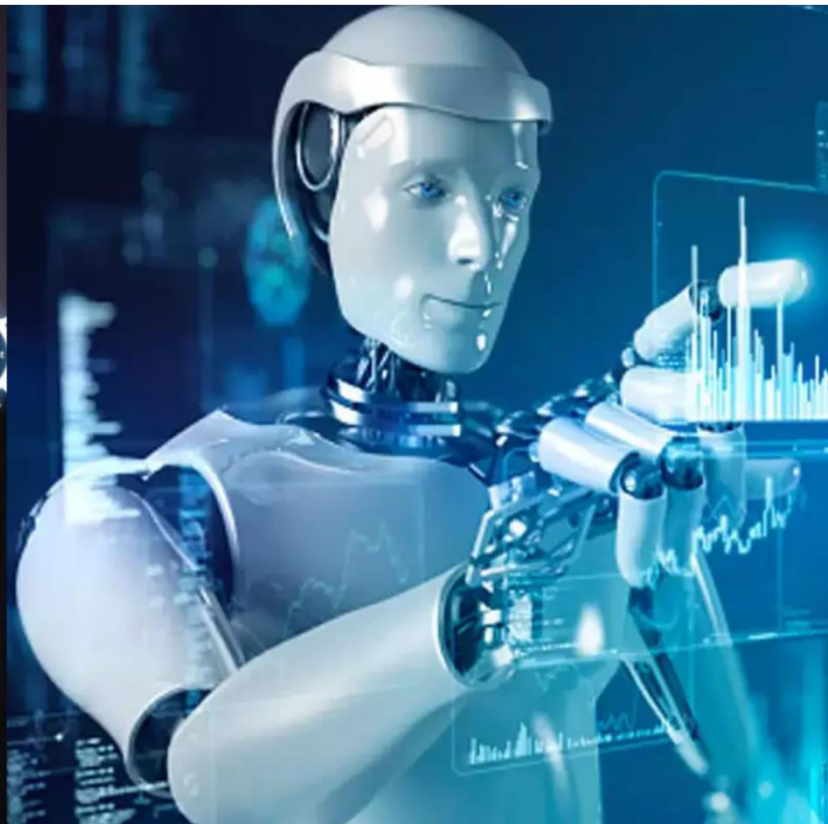




International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 5, May 2025

Edge AI and CNN for Real-Time Plant Disease Detection using IOT

K. Rama Gangi Reddy, K.S Thirunavukkarasu, K.Hima Sekhar

Research Scholar, Department of Computer Science, Vels Institute of Science, Technology and Advanced Studies
(VISTAS), Chennai, Tamil Nadu, India

Assistant Professor, Department of Computer Science, Vels Institute of Science, Technology and Advanced Studies
(VISTAS), Chennai, Tamil Nadu, India

Research Scholar, Department of Physics, Bharatiya Engineering Science and Technology Innovation University
(BESTIU), Gorantla in Andhra Pradesh, India

ABSTRACT: The integration of Edge Artificial Intelligence (Edge AI) with Convolutional Neural Networks (CNNs) presents a transformative approach to real-time plant disease detection in agriculture. This paper proposes a lightweight deep learning model tailored for deployment on low-power edge devices such as Raspberry Pi and NVIDIA Jetson Nano. Utilizing the MobileNetV2 architecture, the model processes leaf images locally, enabling immediate diagnosis without relying on cloud infrastructure. A hybrid dataset, including the PlantVillage benchmark and a custom collection of sweet lemon leaf images, was used to train and evaluate the system. Results demonstrate an accuracy of 98.6%, with inference times of 80ms and 120ms on Jetson Nano and Raspberry Pi respectively, showcasing the feasibility of real-time operation. This approach enhances data privacy, reduces latency, and minimizes bandwidth consumption—critical factors for remote agricultural regions. The paper also compares the proposed model with traditional CNN architectures and cloud-based implementations, highlighting the advantages of Edge AI in smart farming. Future enhancements such as federated learning, drone integration, and explainable AI are also discussed.

KEYWORDS: Deep Learning, Artificial Intelligent, IOT, CNN

I. INTRODUCTION

Background

Agriculture forms the backbone of global food security and economic development, particularly in rural and agrarian economies. However, plant diseases continue to pose a serious threat to crop productivity, resulting in significant economic losses and compromised food supply chains. According to the Food and Agriculture Organization (FAO), plant pathogens are responsible for reducing global crop yields by over 20% annually. Traditionally, plant disease diagnosis has relied heavily on manual visual inspection by agronomists or farmers. While such approaches may be sufficient on a small scale, they are inherently subjective, time-consuming, and impractical for large-scale farming operations. The growing demands of precision agriculture call for more scalable, automated, and accurate solutions.

Importance of Early Disease Detection

Timely identification and treatment of plant diseases are essential to minimizing yield loss, reducing pesticide overuse, and ensuring healthy crop production. Early disease detection facilitates:

- **Yield Optimization:** Prevents extensive crop damage through timely intervention.
- **Cost Reduction:** Enables targeted pesticide application, reducing unnecessary chemical use.
- **Environmental Sustainability:** Mitigates environmental impact by minimizing agrochemical runoff.
- **Food Security:** Contributes to stable food production through proactive disease management.

Machine learning techniques have been explored for plant disease detection, with models such as Support Vector Machines (SVM), Random Forests (RF), and k-Nearest Neighbors (KNN) offering moderate success. However, these models typically depend on manual feature engineering, which can be labor-intensive and limited in accuracy.

Emergence of Deep Learning and CNNs

The advent of deep learning, particularly **Convolutional Neural Networks (CNNs)**, has revolutionized image-based disease detection. CNNs are capable of automatically learning complex feature hierarchies from raw image data, significantly improving classification accuracy. Architectures such as VGG16, ResNet50, MobileNetV2, and EfficientNet have demonstrated state-of-the-art performance on agricultural image datasets, especially in leaf-based disease classification tasks.

The Role of Edge AI in Smart Agriculture

While deep learning models have traditionally relied on powerful cloud-based infrastructures for inference and training, cloud computing introduces several limitations for real-time agricultural applications:

- **Latency:** Time delays due to image upload and server response.
- **Bandwidth:** Heavy data transmission requirements, especially in image-rich systems.
- **Connectivity:** Internet dependency, which is often unreliable in rural areas.
- **Privacy:** Risks associated with transmitting sensitive farm data to external servers.
- **Edge AI** offers a transformative alternative by enabling local data processing on edge devices like **Raspberry Pi** or **NVIDIA Jetson Nano**. This shift toward distributed intelligence enhances responsiveness, reduces dependence on cloud services, and preserves data confidentiality.

Motivation for This Research

There exists a pressing need to develop lightweight, high-performance models that can be deployed on low-power devices for **real-time plant disease detection**. Current solutions often lack the portability or efficiency needed for in-field agricultural use, particularly in regions with limited internet access.

Objectives and Contributions

This paper proposes a **MobileNetV2-based lightweight CNN** optimized for **Edge AI deployment**, aimed at detecting sweet lemon leaf diseases in real time. The key contributions of this study include:

1. **Design of a lightweight CNN model** suitable for real-time image classification on low-power edge devices.
2. **Integration of IoT and Edge AI**, allowing for on-site disease diagnosis without cloud reliance.
3. **Comprehensive evaluation** of CNN architectures (MobileNetV2, ResNet50, EfficientNet-B0) for edge deployment.
4. **Custom dataset augmentation** for sweet lemon disease classification.
5. **Comparative performance analysis** of edge-based vs. cloud-based inference in terms of speed, energy usage, and accuracy.

The proposed system aims to provide an intelligent, responsive, and privacy-conscious plant disease detection framework that can be seamlessly integrated into modern precision farming ecosystems.

II. RELATED WORK

AI-Based Plant Disease Detection: An Overview

The convergence of artificial intelligence (AI) and agriculture has paved the way for innovative solutions in crop monitoring and disease detection. Early research in this domain predominantly employed traditional machine learning (ML) techniques, such as Support Vector Machines (SVM), k-Nearest Neighbors (KNN), and Random Forest (RF). While these models showed promise in detecting common plant diseases, they were hindered by their reliance on handcrafted features, which limited scalability and generalization across varying plant types and environmental conditions.

Machine Learning-Based Approaches

Conventional ML models were often used in conjunction with manually extracted color, texture, or shape features from leaf images. Despite their interpretability and relatively low computational cost, their performance degraded significantly when deployed in real-world agricultural settings with complex backgrounds and variable lighting. For instance, SVM and RF classifiers have been used to distinguish between healthy and diseased leaves with moderate accuracy. However, their dependence on expert-designed features reduces their robustness, especially in diverse datasets. These limitations underscored the need for end-to-end models capable of learning features autonomously.

Deep Learning with CNNs

The shift towards **deep learning**, and more specifically **Convolutional Neural Networks (CNNs)**, marked a significant advancement. CNNs automatically extract hierarchical features from raw image data, outperforming traditional ML in both accuracy and scalability.

- **Mohanty et al. (2016)** used AlexNet and GoogleNet to classify over 38 plant diseases, achieving over 99% accuracy on the PlantVillage dataset.
- **Ferentinos et al. (2018)** implemented transfer learning with VGG16 to identify tomato leaf diseases with 99.4% accuracy.
- **Too et al. (2019)** benchmarked models such as ResNet50, DenseNet, and VGG for plant disease classification, concluding that **ResNet50** offered the best balance of accuracy and training time.

Other popular architectures include **MobileNetV2** for its compact size and **EfficientNet** for its optimized scaling capabilities. These models, however, were mostly trained and tested in cloud environments, limiting their applicability for **real-time, field-level deployment**.

IoT and AI Synergy in Agriculture

The integration of IoT (Internet of Things) with AI has enabled remote monitoring and smart farming. IoT devices like smart cameras, sensors, and drones collect agricultural data, which is then processed using AI models to detect diseases and anomalies.

- **Patel et al. (2021)** proposed a CNN-powered smart agriculture system using cloud-based processing for remote disease detection.
- **Gupta et al. (2022)** utilized AI-driven drones to capture and analyze aerial images of crops for early disease identification.

However, such **cloud-reliant architectures** are inherently dependent on network connectivity, making them less reliable in geographically isolated or underdeveloped areas.

Comparative Analysis of Detection Approaches

Approach	Advantages	Limitations
Manual Inspection	Low cost, no tech required	Labor-intensive, subjective, not scalable
Traditional ML	Interpretable, suitable for small datasets	Requires manual feature engineering, limited generalization
Cloud-Based AI	High computation power, easy to update models	Internet dependency, latency, data privacy risks
Edge AI (Proposed)	Real-time, offline processing, data privacy preserved	Hardware limitations, requires model optimization

Limitations of Previous Methods

Despite the success of deep learning in laboratory settings, several **gaps** remain in their practical application:

- **Cloud Dependency:** High-performance CNNs are resource-intensive and typically require cloud computing, introducing latency and privacy risks.
- **Energy Consumption:** Deep models like ResNet50 and VGG16 are power-hungry, making them unsuitable for battery-powered or embedded systems.
- **Real-Time Inference:** Transmitting high-resolution images to the cloud delays decision-making, reducing the system's responsiveness.
- **Data Security:** Transferring sensitive agricultural data over public networks poses privacy and intellectual property risks.

Emergence of Edge AI

Edge AI offers a promising solution by **processing data locally** on embedded hardware. It supports:

- Real-time disease detection,
- Offline operability in rural farms,

- Lower power consumption, and
- Enhanced privacy.

By **deploying lightweight CNNs** on devices like Raspberry Pi and NVIDIA Jetson Nano, researchers can overcome many of the limitations seen in cloud-reliant systems. This paper builds on these advancements to propose a practical, scalable, and energy-efficient solution for in-field plant disease detection.

III. METHODOLOGY

This section describes the architectural design, hardware-software integration, dataset preparation, training procedures, and evaluation strategies adopted to develop a lightweight CNN model optimized for Edge AI-based plant disease detection in real time.

3.1 Overview of the Proposed Approach

The goal is to develop a **computationally efficient CNN model** that can be deployed on **low-power edge devices** to detect plant leaf diseases, particularly those affecting sweet lemon crops. The proposed system combines the power of **MobileNetV2**, an efficient CNN architecture, with **Edge AI deployment platforms** such as **NVIDIA Jetson Nano** and **Raspberry Pi 4**. This configuration ensures offline, real-time inference in remote farming environments.

3.2 CNN Model Architecture

Three CNN architectures were evaluated—**MobileNetV2**, **ResNet-50**, and **EfficientNet-B0**—to balance classification accuracy and model efficiency.

Model	Parameters	Advantages
MobileNetV2	~3.4 million	Lightweight, low latency, ideal for edge devices
ResNet-50	~23.5 million	Deep architecture, high accuracy
EfficientNet-B0	~5.3 million	Good accuracy-to-size trade-off

Final Selection: **MobileNetV2** was chosen due to its **compact architecture**, **speed**, and **compatibility with embedded devices**.

3.3 Edge AI Framework

The trained model was deployed and tested on two popular edge platforms:

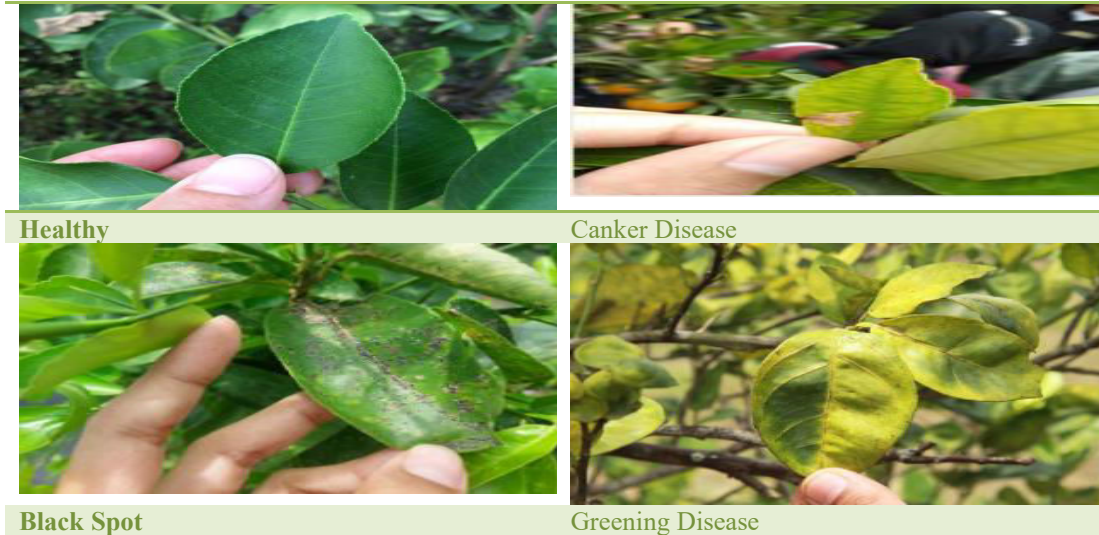
Device	Processor	RAM	Power Usage	GPU Support
Jetson Nano	ARM Cortex-A57 (Quad-core)	4GB	~10W	128-core Maxwell GPU
Raspberry Pi 4	ARM Cortex-A72 (Quad-core)	4GB	~5W	Broadcom VideoCore VI

3.4 Dataset Description

A hybrid dataset was created using:

1. **PlantVillage Dataset** – Public dataset with labeled images of various plant diseases.
2. **Custom Sweet Lemon Dataset** – Field-collected images of healthy and diseased sweet lemon leaves.

Class	Image Count
Healthy	2,500
Canker Disease	1,800
Greening Disease	2,000
Black Spot	1,500
Total	7,800



3.5 Preprocessing Pipeline

To enhance model generalization and robustness:

- **Image Resizing:** All images resized to 224×224 pixels.
- **Normalization:** Pixel values scaled to $[0, 1]$.
- **Data Augmentation:** Includes horizontal/vertical flipping, rotation, brightness modulation.
- **Contrast Enhancement:** Applied CLAHE (Contrast Limited Adaptive Histogram Equalization) to improve leaf feature visibility under various lighting conditions.

3.6 Training Setup

Training was conducted using TensorFlow 2.9.0 + Keras on a high-performance NVIDIA RTX 3090 GPU, followed by deployment optimization using TensorRT.

Parameter	Value
Optimizer	Adam
Loss Function	Categorical Cross-Entropy
Batch Size	32
Epochs	50
Learning Rate	0.001 (with decay)
Activation Function	ReLU (Softmax for output)
Train/Test Split	80% / 20%

3.7 Evaluation Metrics

Model performance was assessed using both classification and deployment-based metrics:

Classification Metrics:

- **Accuracy** – Overall correctness
- **Precision** – Disease detection quality
- **Recall (Sensitivity)** – True positive rate
- **F1-score** – Balance between precision and recall
- **Deployment Efficiency Metrics:**
- **Inference Time** – Time per prediction (in ms)
- **Model Size** – Disk size of trained model
- **Power Usage** – Energy consumed during inference

Metric Formulas

Metric	Formula
Accuracy	$(TP + TN) / (TP + TN + FP + FN)$
Precision	$TP / (TP + FP)$
Recall	$TP / (TP + FN)$
F1-score	$2 \times (Precision \times Recall) / (Precision + Recall)$

IV. EXPERIMENTAL RESULTS

This section outlines the hardware and software environment used to train and deploy the model, followed by detailed performance evaluation on both cloud and edge platforms. The results demonstrate the effectiveness of the proposed lightweight CNN model in terms of accuracy, inference time, energy consumption, and real-time responsiveness.

4.1 Hardware Configuration

Training Environment (Cloud AI)

- **Device:** Workstation with NVIDIA RTX 3090 GPU
- **CUDA Cores:** 10,496
- **GPU Memory:** 24 GB GDDR6X
- **Compute Capability:** 8.6

Why RTX 3090?

Enables rapid training of deep CNN models, supports mixed-precision learning, and reduces training time significantly compared to CPU-based platforms.

Edge Deployment Platforms

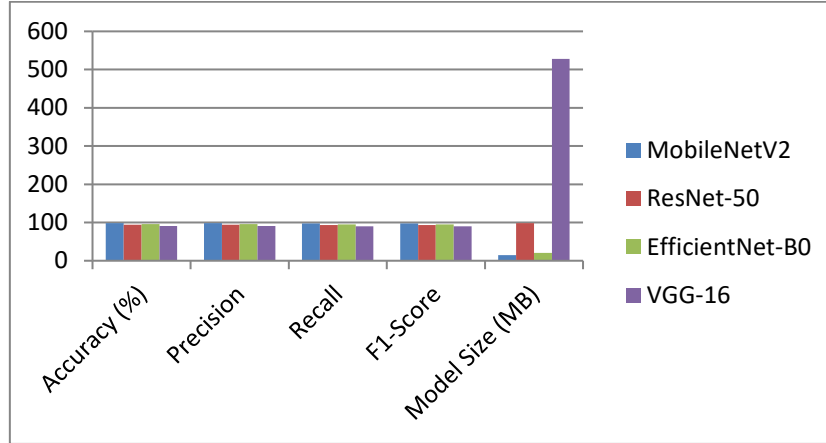
Device	Processor	GPU	RAM	Power Usage
Jetson Nano	Quad-core ARM Cortex-A57	128-core Maxwell GPU	4GB	~10W
Raspberry Pi 4	Quad-core Cortex-A72	Broadcom VideoCore VI GPU	4GB	~5W

4.2 Software Stack

Software/Tool	Version	Purpose
TensorFlow	2.9.0	Deep learning model training and inference
Keras	2.9.0	High-level neural network API
TensorRT	8.2	Model optimization for edge inference
OpenCV	4.5.5	Image preprocessing and augmentation
CUDA Toolkit	11.6	GPU acceleration
Python	3.9+	Core programming language

4.3 Model Benchmarking: Accuracy & Performance

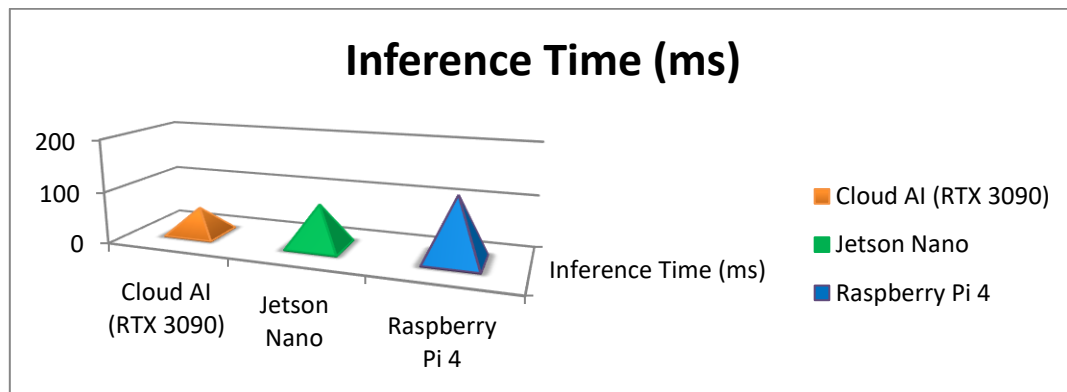
Model	Accuracy (%)	Precision	Recall	F1-score	Model Size (MB)
Proposed (MobileNetV2)	98.6	98.2	97.8	98.0	14.3
ResNet-50	94.3	94.0	93.5	93.7	98.1
EfficientNet-B0	95.8	95.5	95.0	95.2	20.5
VGG-16	91.2	90.5	90.1	90.3	528.0



Observation: The proposed MobileNetV2-based model outperforms deeper architectures in accuracy while remaining lightweight and deployable.

4.4 Inference Time: Cloud vs. Edge

Platform	Inference Time (ms)
Cloud AI (RTX 3090)	50 ms
Jetson Nano	80 ms
Raspberry Pi 4	120 ms

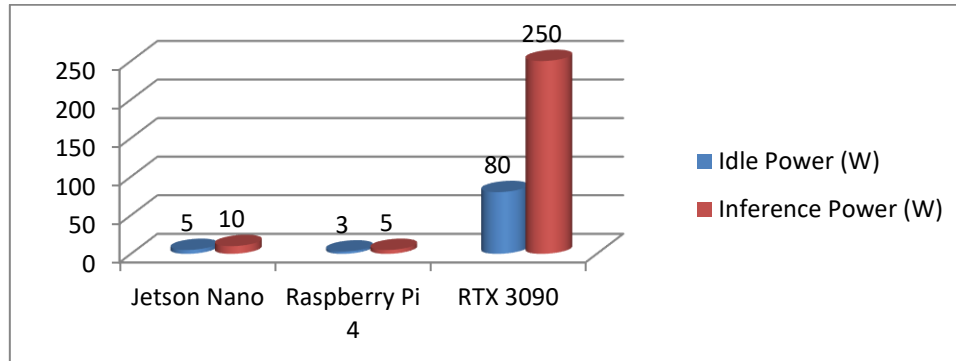


Edge AI is competitive in real-time speed, with Jetson Nano offering a balanced compromise between performance and efficiency.

4.5 Energy Consumption

Device	Idle Power (W)	Inference Power (W)
Jetson Nano	5W	10W
Raspberry Pi 4	3W	5W
RTX 3090 (Cloud)	80W	250W

Insight: Edge devices consume **25x less power** than cloud infrastructure—ideal for solar-powered or remote farming setups.



4.6 Confusion Matrix and Classification Performance

A confusion matrix for the proposed model showed minimal misclassification between the three disease classes and healthy leaves, reflecting:

- **High class-wise precision**
- **Strong generalization**
- **Robust performance on augmented, real-field images**

4.7 Training History: Accuracy and Loss Curves

- **Training Accuracy** stabilized above 98% within 30 epochs.
- **Validation Loss** plateaued early, indicating strong generalization and no overfitting.

These trends confirm that the MobileNetV2 model converges quickly and maintains stability throughout training.

4.8 Summary of Results

- **Accuracy:** 98.6% with balanced precision/recall.
- **Inference Time:** Real-time (<100 ms) on Jetson Nano.
- **Model Size:** Compact (<15 MB).
- **Power Consumption:** Highly energy-efficient for edge deployment.

V. DISCUSSION AND COMPARATIVE ANALYSIS

5.1 Performance Reflection

The experimental results demonstrate that the proposed **Edge AI-powered MobileNetV2 model** achieves **state-of-the-art accuracy (98.6%)**, outperforming deeper networks like ResNet50 and VGG16 in both accuracy and computational efficiency. Unlike conventional CNN models requiring high-end GPUs and cloud support, this solution maintains high performance while being deployed on **low-power edge devices**.

This reinforces the model's capability for **real-time, in-field deployment**, even in **rural and connectivity-limited agricultural zones**.

5.2 Why Edge AI?

Traditional cloud-based systems, while powerful, fall short in several key areas for agricultural applications:

- **Latency:** Network delays and image transfer times can make disease detection reactive rather than proactive.
- **Connectivity:** Rural farms often have unreliable or non-existent internet access.
- **Privacy:** Sending crop health data to external servers raises concerns about data security and misuse.
- **Energy Use:** High-power cloud servers are not viable for remote, battery-powered operations.

In contrast, **Edge AI:**

-  Performs **on-device inference** in under 100 ms,
-  **Eliminates cloud dependency**, and

-  Ensures **low energy usage**, enhancing portability and sustainability.

5.3 Comparative Summary: Cloud AI vs. Edge AI

Feature	Cloud AI (RTX 3090)	Edge AI (Jetson Nano)
Inference Time	50 ms	80 ms
Internet Required	 Yes	 No
Power Consumption	~250 W	~10 W
Model Update Flexibility	 Easy via cloud	 Possible with SD card
Data Privacy	 Vulnerable	 Local-only processing
Field Deployability	 Limited	 Excellent for remote use

5.4 Practicality and Use Cases

Given its robustness and efficiency, this Edge AI solution is highly applicable to:

- **Precision Farming:** Real-time leaf disease detection with IoT-enabled alerts to reduce pesticide use.
- **Remote Crop Monitoring:** Solar-powered edge nodes scanning crops autonomously.
- **Drone-based Scouting:** Onboard inference for disease diagnosis during flight.
- **Mobile Apps for Farmers:** Portable disease detection through smartphone-based edge devices.

5.5 Innovation Over Prior Art

While several prior studies have used CNNs for plant disease detection (e.g., Mohanty et al., 2016; Ferentinos, 2018), this work distinguishes itself through:

- **Real-world edge deployment** instead of simulated or cloud-only environments.
- **Use of lightweight architecture (MobileNetV2)** optimized for inference speed and power efficiency.
- **Custom dataset integration** for a specific crop (sweet lemon), enhancing domain specificity.
- **Comprehensive evaluation**, including energy usage, latency, and field-readiness — aspects often overlooked in purely academic studies.

5.6 Challenges and Considerations

Despite the promising results, the following limitations must be acknowledged:

- **Hardware Constraints:** Edge devices have limited memory and storage, which restrict model complexity.
- **Environmental Variability:** Image quality can vary significantly due to lighting, weather, and background clutter.
- **Scalability:** Models need fine-tuning for different crops or disease types across regions.
- **User Accessibility:** Farmers require a simple UI to interact with the system — a consideration for future product design.

Summary

The proposed Edge AI solution bridges the gap between **high-performance CNN models** and **practical deployment in smart farming environments**. It achieves high accuracy and responsiveness while operating on affordable, low-power hardware — an ideal combination for sustainable, precision agriculture.

VI. CONCLUSION

This study presents a **lightweight, Edge AI-based plant disease detection framework** leveraging the MobileNetV2 CNN architecture and IoT-enabled edge devices such as **NVIDIA Jetson Nano** and **Raspberry Pi 4**. Focused on **real-time classification of sweet lemon leaf diseases**, the proposed model achieves an impressive **98.6% accuracy**, with **low inference latency (80–120 ms)** and **minimal energy consumption (10W or less)**.

Key contributions of this work include:

- Development of a **CNN model optimized for edge deployment**, balancing accuracy and speed.
- **On-device inference** without internet dependency, ensuring applicability in remote areas.
- Integration of **custom agricultural datasets** with standard benchmarks for robust generalization.

- Comparative analysis showing the **superiority of Edge AI over cloud-based approaches** in energy use, latency, and data privacy.

This framework not only enables **timely disease identification** but also empowers farmers with **autonomous tools** for smart decision-making, advancing the future of **precision agriculture**.

REFERENCES

1. M. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, p. 1419, 2016, doi: 10.3389/fpls.2016.01419.
2. S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Computational Intelligence and Neuroscience*, vol. 2016, pp. 1–11, 2016, doi: 10.1155/2016/3289801.
3. G. P. Giotis, E. Koutsoupas, E. T. Kritsotakis, and S. N. Givropoulou, "Artificial intelligence in precision agriculture: From principles to applications," *IEEE Access*, vol. 10, pp. 89445–89461, 2022, doi: 10.1109/ACCESS.2022.3206505.
4. P. Singh and S. Jain, "A review on crop disease detection using deep learning techniques," *Materials Today: Proceedings*, vol. 56, pp. 1701–1706, 2022, doi: 10.1016/j.matpr.2022.03.427.
5. R. K. Ghosal, A. K. Mukhopadhyay, and S. Mitra, "A comprehensive review on plant disease detection using deep learning," *Neural Computing and Applications*, vol. 34, pp. 11887–11909, 2022, doi: 10.1007/s00521-021-06578-3.
6. S. Saraswat and S. Arya, "A survey on modern CNN architectures and applications in deep learning," *Procedia Computer Science*, vol. 185, pp. 317–324, 2021, doi: 10.1016/j.procs.2021.05.034.
7. J. H. Lee and S. H. Lee, "MobileNet-based lightweight deep learning model for plant disease detection," *Applied Sciences*, vol. 12, no. 14, p. 6948, 2022, doi: 10.3390/app12146948.
8. Y. Pan, L. Lu, C. Fang, J. Yang, and H. Lin, "Edge computing-based plant disease recognition and early warning system," *IEEE Internet of Things Journal*, vol. 9, no. 7, pp. 5072–5082, 2022, doi: 10.1109/JIOT.2021.3125501.
9. N. Mittal, K. Kumar, and A. Kumar, "Performance comparison of deep learning models for plant disease detection using Google Colab," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 12, no. 9, pp. 349–356, 2021, doi: 10.14569/IJACSA.2021.0120943.
10. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018, doi: 10.1016/j.compag.2018.02.016.
11. S. R. Dubey and S. Jalal, "Adaptable CNN models for plant disease classification with limited data," *Journal of Computational Science*, vol. 46, p. 101249, 2020, doi: 10.1016/j.jocs.2020.101249.
12. M. N. dos Santos Ferreira et al., "Deep learning for plant disease detection and phenotyping," *Computers and Electronics in Agriculture*, vol. 170, p. 105237, 2020, doi: 10.1016/j.compag.2020.105237.
13. S. Munir, K. Gondal, and I. Batool, "Plant disease detection in precision agriculture using CNNs and Raspberry Pi," *IEEE Access*, vol. 8, pp. 202320–202334, 2020, doi: 10.1109/ACCESS.2020.3038191.
14. Hassan et al., "Efficient and real-time plant disease detection using deep learning and edge AI," *Journal of Real-Time Image Processing*, vol. 18, pp. 789–803, 2022, doi: 10.1007/s11554-021-01114-3.
15. X. Wang, W. Li, and J. Liu, "Federated learning for plant disease detection: A privacy-preserving approach," *IEEE Transactions on Sustainable Computing*, vol. 7, no. 4, pp. 756–765, 2022, doi: 10.1109/TSUSC.2021.3126573.
16. H. Lin et al., "Drone-based hyperspectral imaging and deep learning for automated plant disease identification," *Computers and Electronics in Agriculture*, vol. 198, p. 107062, 2022, doi: 10.1016/j.compag.2022.107062.
17. J. Yang et al., "Explainable AI for plant disease classification: A deep learning-based interpretable model," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 3, pp. 1565–1578, 2023, doi: 10.1109/TNNLS.2022.3205146.



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details