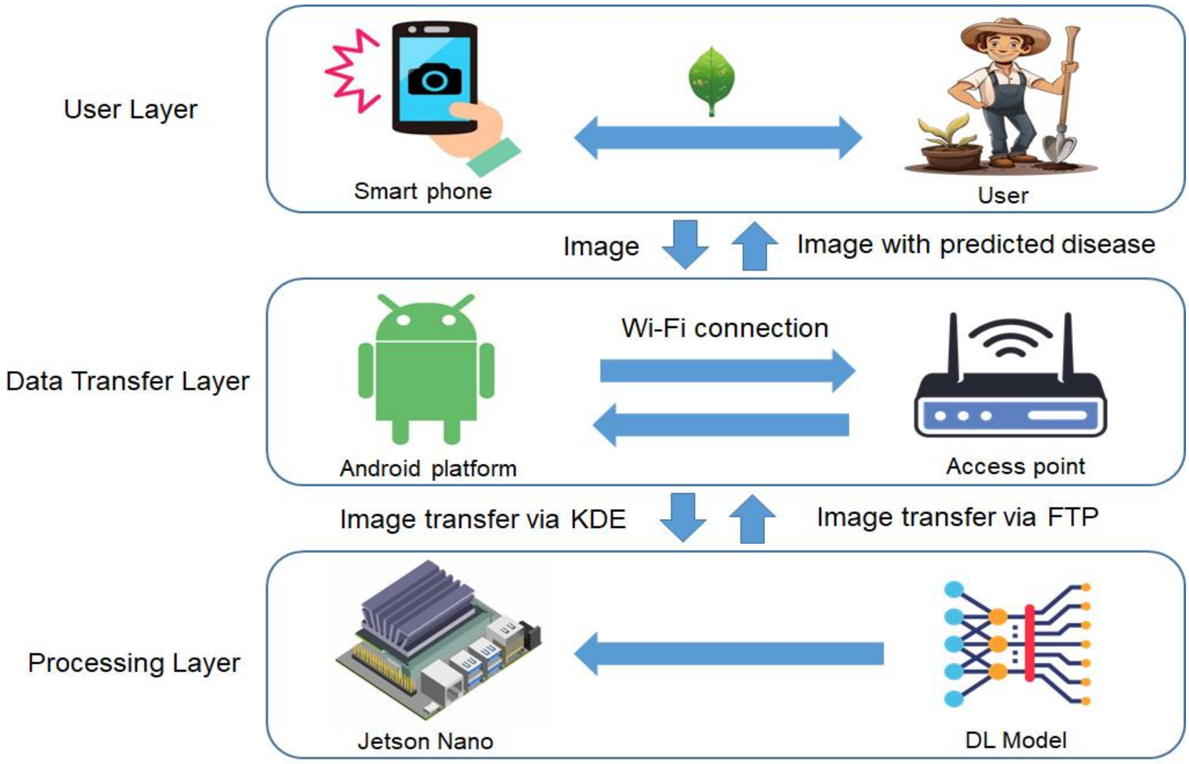

Figure 3. Architecture of the proposed system for Plant disease detection

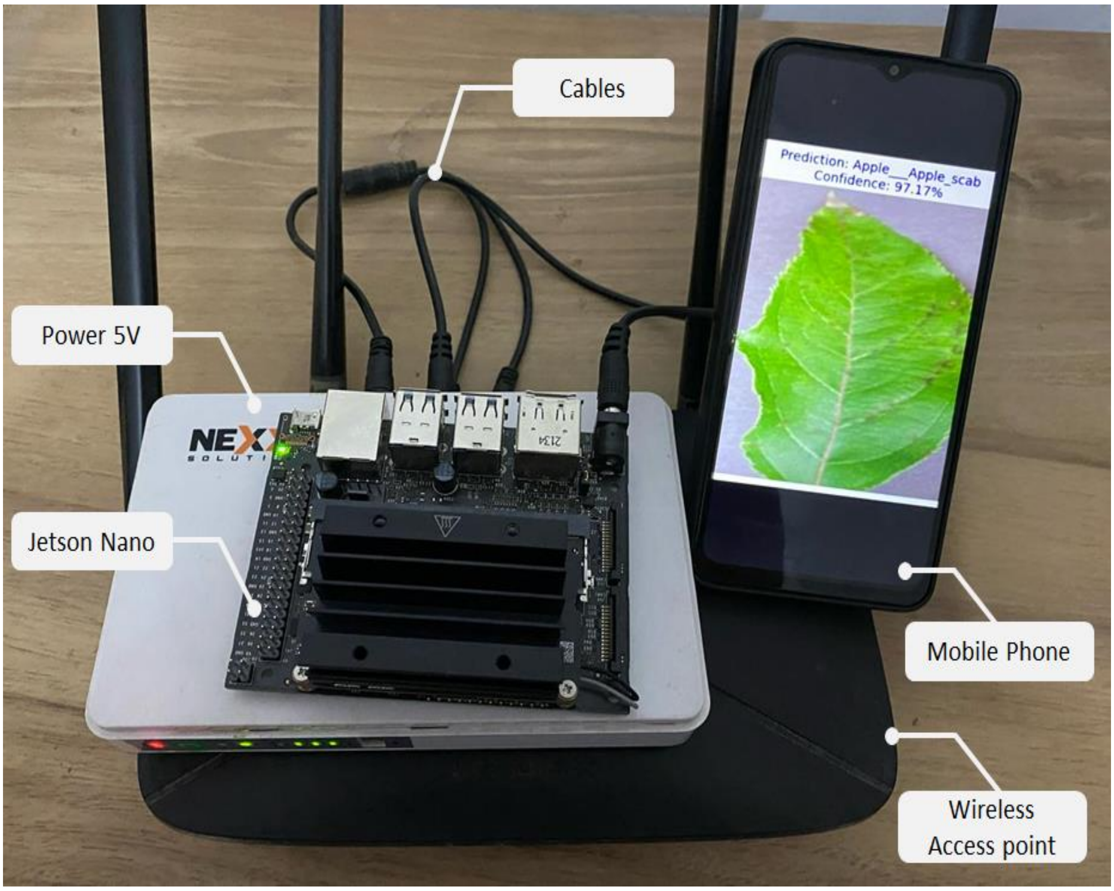

Figure 4. Hardware components of the proposed system

The overall process is described in Algorithm 1, which outlines the steps for capturing, processing, and classifying plant disease images.

Algorithm 1: WiFi-based plant disease detection
Input: Image captured from an Android phone
Output: The image with the prediction result
Step 1: Capture an image using the camera of the Android phone.
Step 2: Send the captured image to Jetson Nano using KDE Connect.
Step 3: Store the received image temporarily on the Jetson Nano.
Step 4: Preprocess the received image by resizing and normalizing it.
Step 5: Feed preprocessed image to the deep learning model for inference.
Step 6: Overlay the predicted disease label on the image.
Step 7: Establish an FTP connection between Jetson Nano and the phone.
Step 8: Transfer the processed image to a designated folder on the phone.
Step 9: Repeat the process for the next incoming image.

4. Results and Discussion

This section presents the performance evaluation of the proposed plant disease detection system by analyzing the effectiveness of the deep learning model and the overall system efficiency. The results are assessed based on various evaluation metrics including classification accuracy, precision, recall, F1 score, model parameters, memory usage, and inference time. The proposed model is trained using the standard version of the PlantVillage dataset, which includes all 38 categories, and tested on unseen images. The model is implemented using the PyTorch deep learning framework and trained on a laptop with an NVIDIA RTX 3060 GPU, Core i7 processor, and 16 GB RAM. The final trained model is deployed on the NVIDIA Jetson Nano, which includes a 4 GB LPDDR4 RAM, 128-core Maxwell GPU, and Quad-core ARM Cortex-A57 CPU. This device works with 5 volts, so it can be considered an energy-efficient edge solution and a power-saving option in this field.

To choose the best model that is more compatible with the Jetson Nano based on its computing power, different CNNs involving GoogleNet, VGG19, ResNet18, ResNet101, AlexNet, DenseNet121, and DenseNet169 are evaluated in terms of accuracy and computational cost as reported in Table 3 and Table 4, respectively. Figure 5 shows the training/testing loss and accuracy across 100 epochs utilizing the DenseNet169 model. The hyperparameters that are used for model training after tuning are presented in Table 5.

Table 3: Accuracy analysis of proposed method (best results in bold).

Model	F1	Precision	Recall	Accuracy
GoogleNet	9610	9653	9579	97.01
VGG19	96.34	96.87	95.97	97.19
AlexNet	9698	9737	9663	97.61
ResNet18	9692	9711	9675	97.94
ResNet101	9805	9823	9791	98.80
DenseNet121	9833	9843	9824	98.88
DenseNet169	**9882**	**9894**	**9872**	**99.11**

Table 4: Computational cost analysis of proposed method (best results in bold).

Model	Para. (M)	Memory usage (MB)	Inference speed on Jetson Nano (ms)	Training time using RTX 3060 (minutes)
GoogleNet	6.1	833.0	86	4.1
VGG19	20.3	955.3	228	6.5
AlexNet	**2.6**	**818.3**	**46**	**3.4**
ResNet18	11.4	892.1	69	3.8
ResNet101	43.5	954.9	252	7.0
DenseNet121	7.4	845.3	152	5.4
DenseNet169	13.3	889.1	198	6.1

Table 5: Hyperparameters configuration for model training

Hyperparameter	Configuration
Learning rate	0.0001
Batch size	64
Number of epochs	100
Dropout ratio	0.5
Size of hidden layers	512
Activation function	ReLU
Optimizer	Adam

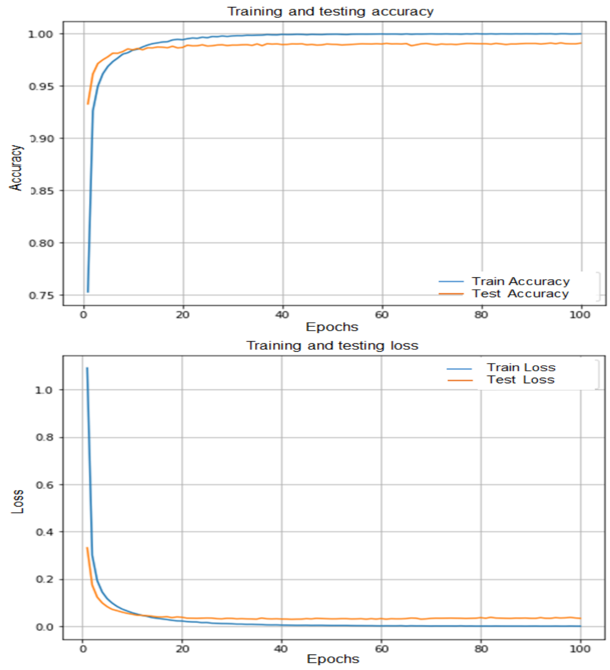

Figure 5. Training and testing performance: Accuracy and loss over epochs

The results in Tables 3 and 4 emphasize the effectiveness of AlexNet and DenseNet169 for plant disease detection. The lightweight nature of AlexNet allowed fast inference on the Jetson Nano by providing an excellent balance between efficiency and accuracy, with only 2.6M parameters, 818.3 MB memory usage, and a fast 46 ms inference speed, obtaining 97.61% accuracy. This makes it ideal for real-time applications on resource-limited devices. On the other hand, DenseNet169 slightly heavier (13.3M parameters, 889.1 MB memory usage), achieves the highest accuracy (99.11%). However, its 198 ms inference speed is slower, which makes it better suited for precision-focused tasks where slight delays are acceptable. According to Tables 3 and 4, this study offers multiple options to provide the desired trade-off between accuracy and inference speed for supporting various deployment priorities and aligning with the system's goal of offering a cost-effective, portable, and energy-efficient edge solution. Moreover, Table 6 presents a performance comparison of the proposed approach with existing methods for plant disease classification. The results for other methods (Table 6) are obtained directly from their original publications, as reported by the respective authors. This Table proves the effectiveness of the proposed approach in enhancing classification performance compared with other existing methods under similar experimental conditions.

Table 6: Performance comparison with existing studies using the PlantVillage dataset

Study	Year	Number of classes	Accuracy
[21]	2021	4	89.0
[15]	2021	38	94.0
[25]	2021	38	98.2
[26]	2023	38	94.3
[27]	2023	9	97.0
[24]	2023	subset	97.7
[28]	2024	38	92.3
[20]	2024	38	96.1
[29]	2025	38	98.0
Proposed (AlexNet)	2025	38	97.6
Proposed (DenseNet169)	2025	38	99.1

5. Limitations and Future Work

Although the proposed plant disease detection system reaches a high classification accuracy of 99.1% on the PlantVillage dataset, several limitations must be recognized to enhance the real-world applicability.

First, the proposed model is trained and evaluated using images taken under controlled settings like uniform lighting and backgrounds. This issue can limit the robustness of the system when applied in real agricultural environments, where the captured images may be affected by variable lighting, shadows, and weather conditions. To address this matter, future work will combine image augmentation techniques for the training data for simulating environmental variations and integrate preprocessing methods like histogram equalization and brightness normalization to improve adaptability in a variant environment.

Second, the performance of the proposed system can be reduced when faced with new plant species or disease types that are not included in the original dataset. The generalization capability is restricted by the utilized dataset, which handles only 14 crop species and 38 disease categories. Future improvements will explore open-set recognition methods and recurrent learning strategies to enable the system to identify and adapt to unknown plant diseases.

Third, although the system is adjusted for portable and real-time use on the Jetson Nano device, the scalability for broader agricultural applications requires further investigation. The framework should be retrained and validated on diverse datasets for covering other crop types, environments, and regions. This can be done by merging several datasets into a single and larger dataset. Furthermore, incorporation with additional sensors such as soil moisture or temperature monitors can enable multimodal analysis for more accurate diagnostics.

Lastly, to ensure long-term usability and efficiency, future research should explore ways for improving model performance such as integrating feedback mechanisms to include user input and expert corrections. Additionally, optimizing the system for Jetson Nano through model pruning to reduce computational cost and allow the use of a deeper neural network.

6. Conclusions

This paper presents an efficient, low-cost edge computing framework for plant disease detection utilizing the NVIDIA Jetson Nano and an Android-based interface for operation without internet dependency. The system achieves a high classification accuracy of 99.1% on the PlantVillage dataset and provides a portable, user-friendly diagnostic tool for farmers in remote areas. This framework demonstrates the potential of edge AI in precision agriculture for offering timely disease detection to enhance crop protection and food security.

References

[1] R. Rani, J. Sahoo, S. Bellamkonda, S. Kumar, and S. K. Pippal, “Role of Artificial Intelligence in Agriculture: An Analysis and Advancements with Focus on Plant Diseases,” IEEE Access, vol. 11, no. December, pp. 137999–138019, 2023, doi: 10.1109/ACCESS.2023.3339375.

[2] W. Zhou, M. Li, and V. Achal, “A comprehensive review on environmental and human health impacts of chemical pesticide usage,” Emerg. Contam., vol. 11, no. 1, p. 100410, 2025, doi: 10.1016/j.emcon.2024.100410.

[3] S. Deng, W. Wu, K. Zou, H. Qin, L. Cheng, and Q. Liang, “Efficient Real-Time Recognition Model of Plant Diseases for Low-Power Consumption Platform,” IEEE Trans. Artif. Intell., vol. 5, no. 5, pp. 2040–2054, 2024, doi: 10.1109/TAI.2023.3307662.

[4] E. Jain and A. Aneja, "Automated Detection and Classification of Bean Leaf Diseases using InceptionV3: A Deep Learning Approach," 2025 International Conference on Electronics and Renewable Systems (ICEARS), Tuticorin, India, 2025, pp. 1890-1895, doi: 10.1109/ICEARS64219.2025.10941547.

[5] C. H. Bock, K. S. Chiang, and E. M. Del Ponte, “Plant disease severity estimated visually: a century of research, best practices, and opportunities for improving methods and practices to maximize accuracy,” Trop. Plant Pathol., vol. 47, no. 1, pp. 25–42, 2022, doi: 10.1007/s40858-021-00439-z.

[6] Al-Saegh, A., Daood, A. and Ismail, M.H., 2024. Dual Optimization of Deep CNN for Motor Imagery EEG Tasks Classification. Diyala Journal of Engineering Sciences, pp.75-91.

[7] Mahmood, A.F., Alkababji, A.M. and Daood, A., 2024. Resilient embedded system for classification respiratory diseases in a real time. Biomedical Signal Processing and Control, 90, p.105876.

[8] Daood, A. M. A. R., A. L. I. Al-Saegh, and Ahlam Fadhil Mahmood. "Handwriting detection and recognition of Arabic numbers and characters using deep learning methods." Journal of Engineering Science and Technology 18.3 (2023): 1581-1598.

[9] Mohammed, Muhanad Ramzi, and Amar Daood. "Smart surveillance system to monitor the committed violations during the pandemic." International Journal of Computing and Digital System (2021): 1415-1426.

[10] Alhafidh, B.M.H., Hagem, R.M. and Daood, A.I., 2022, March. Face detection and recognition techniques analysis. In 2022 International Conference on Computer Science and Software Engineering (CSASE) (pp. 265-270). IEEE.

[11] L. Li, S. Zhang, and B. Wang, “Plant Disease Detection and Classification by Deep Learning - A

Review," IEEE Access, vol. 9, no. Ccv, pp. 56683–56698, 2021, doi: 10.1109/ACCESS.2021.3069646.

[12] A. J. Yousif and M. H. Al-Jammas, "Exploring deep learning approaches for video captioning: A comprehensive review," e-Prime - Adv. Electr. Eng. Electron. Energy, vol. 6, no. October, p. 100372, 2023, doi: 10.1016/j.prime.2023.100372.

[13] S. D. A. Ahmed, T. Abbas, and A. R. Abbas, "Review of Detecting Text generated by ChatGPT Using Machine and Deep-Learning Models: A Tools and Methods Analysis," Diyala J. Eng. Sci., vol. 18, no. 1, pp. 34–54, 2025, doi: 10.24237/djes.2025.18102.

[14] R. K. Sabri et al., "Diagnosing Gingiva Disease Using Artificial Intelligence Techniques," Diyala J. Eng. Sci., vol. 18, no. 2, pp. 179–190, 2025, doi: 10.24237/djes.2024.18211.

[15] A. A. Ahmed and G. Harshavardhan Reddy, "A Mobile-Based System for Detecting Plant Leaf Diseases Using Deep Learning," AgriEngineering, vol. 3, no. 3, pp. 478–493, 2021, doi: 10.3390/agriengineering3030032.

[16] Y. Fang and R. P. Ramasamy, "Current and prospective methods for plant disease detection," Biosensors, vol. 5, no. 3, pp. 537–561, 2015, doi: 10.3390/bios5030537.

[17] A. J. Yousif, G. J. Ahmed, and A. S. Abbood, "Design of linear phase high pass FIR filter using Weight Improved Particle Swarm Optimization," Int. J. Adv. Comput. Sci. Appl., vol. 9, no. 9, pp. 270–275, 2018, doi: 10.14569/ijacsa.2018.090936.

[18] R. Kumar et al., "Hybrid Approach of Cotton Disease Detection for Enhanced Crop Health and Yield," IEEE Access, vol. 12, no. June, pp. 132495–132507, 2024, doi: 10.1109/ACCESS.2024.3419906.

[19] S. R. Maniyath et al., "Plant disease detection using machine learning," Proc. - 2018 Int. Conf. Des. Innov. 3Cs Comput. Commun. Control. ICDI3C 2018, no. July 2019, pp. 41–45, 2018, doi: 10.1109/ICDI3C.2018.00017.

[20] M. Rajagopal, S. Kayikci, M. Abbas, and R. Sivasakthivel, "A novel technique for leaf disease classification using Legion Kernels with parallel support vector machine (LK-PSVM) and fuzzy C means image segmentation," Heliyon, vol. 10, no. 12, p. e32707, 2024, doi: 10.1016/j.heliyon.2024.e32707.

[21] M. Sibiya and M. Sumbwanyambe, "Automatic fuzzy logic-based maize common rust disease severity predictions with thresholding and deep learning," Pathogens, vol. 10, no. 2, pp. 1–17, 2021, doi: 10.3390/pathogens10020131.

[22] S. K. Pravin Kumar, M. G. Sumithra, and N. Saranya, "Particle Swarm Optimization (PSO) with fuzzy c means (PSO-FCM)–based segmentation and machine learning classifier for leaf diseases prediction," Concurr. Comput. Pract. Exp., vol. 33, no. 3, pp. 1–13, 2021, doi: 10.1002/cpe.5312.

[23] S. S. Slathia, A. Chhajer, and P. Gouthaman, "An Optimized Method for Segmentation and Classification of Apple Leaf Diseases Based on Machine Learning," Lect. Notes Networks Syst., vol. 173 LNNS, pp. 781–793, 2021, doi: 10.1007/978-981-33-4305-4_57.

[24] D. Angayarkanni and L. Jayasimman, "Recognition of Disease in Leaves Using Genetic Algorithm and Neural Network Based Feature Selection," Indian J. Sci. Technol., vol. 16, no. 19, pp. 1444–1452, 2023, doi: 10.17485/ijst/v16i19.218.

[25] V. Monigari, "Plant Leaf Disease Prediction," Int. J. Res. Appl. Sci. Eng. Technol., vol. 9, no. VII, pp. 1295–1305, 2021, doi: 10.22214/ijraset.2021.36582.

[26] M. Belmir, W. Difallah, and A. Ghazli, "Plant Leaf Disease Prediction and Classification Using Deep Learning," 2023 Int. Conf. Decis. Aid Sci. Appl. DASA 2023, pp. 536–540, 2023, doi: 10.1109/DASA59624.2023.10286672.

[27] S. Bensaadi and A. Louchene, "Low-cost convolutional neural network for tomato plant diseases classifiation," IAES Int. J. Artif. Intell., vol. 12, no. 1, p. 162, 2023.

[28] A. Oad et al., "Plant leaf disease detection using ensemble learning and Explainable AI," IEEE Access, vol. 12, no. November, pp. 156038–156049, 2024, doi: 10.1109/ACCESS.2024.3484574.

[29] A. Y. Ashurov et al., "Enhancing plant disease detection through deep learning: a Depthwise CNN with squeeze and excitation integration and residual skip connections," Front. Plant Sci., vol. 15, no. January, pp. 1–16, 2024, doi: 10.3389/fpls.2024.1505857.

[30] Y. Aufar, M. H. Abdillah, J. Romadoni, and others, "Web-based cnn application for arabica coffee leaf disease prediction in smart agriculture," J. Resti (Rekayasa Sist. Dan Teknol. Informasi), vol. 7, no. 1, pp. 71–79, 2023.

[31] D. P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," 2015, [Online]. Available: http://arxiv.org/abs/1511.08060.