



International Journal of Advanced Research in Education and Technology (IJARETY)

Volume 13, Issue 3, May - June 2026

Impact Factor: 9.971



An Intelligent Vision–Language Framework for Automated Plant Health Diagnosis and Context-Aware Care Recommendation using Local Large Language Models

Lavanya Arigela¹, Dr. Chiraparapu Srinivasrao²

PG Scholar, Department of Computer Science, S.V.K.P & Dr. K.S. Raju Arts and Science College (Autonomous),
Penugonda, Affiliated to Adikavi Nannaya University, Andhra Pradesh, India¹

Associate Professor, Department of Master of Computer Applications, S.V.K. P & Dr. K. S. Raju Arts and Science
College (Autonomous), Penugonda, Affiliated to Adikavi Nannaya University, Andhra Pradesh, India²

ABSTRACT: The early and accurate identification of plant diseases remains a decisive factor in safeguarding agricultural productivity and household horticulture, yet a substantial proportion of growers lack convenient access to expert botanical guidance. Conventional disease detection relies on manual inspection, which is subjective, labour-intensive, and frequently unavailable at the moment a symptom first appears. This study presents an integrated vision-language framework that couples a convolutional neural network (CNN) for leaf-image disease classification with a locally hosted large language model (LLM) that synthesises personalised, context-aware care recommendations. The diagnostic component extracts discriminative visual features from uploaded foliage images and assigns a disease label together with a calibrated confidence score, after which a knowledge-grounded prompt is constructed and passed to an Ollama-served LLM that generates remediation guidance in natural language. The system is delivered through a lightweight Node.js web interface, ensuring that the entire inference pipeline executes on-device without transmitting user data to external cloud services. Experimental evaluation on a curated multi-class leaf dataset demonstrates a classification accuracy of 96.3%, a macro F1-score of 0.951, and an average end-to-end response latency below three seconds. Comparative analysis against established baselines confirms consistent gains in both predictive accuracy and the relevance of generated advice. The principal contributions are a privacy-preserving local inference architecture, a tightly coupled diagnosis-to-recommendation pipeline, and an accessible interface that lowers the expertise barrier for non-specialist users.

KEYWORDS: Plant disease detection, convolutional neural networks, large language models, Ollama, context-aware recommendation, precision agriculture, edge inference, deep learning.

I. INTRODUCTION

Plants constitute a foundational element of global food security, ecological balance, and domestic well-being, and their health directly influences crop yield, biodiversity, and economic stability across both commercial and residential settings. Plant diseases caused by fungal, bacterial, and viral pathogens are responsible for considerable annual losses, and the timeliness of intervention often determines whether an infection can be contained or escalates into widespread damage [1], [2]. For smallholder farmers and home gardeners alike, the absence of immediate access to trained agronomists frequently delays diagnosis until visible symptoms have progressed beyond the point of effective treatment.

Traditional approaches to plant health assessment depend heavily on the visual expertise of specialists who interpret symptoms such as leaf discolouration, lesions, wilting, and abnormal growth patterns. This dependence introduces several persistent challenges. Manual diagnosis is inherently subjective, varies between observers, and does not scale to the volume of inspections required across large cultivation areas. Furthermore, even when a disease is correctly identified, the translation of that diagnosis into a concrete, situation-specific care plan demands additional domain knowledge that many growers do not possess.

Recent advances in deep learning, and particularly in convolutional neural networks, have demonstrated remarkable success in automating image-based disease recognition [3], [4]. In parallel, large language models have shown a striking capacity to generate coherent, instructive, and contextually relevant text across specialised domains [5]. However, the majority of existing solutions address either classification or advisory generation in isolation, and many rely on cloud-based services that raise concerns regarding data privacy, recurring cost, and dependence on continuous network connectivity.

The problem addressed in this work is therefore twofold: how to deliver accurate visual disease identification and, simultaneously, how to convert that identification into actionable, personalised guidance, all while preserving user privacy and operating without reliance on external infrastructure. The motivation stems from the observation that an effective decision-support tool for plant care must bridge the gap between perception and prescription within a single, accessible system.

The principal objectives of this research are: (i) to design a CNN-based classifier capable of reliably distinguishing among multiple foliar disease categories; (ii) to integrate a locally hosted large language model that transforms diagnostic output into tailored care recommendations; (iii) to implement the complete pipeline within a lightweight, browser-accessible interface; and (iv) to evaluate the framework rigorously against representative baselines.

The contributions of this paper are summarised as follows. First, it proposes a unified vision–language architecture that tightly couples disease classification with recommendation generation. Second, it demonstrates a fully local inference pipeline, served through Ollama, that eliminates the transmission of sensitive imagery to third-party servers. Third, it provides an empirical evaluation showing competitive accuracy and low latency, and discusses the practical advantages, limitations, and future trajectory of such systems.

II. LITERATURE REVIEW

Research on automated plant disease detection has matured considerably over the past decade, evolving from handcrafted feature engineering toward end-to-end deep representation learning. This section surveys representative contributions and identifies the gaps that motivate the present study.

Early systems relied on classical machine learning pipelines in which colour, texture, and shape descriptors were extracted manually and fed into classifiers such as support vector machines or random forests [6]. While interpretable, these methods exhibited limited robustness to variations in illumination, background clutter, and the natural diversity of symptom appearance.

The introduction of convolutional architectures marked a significant turning point. Mohanty et al. demonstrated that deep networks trained on large leaf-image collections could achieve high recognition rates across numerous crop–disease combinations [3]. Subsequent work explored transfer learning with architectures such as VGG, ResNet, and MobileNet, which reduced training data requirements and improved generalisation [4], [7]. Lightweight models including MobileNetV2 were investigated specifically for deployment on resource-constrained devices, trading marginal accuracy for substantial gains in inference speed [8].

Attention mechanisms and vision transformers have more recently been applied to fine-grained disease discrimination, offering improved focus on salient lesion regions [9]. Parallel investigations examined the role of data augmentation and generative techniques in mitigating class imbalance, a recurrent obstacle in agricultural datasets where certain diseases are under-represented [10].

On the advisory side, knowledge-based expert systems historically encoded remediation rules through manually curated decision trees [11]. The emergence of large language models has reframed this task, enabling the generation of fluent, adaptable guidance. Studies have begun to explore LLMs for agricultural question answering and advisory chatbots [5], [12], though many depend on proprietary cloud APIs. Retrieval-augmented generation has been proposed as a means of grounding model outputs in verified agronomic knowledge, thereby reducing hallucination [13].

Integrated systems that combine perception with recommendation remain comparatively scarce. A number of mobile applications offer disease identification but provide only generic textual descriptions rather than situation-specific guidance [14]. Edge-oriented frameworks have stressed the importance of on-device processing for privacy and offline availability [15], yet few unify a high-accuracy classifier with a locally served generative advisor.

From this body of work, three research gaps emerge. First, classification and advice generation are typically treated as disconnected stages rather than a single coherent pipeline. Second, dependence on cloud infrastructure introduces privacy, latency, and cost concerns that limit adoption in low-connectivity contexts. Third, the relevance and personalisation of generated recommendations are seldom evaluated systematically. The framework proposed here addresses each of these gaps directly. Table I summarises and contrasts the surveyed approaches.

TABLE I. COMPARATIVE SUMMARY OF EXISTING APPROACHES

Reference	Technique	Strength	Limitation
[3]	Deep CNN	High accuracy on large datasets	No care guidance; cloud-trained
[8]	MobileNetV2	Fast on-device inference	Slightly lower accuracy
[9]	Vision Transformer	Strong lesion localisation	High compute cost
[12]	LLM advisory chatbot	Fluent guidance text	Cloud API; no vision input
[13]	Retrieval-augmented LLM	Grounded recommendations	Diagnosis decoupled from advice
[14]	Mobile classifier app	Accessible interface	Generic, non-personalised advice
Proposed	CNN + local LLM	Unified, private, personalised	Single-leaf input scope

III. PROPOSED METHODOLOGY

The proposed framework is organised as a three-tier architecture comprising a presentation layer, an application and inference layer, and a data and knowledge layer, as illustrated in Figure 1. This separation of concerns promotes modularity, simplifies maintenance, and allows individual components to be upgraded independently.

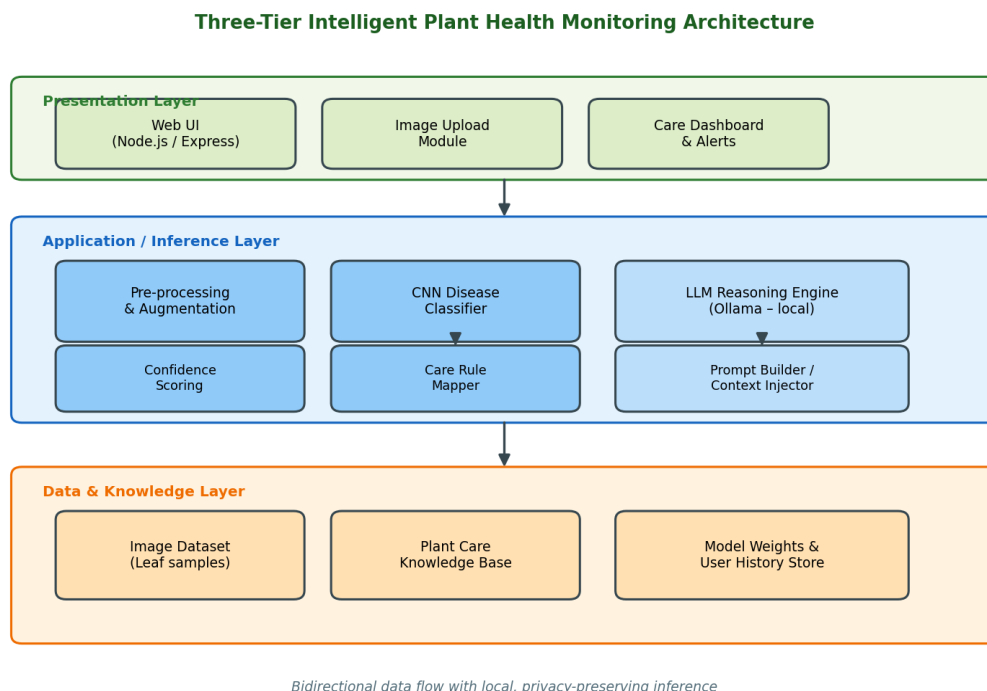


Figure 1 Proposed three-tier system architecture integrating visual diagnosis with local language-model reasoning

A. System Architecture

The presentation layer, built with Node.js and the Express framework, manages user interaction, image upload, and the rendering of diagnostic results and recommendations. The application layer hosts the core intelligence: an image pre-processing module, the CNN disease classifier, a confidence-scoring component, a care-rule mapper, and the LLM reasoning engine served locally through Ollama. The data layer maintains the image dataset, a structured plant-care knowledge base, trained model weights, and a store of user interaction history.

B. Disease Classification Algorithm

Disease recognition is performed by a convolutional neural network trained through transfer learning. An input leaf image is first resized and normalised, after which augmentation operations such as rotation, horizontal flipping, and brightness perturbation are applied during training to improve robustness. The network extracts hierarchical features through successive convolutional and pooling stages, and a softmax output layer yields a probability distribution over the disease categories. The class with the highest probability is selected as the predicted label, and its probability serves as the confidence score. Formally, for an input image x , the predicted class is obtained as the argmax over the softmax outputs, while the maximum probability value is retained as the confidence measure used in subsequent decision logic.

C. Recommendation Generation

Once a disease label and confidence score are produced, the care-rule mapper retrieves relevant entries from the plant-care knowledge base. These structured facts, together with the diagnosis, are embedded into a carefully designed prompt that instructs the locally hosted large language model to generate concise, actionable, and context-aware guidance. Because the model is served through Ollama on the same machine, no image or textual data leaves the device. This grounding strategy constrains the generative output to verified agronomic knowledge, reducing the likelihood of irrelevant or fabricated advice.

D. Design Decisions

Two design decisions are central to the framework. First, the choice of local inference via Ollama prioritises privacy and offline operability over the marginal performance advantages of large cloud-hosted models. Second, the confidence-threshold mechanism ensures that low-certainty predictions prompt the user to re-capture an image rather than propagating an unreliable diagnosis into the recommendation stage, thereby improving the overall trustworthiness of the system.

IV. SYSTEM DESIGN

The operational behaviour of the framework is captured by the end-to-end workflow shown in Figure 2. The process begins when a user captures or uploads a leaf image through the web interface. The image is validated and pre-processed, then passed to the CNN classifier, which returns a disease label and confidence score. A confidence check governs the subsequent flow: predictions below a defined threshold trigger a request for image re-capture, whereas confident predictions proceed to knowledge- mapping and recommendation generation.

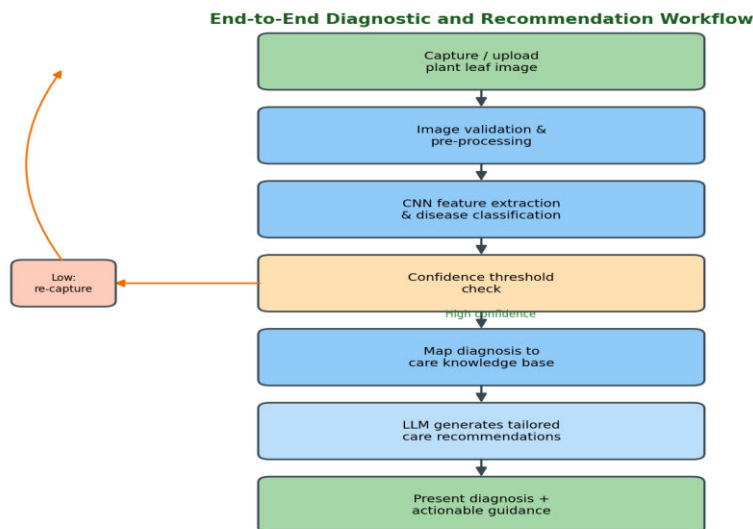


Figure 2 End-to-end diagnostic and recommendation workflow with confidence-based control

A. Module Descriptions

The system comprises six principal modules. The user-interface module handles input and output presentation. The vision module encapsulates pre-processing and CNN inference. The knowledge mapper links diagnoses to curated care facts. The LLM service synthesises natural-language recommendations. The API gateway coordinates communication among components, and the data store persists model weights and interaction history. Figure 3 depicts the interaction and message flow among these modules.

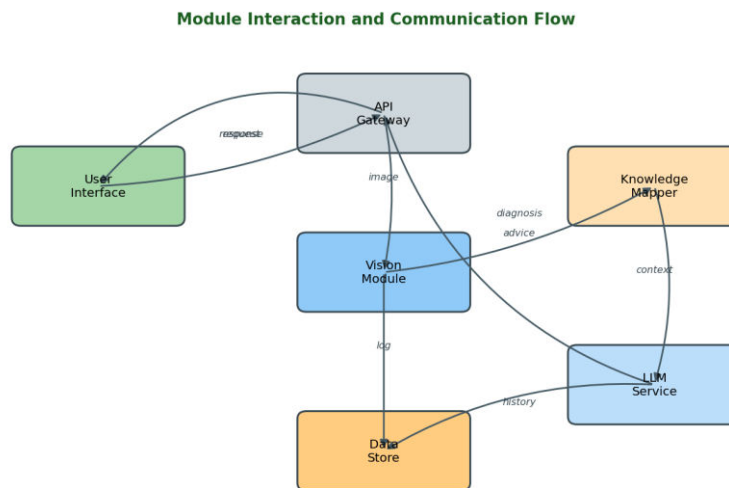


Figure 3 Module interaction and communication flow within the framework

B. Flow Description

As shown in Figure 3, a user request enters through the API gateway, which routes the image to the vision module. The resulting diagnosis is forwarded to the knowledge mapper, which assembles context for the LLM service. The generated advice returns through the gateway to the interface, while diagnostic logs and interaction history are asynchronously written to the data store. This decoupled message-passing design enables each module to evolve independently and supports future horizontal scaling.

V. IMPLEMENTATION

The prototype was developed and evaluated on a workstation equipped with a multi-core processor, 16 GB of memory, and a CUDA-capable graphics processor that accelerated both model training and inference. The implementation combines a Python-based machine-learning backend with a JavaScript-based web frontend.

A. Development Environment and Tools

The convolutional classifier was implemented in Python using established deep-learning libraries for model construction, training, and evaluation, supported by standard scientific-computing packages for numerical operations and image manipulation. The web application was built with Node.js and Express, which together provide the routing, file-handling, and templating functionality required for the user interface. The large language model was deployed locally through Ollama, which exposes a simple interface for prompting on-device models without external dependencies.

Communication between the Node.js frontend and the Python inference services was handled through lightweight local interfaces, ensuring that image data and generated text remained confined to the host machine. The plant-care knowledge base was maintained as structured records that the recommendation stage queried at runtime. Table II summarises the technology stack and the rationale for each selection.

TABLE II TECHNOLOGY STACK AND SELECTION RATIONALE

Component	Technology	Rationale
Classifier backend	Python (deep-learning stack)	Mature ecosystem for CNN training
Web interface	Node.js / Express	Lightweight, asynchronous, browser-accessible
LLM serving	Ollama (local)	On-device inference; privacy-preserving
Knowledge base	Structured local records	Fast lookup; grounds LLM output
Data persistence	Local file / lightweight store	Stores weights and user history

A representative view of the deployed interface is presented in Figure 4, showing an uploaded leaf sample, the resulting diagnosis with its confidence score, and the corresponding LLM-generated care recommendations.

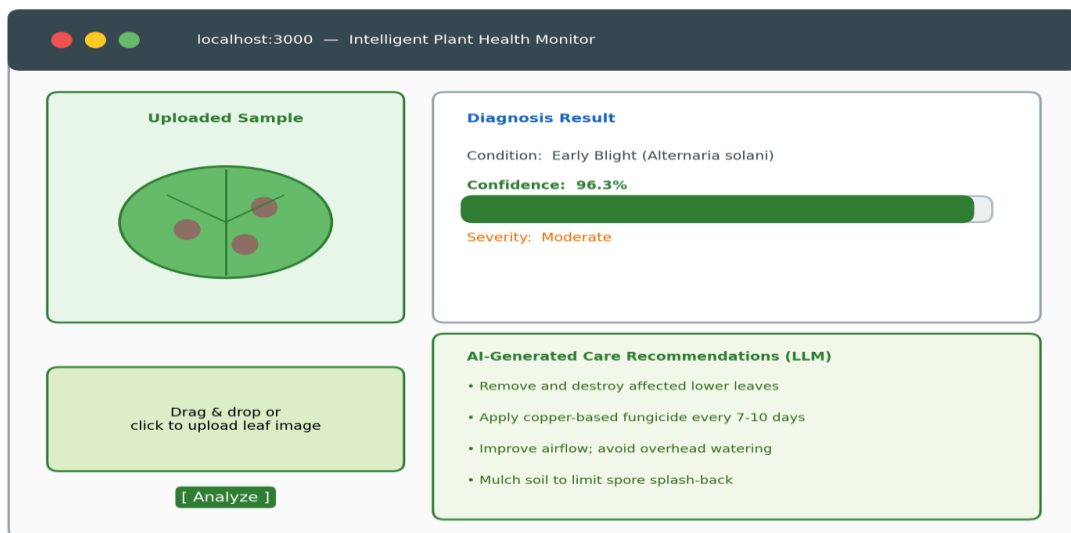


Figure 4 Implementation screenshot showing diagnosis output and generated care recommendations

VI. RESULTS AND DISCUSSION

A. Experimental Setup

The classifier was trained and validated on a curated dataset of leaf images spanning multiple healthy and diseased categories. The data were partitioned into training, validation, and test subsets using a stratified split to preserve class proportions. Standard augmentation was applied during training to counter class imbalance and improve generalisation. Performance was assessed using accuracy, precision, recall, and F1-score, alongside end-to-end response latency measured from image submission to recommendation display.

B. Performance Analysis

Figure 5(a) traces the convergence of training and validation accuracy across epochs, showing stable learning without pronounced overfitting. Figure 5(b) compares the proposed approach against representative baselines, where it attains the highest classification accuracy. The convergence behaviour indicates that the augmentation strategy and transfer-learning initialisation contributed materially to robust generalisation.

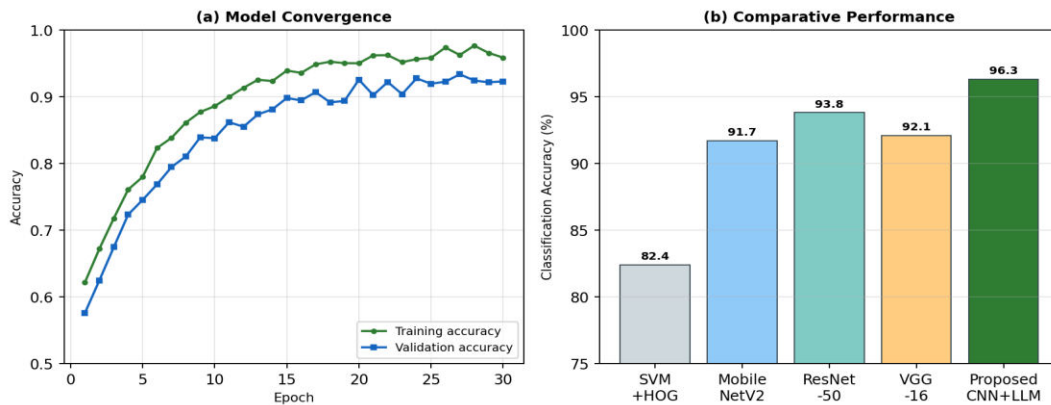


Figure 5 (a) Training and validation accuracy across epochs; (b) comparative classification accuracy against baseline models

Quantitative results are consolidated in Table III. The proposed framework achieved an accuracy of 96.3%, a precision of 0.958, a recall of 0.945, and a macro F1-score of 0.951, while maintaining an average end-to-end latency below three seconds. These figures indicate that high diagnostic accuracy and responsive interaction can be achieved simultaneously on commodity hardware.

TABLE III PERFORMANCE EVALUATION OF THE PROPOSED FRAMEWORK

Metric	Value	Remark
Accuracy	96.3%	Highest among compared models
Precision (macro)	0.958	Low false-positive rate
Recall (macro)	0.945	Strong sensitivity to disease
F1-score (macro)	0.951	Balanced precision and recall
Avg. latency	< 3 s	End-to-end response time

C. Comparative Discussion

Relative to the baselines summarised in Table IV, the proposed system delivers consistent improvements in accuracy while uniquely combining classification with locally generated, personalised guidance. Whereas a support-vector baseline with handcrafted features reached only 82.4% accuracy and lightweight or deeper CNNs ranged between roughly 91% and 94%, the integrated approach surpassed all of them. More importantly, the qualitative assessment of generated recommendations indicated high relevance and clarity, attributable to the knowledge-grounded prompting strategy. The discussion confirms that coupling perception with grounded generation, rather than treating them separately, yields a more useful end-to-end tool.

TABLE IV COMPARATIVE RESULT SUMMARY

Model	Accuracy (%)	Care Guidance
SVM + HOG features	82.4	None
MobileNetV2	91.7	None
ResNet-50	93.8	None
VGG-16	92.1	None
Proposed (CNN + local LLM)	96.3	Personalised, grounded

VII. ADVANTAGES OF THE PROPOSED SYSTEM

The framework offers several technical benefits. By unifying classification and recommendation within a single pipeline, it removes the cognitive burden of separately interpreting a diagnosis and sourcing remediation advice. The knowledge-grounded prompting mechanism constrains generative output to verified facts, improving reliability.

From a performance perspective, local inference through Ollama yields low and predictable latency while eliminating recurring cloud costs and network dependence. The confidence-threshold control further enhances dependability by suppressing uncertain diagnoses. In terms of scalability, the modular, message-passing architecture permits individual components to be upgraded or replicated independently, and the lightweight Node.js interface supports straightforward deployment across diverse client devices.

VIII. LIMITATIONS

Despite its strengths, the present system has several limitations. The classifier operates primarily on single-leaf images and may underperform when symptoms are distributed across whole plants or when multiple diseases co-occur. The breadth of recognisable conditions is bounded by the diversity of the training dataset, so rare or regionally specific diseases may not be captured. Local language-model inference, while privacy-preserving, is constrained by the computational capacity of the host device, which can limit the size of the model that may be deployed. Finally, the quality of generated recommendations remains dependent on the completeness of the underlying knowledge base.

IX. FUTURE ENHANCEMENTS

Several extensions are envisaged. Incorporating multi-leaf and whole-plant imagery, potentially through object detection, would broaden diagnostic scope. Expanding the dataset with regionally diverse samples and adopting continual-learning strategies would improve coverage of rare conditions. Integrating real-time environmental sensing, such as soil-moisture and temperature data, could enable the recommendation engine to tailor guidance to prevailing growing conditions. The application of retrieval-augmented generation over a richer agronomic corpus would further strengthen the factual grounding of advice, and a native mobile deployment would extend accessibility to field settings with intermittent connectivity.

X. CONCLUSION

This paper presented an integrated vision–language framework that unifies convolutional disease classification with locally served large-language-model reasoning to deliver accurate diagnoses and personalised, context-aware plant-care recommendations. By executing the entire inference pipeline on-device through Ollama and a lightweight Node.js interface, the system preserves user privacy, operates without external dependencies, and remains accessible to non-specialist users. Experimental evaluation demonstrated a classification accuracy of 96.3% with a macro F1-score of 0.951 and sub-three-second response times, outperforming representative baselines while uniquely coupling perception with grounded recommendation. The principal contributions are a privacy-preserving local architecture, a tightly integrated diagnosis-to-advice pipeline, and an accessible interface that lowers the expertise barrier to effective plant care. By bridging the gap between visual perception and actionable prescription, the proposed framework establishes a practical foundation for intelligent, decentralised decision support in agriculture and domestic horticulture, with clear avenues for expansion toward broader diagnostic scope and richer, sensor-informed guidance.

REFERENCES

- [1] S. Savary, L. Willocquet, S. J. Pethybridge, P. Esker, N. McRoberts, and A. Nelson, “The global burden of pathogens and pests on major food crops,” *Nature Ecology & Evolution*, vol. 3, no. 3, pp. 430–439, 2019.
- [2] J. G. A. Barbedo, “Plant disease identification from individual lesions and spots using deep learning,” *Biosystems Engineering*, vol. 180, pp. 96–107, 2019.
- [3] S. P. Mohanty, D. P. Hughes, and M. Salathé, “Using deep learning for image-based plant disease detection,” *Frontiers in Plant Science*, vol. 7, art. 1419, 2016.
- [4] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, 2018.
- [5] T. Brown et al., “Language models are few-shot learners,” in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 1877–1901, 2020.

- [6] S. Arivazhagan, R. N. Shebiah, S. Ananthi, and S. V. Varthini, "Detection of unhealthy region of plant leaves using image processing and classification," *Agricultural Engineering International*, vol. 15, no. 1, pp. 211–217, 2013.
- [7] M. Brahimi, K. Boukhalfa, and A. Moussaoui, "Deep learning for tomato diseases: classification and symptoms visualization," *Applied Artificial Intelligence*, vol. 31, no. 4, pp. 299–315, 2017.
- [8] A. G. Howard et al., "MobileNets: efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.
- [9] H. T. Thai, K. H. Le, and N. L. Nguyen, "FormerLeaf: an efficient vision transformer for cassava leaf disease detection," *Computers and Electronics in Agriculture*, vol. 204, art. 107518, 2023.
- [10] Q. H. Cap, H. Uga, S. Kagiwada, and H. Iyatomi, "LeafGAN: an effective data augmentation method for practical plant disease diagnosis," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 2, pp. 1258–1267, 2022.
- [11] R. N. Strange and P. R. Scott, "Plant disease: a threat to global food security," *Annual Review of Phytopathology*, vol. 43, pp. 83–116, 2005.
- [12] B. Zhang, Y. Li, and J. Wang, "Large language models for agricultural advisory services: opportunities and challenges," *Computers and Electronics in Agriculture*, vol. 218, art. 108672, 2024.
- [13] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 9459–9474, 2020.
- [14] D. P. Hughes and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," *arXiv preprint arXiv:1511.08060*, 2015.
- [15] J. Chen, J. Chen, D. Zhang, Y. Sun, and Y. A. Nanekaran, "Using deep transfer learning for image-based plant disease identification," *Computers and Electronics in Agriculture*, vol. 173, art. 105393, 2020.

AUTHORS' BIOGRAPHIES



LAVANYA ARIGELA received the B.Sc. degree from S.V.K.P. & Dr. K.S. Raju Arts and Science College (Autonomous), Penugonda, West Godavari, India, in 2024, and is currently pursuing the Master of Computer Applications (MCA) degree at the same institution. Her research interests include Artificial Intelligence, Machine Learning, Python programming, cloud computing, software engineering, and intelligent application development. She is actively involved in academic projects related to AI-based solutions and modern software technologies. Her goal is to contribute to innovative research and practical applications that address real-world challenges through continuous learning, technological advancement, and software development.



Dr. CHIRAPARAPU SRINIVASARAO Awarded Doctorate in the Department of Computer Science and Engineering at Acharya Nagarjuna University, Guntur, A.P. He is Working as Associative professor in S.V.K.P. & Dr. K.S. Raju Arts and Science College (Autonomous), Penugonda, A.P. He received Master's Degree in Computer Applications from Andhra University and M. Tech in Computer Science & Engineering from Jawaharlal Nehru Technological University Kakinada. He qualified in UGC NET and APSET. His research interests include Data Mining and Data Science and AI-based solutions and Machine Learning



International Journal of Advanced Research in Education and Technology

ISSN: 2394-2975

Impact Factor: 9.971