International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

AI-Driven Crop Quality Assessment using Deep Learning and Image Analysis with Integrated Smart Farming Support System

Dr. Amol D. Potgantwar
Principal, Sandip Institute of Technology and Research Center [SITRC]
Nashik, India

Prof. Savita B. Mogare
Project Guide, Department of Information Technology
Sandip Institute of Technology and Research Center [SITRC]
Nashik, India

Samruddhi B. Sonawane
Department of Information Technology
Sandip Institute of Technology and Research Center [SITRC]
Nashik, India

Neha J. Bora
Department of Information Technology
Sandip Institute of Technology and Research Center [SITRC]
Nashik, India

Janhavi R. Patil
Department of Information Technology
Sandip Institute of Technology and Research Center [SITRC]
Nashik, India

Prerna P. Suryawanshi
Department of Information Technology
Sandip Institute of Technology and Research Center [SITRC]
Nashik, India

***Abstract*—While deep learning has significantly advanced automated crop disease detection, existing models frequently suffer from severe performance degradation when deployed in unstructured field environments. Furthermore, current computer vision applications in agriculture typically function in isolation, evaluating visual symptoms without incorporating the critical microenvironmental context necessary for practical decision-making. This study proposes an integrated, multimodal smart farming support system specifically engineered for tomato crop management, a high-value but disease-susceptible crop. The architecture utilizes a dual-model deep learning pipeline: a YOLOv8-based object detection model for real-time fruit localization and maturity grading, coupled with a lightweight MobileNetV2 Convolutional Neural Network (CNN) incorporating depthwise separable convolutions for leaf and fruit disease classification. To bridge the gap between visual diagnosis and actionable agronomy, the system fuses the visual AI outputs with manually inputted soil parameters (pH, NPK) and automatically fetched geolocation-based meteorological data. A contextual Python-based rule engine processes this fused data to generate highly specific organic bio-fertilizer recommendations and automated seed-count requirements based on farm area. Addressing rural technological barriers, the web-based prototype, supported by a scalable Flask and MySQL backend, incorporates Web Speech APIs for voice-assisted outputs specifically in Marathi, ensuring accessibility for local illiterate farmers. The system's predictive robustness is evaluated by comparing classification accuracy on controlled laboratory datasets against unstructured field-captured images, providing a scalable, edge-compatible framework for low-resource agricultural deployment.**

***Index Terms*—Deep Learning, Computer Vision, Smart Agriculture, Multimodal Fusion, YOLOv8, MobileNetV2, Edge AI, Precision Agriculture.**

I. Introduction

AGRICULTURE forms the economic backbone of numerous developing nations, particularly across South Asia, where high-value cash crops like tomatoes play a critical role in global food security and the livelihoods of smallholder farmers [1]. However, tomatoes are highly susceptible to pathogenic infections—such as *Alternaria solani* (Early Blight) and *Phytophthora infestans* (Late Blight)—and require rigorous physiological monitoring throughout their growth cycle to ensure optimal yield and marketability. Maintaining strict quality control is essential, as physical defects, pest damage, and fungal diseases directly dictate the final commercial value in regional agricultural markets.

Conventionally, crop quality assessment and disease diagnosis rely on manual visual inspection by farmers or agricultural extension workers. This traditional approach is labor-intensive, highly subjective, and prone to significant error margins, often resulting in delayed interventions [2]. Furthermore, manual grading at local markets (mandis) frequently disadvantages smallholder farmers who lack access to standardized, objective quality assaying. This lack of transparency leads to severe price exploitation and substantial post-harvest economic losses.

To address these inefficiencies, computer vision and deep learning have emerged as powerful tools for automating agricultural monitoring. Convolutional Neural Networks (CNNs) and advanced object detection architectures have demonstrated exceptional capabilities in extracting hierarchical spatial features from plant images. These models enable the rapid classification of leaf diseases and the assessment of fruit maturity stages without requiring manual, hand-crafted feature engineering [3].

Despite these algorithmic advancements, significant limitations hinder the practical deployment of AI in real-world agricultural scenarios. A primary research gap is the stark discrepancy between laboratory and field performance. Deep learning

models trained on highly controlled, uniform datasets (such as PlantVillage) frequently fail to generalize when exposed to the chaotic illumination, varying camera angles, occlusion, and background complexities of actual farm environments [4]. Moreover, current agricultural AI systems operate in a diagnostic vacuum. They provide isolated visual classifications but critically fail to account for environmental vectors like soil nutrient deficiencies (Nitrogen, Phosphorus, Potassium) and impending humidity, which biologically dictate the spread velocity and required treatment of a disease.

Consequently, an accurate visual diagnosis is insufficient for comprehensive agricultural decision-making if it ignores micro-environmental context and resource planning. There is a pressing need for multimodal smart farming systems that synthesize visual AI data with meteorological forecasts, soil parameters, and spatial farm constraints. Additionally, to be practically viable for resource-constrained farmers, such systems must operate on lightweight architectures accessible via low-cost mobile devices and include accessibility features, such as regional voice-assisted interfaces, to bridge the severe digital literacy divide [5].

This study proposes a comprehensive, context-aware smart farming architecture tailored for tomato cultivation. The system integrates a You Only Look Once (YOLO) framework for maturity grading with a MobileNet-based CNN for lightweight disease classification. Moving beyond pure image analysis, the architecture incorporates a Flask-based backend and a Python logic engine that dynamically fuses the AI predictions with local weather APIs and soil data to generate specific, organic interventions. Furthermore, the system includes automated seed-requirement calculations and integrates with real-time market pricing APIs, all deployed via a scalable PaaS (Platform-as-a-Service) environment.

The primary contributions of this research are as follows:

1) The development of a dual-model, edge-compatible deep learning pipeline for simultaneous tomato disease classification and maturity grading.
2) The design of a multimodal decision engine that synthesizes computer vision outputs with meteorological and soil data to deliver context-aware agronomic interventions.
3) The formulation of a programmatic rule-engine algorithm that maps visual plant pathology to localized biofertilizer recommendations.
4) The implementation of an accessible, web-based prototype featuring automated resource calculations, real-time market intelligence, and Marathi voice-assisted outputs for illiterate users.
5) An experimental evaluation quantifying the performance degradation of the proposed models when transitioning from controlled laboratory datasets to unstructured field imagery.

II. Background and Related Work

A. Deep Learning in Agricultural Computer Vision

The application of deep learning to agricultural computer vision has fundamentally shifted crop disease diagnosis from manual feature extraction to automated spatial feature learning. Deep CNN architectures, such as ResNet-50 and InceptionV3, have established baseline supremacy in leaf disease classification [6]. However, models trained on highly controlled datasets, such as PlantVillage, often experience severe accuracy drops—sometimes falling below 50%—when tested on unstructured, field-captured images containing overlapping leaves and varied illumination [4]. Singh et al. [4] highlighted this through the PlantDoc dataset, proving that deploying lightweight, field-robust models remains a critical, unsolved challenge in precision agriculture.

B. Tomato Maturity and Yield Assessment

Object detection architectures, particularly the You Only Look Once (YOLO) series, have proven highly effective for real-time fruit localization and counting in complex canopy environments due to their single-stage detection mechanics [7]. Recent studies have adapted YOLO variants to grade tomato ripeness (e.g., green, breaker, turning, red) to estimate harvest readiness. However, these studies typically treat quality assessment as an isolated computational task, failing to integrate these visual metrics into a broader economic or agronomic recommendation framework.

C. Context-Aware Smart Farming and Multimodal Fusion

Precision agriculture inherently relies on synthesizing multiple data streams to inform agronomic interventions. Despite advancements in IoT, most AI-driven disease detection applications evaluate pixels without context. The required agronomic intervention for a fungal infection depends heavily on current soil nitrogen levels and impending rainfall [8]. Few studies have successfully fused edge-based computer vision outputs with real-time meteorological APIs and localized soil data to dynamically adjust the AI's final recommendation.

D. Accessibility and Market Linkage in Agriculture

For smallholder farmers, technological adoption is often hindered by literacy barriers and a lack of market transparency. While commercial applications offer crowdsourced disease identification, they primarily rely on text-based interfaces and lack integration with real-time agricultural market pricing. The absence of regional voice-assisted navigation and localized post-harvest market intelligence severely limits the socio-economic impact of these AI tools.

III. Proposed System Architecture

A. System Overview and Architectural Framework

The proposed system is engineered as a highly scalable, edge-compatible web application. To ensure high availability and smooth version control, the application is designed to be hosted on modern Platform-as-a-Service (PaaS) environments. The architectural framework is partitioned into four primary layers: the Client/Accessibility Layer, the AI Vision Layer, the Contextual Logic Engine, and the Data Persistence Layer. The core backend infrastructure is built on Python and Flask, providing a robust RESTful API routing environment capable of handling concurrent multimodal requests.



Fig. 1. System Execution Flow demonstrating the step-by-step chronological pipeline: from multimodal client ingestion (RGB imagery, NPK, Geolocation) through Flask API routing, parallel AI and rule-engine processing, to final MySQL data persistence and Marathi voice synthesis.

TABLE I
SUMMARY OF RELATED WORK AND IDENTIFIED RESEARCH GAPS

Authors (Year)	Core Focus / Task	Model / Dataset	Main Result / Strength	Key Limitation
Singh et al. (2020) [4]	Plant disease detection in real-world environments	VGG, Inception / PlantDoc	Highlighted the failure of lab-trained models in the field.	Low baseline accuracy; lacked agronomic decision support.
Chettri et al. (2023) [8]	Deep learning for precision agriculture	CNNs, Vision Transformers	Emphasized edge AI optimization for rural deployment.	Multimodal fusion (vision + IoT) is rarely implemented.
Liu et al. (2023) [7]	Tomato detection and ripeness grading	YOLOv8 / Custom Field	High real-time mAP for fruit localization on the vine.	Did not assess simultaneous leaf disease or market value.
Commercial Systems [5]	Crowd-sourced disease identification	Proprietary CNNs	Massive disease database; high adoption rate.	Lacks post-harvest market grading and regional voice accessibility.

B. Database Schema and Persistence Layer

A relational MySQL database forms the backbone of the system's persistence layer, ensuring data integrity and historical tracking for farmers. The schema consists of two primary tables:

- **Users Table:** Manages secure authentication, farmer profiles, and default regional preferences.
- **Assessment Logs Table:** Stores foreign keys linking to the user, serialized JSON objects of the manually entered NPK values, weather snapshots, the generated market grade, seed count metrics, and the final AI diagnosis. This enables longitudinal tracking of crop health across harvesting seasons.

C. Data Ingestion and Accessibility Layer

To bridge the digital literacy divide, the client interface prioritizes accessibility, minimalism, and automation.

- **Multimodal Input:** The user uploads an RGB smartphone image of a tomato leaf or batch. To bypass the cost and hardware complexity of physical IoT sensors, the farmer manually inputs localized soil parameters (pH, N, P, K) and the total farm area under cultivation.
- **Automated Geolocation:** Utilizing the HTML5 Geolocation API, the system automatically extracts the user's latitude and longitude upon consent, removing friction from the UX.
- **Marathi Voice Integration:** The interface integrates the Web Speech API tailored for regional Maharashtra accessibility. Navigation can be triggered via speech-to-text, and all final diagnostic reports, fertilizer recommendations, and seed calculations are synthesized via text-to-speech directly in Marathi.

IV. METHODOLOGY: DUAL-MODEL DEEP LEARNING PIPELINE

The AI Vision Layer avoids the bottleneck of monolithic models by employing a dual-model pipeline, decoupling spatial localization from fine-grained pathological classification.

1) Fruit Detection and YOLOv8 Loss Function Formulation: A YOLOv8 architecture scans the input RGB image. Utilizing a CSPDarknet backbone, YOLO provides superior real-time inference speed. To ensure precise bounding box regression for tomato maturity grading, the model minimizes the Complete Intersection over Union (CIoU) loss. The CIoU loss accounts for the overlapping area, central point distance, and aspect ratio, defined as:

$$L_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v \quad (1)$$

where $\rho(\cdot)$ represents the Euclidean distance between the central points of the predicted box b and the ground truth box b^{gt}, c is the diagonal length of the smallest enclosing box covering both, and v measures the consistency of the aspect ratio:

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2 \quad (2)$$

This formulation ensures that bounding boxes accurately encapsulate fruits even in heavily occluded canopy environments.

2) Disease Classification and MobileNetV2 Efficiency: To ensure the disease classification model remains edge-compatible and deployable via a lightweight Flask backend, MobileNetV2 is implemented. The architecture replaces standard convolutions with depthwise separable convolutions. For a standard convolutional layer taking an $M \times M$ input with D_F channels and applying a $D_K \times D_K$ kernel to produce N output channels, the computational cost reduction ratio is:

$$\text{Reduction Ratio} = \frac{1}{N} + \frac{1}{D_K^2} \quad (3)$$

By utilizing 3×3 depthwise convolutions, the model requires approximately 8 to 9 times less computation than standard CNNs, allowing for rapid inference without requiring dedicated cloud GPU instances.

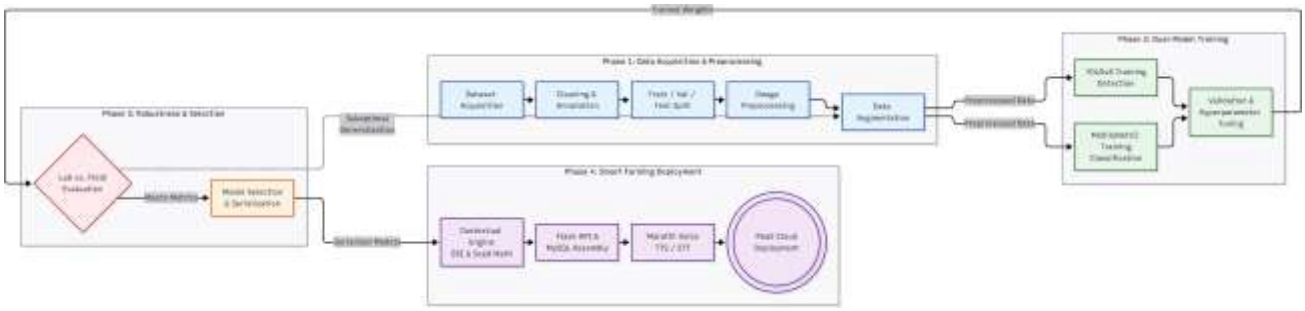


Fig. 2. System Execution Flow demonstrating the step-by-step sequential logic: starting from raw data ingestion and dual-model inference, progressing to the Python rule-based evaluation, calculating required seed counts, fetching dynamic market prices, and culminating in an accessible Marathi voice output for the farmer.

V. Smart Farming Logic and Resource Integration

A. Contextual Fusion and Agronomic Rule Engine

The Python-based Contextual Logic Engine operates as a deterministic rule-based synthesizer. Using the automatically acquired geographic coordinates, the Flask backend queries the OpenWeatherMap API to retrieve 48-hour relative humidity, temperature, and precipitation forecasts. The rule engine evaluates the AI-predicted disease against this localized context.

Algorithm 1 Agronomic Rule Engine Logic

Require: $Image$, $Location$, $Soil_{NPK}$
$Disease, Conf \leftarrow CNN(Image)$
$Weather \leftarrow API(Location)$
$Humid \leftarrow Weather.humidity$
$Temp \leftarrow Weather.temp$
if $Disease$ == "Early Blight" **and** $Conf > 0.8$ **then**
  if $Humid > 80\%$ **and** $Soil_{NPK}.N < threshold$ **then**
    $Rec \leftarrow$ "High Risk: Apply Copper Fungicide & N-Compost"
  else if $Humid \leq 80\%$ **then**
    $Rec \leftarrow$ "Moderate Risk: Apply Neem Oil spray"
  end if
else if $Disease$ == "Healthy" **then**
  $Rec \leftarrow$ "Maintain standard irrigation"
end if
return Rec

B. Automated Agronomic Resource Calculator

To assist farmers in pre-season planning and prevent resource wastage, the system dynamically calculates required inputs. Assuming a standard tomato row spacing (S_r) of 0.6 meters and plant spacing (S_p) of 0.45 meters, the total viable plant count (C) for a user-inputted cultivable area (A, in square meters) is calculated as:

$$C = \frac{A}{S_r \times S_p} \times (1 + \epsilon) \quad (4)$$

where ϵ represents a buffer coefficient (set to 0.10 or 10%) to account for baseline seed germination failure and seedling mortality rates. This prevents under-ordering of seeds.

C. Market Intelligence Integration

A batch "Market Grade" (e.g., Grade A, Grade B) is algorithmically determined based on the YOLO-derived maturity ratio and the CNN-derived disease severity index. The system interfaces with an agricultural market API (mimicking the Indian e-NAM framework) to retrieve live, location-specific pricing per quintal for the assigned grade, providing immediate economic context to the visual diagnosis.

VI. Experimental Setup and Evaluation Metrics

A. Dataset Acquisition and Preprocessing

The CNN classification model utilizes the Tomato subset of the public PlantVillage dataset for baseline training (approx. 18,000 images). A secondary test set is curated from the PlantDoc dataset to explicitly evaluate unstructured field robustness. The YOLO model is trained on a combined dataset from Roboflow Universe featuring images of tomatoes on the vine.

To prevent overfitting, rigorous data augmentation is applied exclusively to the training sets. Augmentations include random rotations (±45°), horizontal/vertical flipping, mosaic augmentations for YOLO, and brightness/contrast adjustments to simulate varying canopy shadow conditions. Let x be the original image array, the normalized image array x' fed into the network is computed as:

$$x'_{i,j,c} = \frac{x_{i,j,c} - \mu_c}{\sigma_c} \quad (5)$$

where μ_c and σ_c are the channel-wise mean and standard deviations of the dataset.

B. Hyperparameter Configuration

Model training is conducted using PyTorch on cloud-based GPU accelerators. The MobileNetV2 model is fine-tuned using an Adam optimizer with an initial learning rate of 1×10^{-4}, categorical cross-entropy loss, and a batch size of 32 over 50 epochs, utilizing early stopping. YOLOv8 is trained for 100 epochs with a batch size of 16 and an image resolution of 640×640.

C. System Evaluation Metrics

The disease classification model is evaluated using standard machine learning metrics. To handle class imbalances appropriately, the F1-Score is emphasized:

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \quad (6)$$

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \quad (7)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (8)$$

The YOLO model is evaluated using Mean Average Precision (mAP) at an Intersection over Union (IoU) threshold of 0.5 (mAP@0.5). Additionally, practical deployment latency is measured to ensure the end-to-end response time is acceptable (≤ 3.0 seconds) for standard 4G mobile networks.

D. Explainable AI (XAI) Integration via Grad-CAM

To enhance the transparency and trustworthiness of the CNN for agricultural stakeholders, Gradient-weighted Class Activation Mapping (Grad-CAM) is integrated into the testing phase. Grad-CAM utilizes the gradients of the target disease class flowing into the final convolutional layer to produce a coarse localization map, highlighting the critical regions in the image. The localization map $L^c_{Grad-CAM}$ for class c is computed as:

$$L^c_{Grad-CAM} = ReLU\left(\sum_k \alpha_k^c A^k\right) \quad (9)$$

where A^k represents the feature map activations and α_k^c are the neuron importance weights. This visualization ensures the model is activating based on actual pathological lesions rather than background noise, which is a critical validation step for field-deployed AI.

VII. Expected Results and Discussion

A. Deep Learning Model Performance

The lightweight CNN is expected to achieve high baseline classification accuracy ($> 92\%$) and F1-scores on the controlled validation set. Confusion matrices are anticipated to show minor misclassifications between early-stage pathogenic symptoms. The YOLO architecture is anticipated to demonstrate high mAP (> 0.85) for detecting tomatoes against dense green canopies, reliably generating bounding boxes for maturity classes.

B. Comparative Analysis: Laboratory vs. Field Robustness

A core contribution is quantifying the lab-to-field performance gap. It is hypothesized that the CNN model will exhibit a statistically significant drop in accuracy (estimated 15-25% reduction) when evaluated against the unstructured Field test set due to real-world environmental noise. This validates the necessity of our contextual fusion approach, as AI alone cannot be solely relied upon in field conditions.

C. Ablation Study (Proposed)

An ablation study will be conducted to isolate the impact of data augmentation and architectural choices:

- **Experiment 1:** CNN without data augmentation vs. CNN with augmentation.
- **Experiment 2:** ResNet50 (Heavy) vs. MobileNetV2 (Lightweight) to compare inference latency against accuracy trade-offs.
- **Experiment 3:** Image-only prediction vs. Context-fused recommendation (qualitative assessment).

D. Evaluation of Contextual Logic and Resource Calculation

Scenario-based validation will demonstrate that the fusion of AI-detected disease, API-fetched weather data, and manually inputted soil parameters produces highly differentiated, scientifically sound agronomic recommendations compared to static visual diagnosis alone. The seed-count calculator will provide accurate scaling for farm-area input.

VIII. Discussion, Limitations, and Future Scope

A. Discussion on Socio-Economic Impact

By coupling YOLO and CNN outputs with weather forecasts and MySQL-backed soil data logs, the system outputs actionable interventions rather than mere pathological labels. Integrating the automated seed-count calculator and live market-pricing API directly links crop health to economic forecasting, addressing critical pain points for smallholder farmers. The inclusion of the Marathi Web Speech API ensures that these advancements bypass regional literacy barriers, allowing for genuine grassroots adoption in Maharashtra and surrounding rural demographics.

B. Limitations

1) **Dataset Bias:** Real-world agricultural environments present infinite variables (e.g., overlapping canopy shadows, multi-disease infections) that may cause performance degradation despite rigorous data augmentation.
2) **Manual Data Entry Dependency:** The system relies on manual user input for soil NPK and pH values, introducing a vulnerability to human estimation error compared to real-time automated sensors.
3) **Connectivity Constraints:** The architecture heavily depends on continuous internet connectivity to fetch meteorological APIs and execute speech models, limiting reliability in deep rural sectors.
4) **Language Limitations:** While the Marathi Web Speech API provides critical functionality, natural language generation can occasionally degrade when handling hyper-local agronomic slang.

C. Future Scope

Future research should focus on transitioning to low-cost, Arduino-based IoT sensors for automated soil data collection, removing manual input friction. Furthermore, models could be compressed using quantization (e.g., TFLite) for complete

offline edge inference on mobile devices. Finally, expanding the NLP models to support additional Indian agricultural languages (e.g., Hindi, Telugu, Tamil) will ensure broader regional accessibility and scalability.

IX. Conclusion

This study presented a scalable, edge-compatible, multi-modal smart farming system designed to address the diagnostic vacuum in tomato cultivation. By deploying a dual-model computer vision pipeline via a robust Flask API backend, the system decoupled maturity grading from disease classification. Integrating a Python-based rule engine allowed the system to synthesize visual predictions with meteorological data and localized soil parameters, generating precise, context-aware organic interventions. Furthermore, the inclusion of an automated seed calculator, market-price fetching, and Marathi voice-assisted outputs transformed an isolated diagnostic tool into a comprehensive economic and resource-planning platform. Future iterations incorporating offline edge inference will further solidify this architecture as a transformative solution for resource-constrained agriculture.

References

[1] FAO, "The State of Food and Agriculture 2023," Food and Agriculture Organization of the United Nations, 2023.
[2] A. Kamilaris and F. X. Prenafeta-Boldu´, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70-90, 2018.
[3] 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, 2016.
[4] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, "PlantDoc: A Dataset for Visual Plant Disease Detection," *Proceedings of the 7th ACM IKDD CoDS*, pp. 249-253, 2020.
[5] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311-318, 2018.
[6] J. Chen, J. Chen, D. Zhang, Y. Sun, and Y. A. Nanehkaran, "Using deep transfer learning for image-based plant disease identification," *Computers and Electronics in Agriculture*, vol. 173, p. 105393, 2020.
[7] G. Liu, J. C. Nouaze, P. L. Touko Mbouembe, and J. H. Kim, "YOLO-Tomato: A Robust Algorithm for Tomato Detection Based on YOLOv3," *Sensors*, vol. 20, no. 7, p. 2145, 2020.
[8] S. R. Chettri and S. B. Bera, "A comprehensive survey on deep learning in smart agriculture," *Artificial Intelligence in Agriculture*, vol. 9, pp. 1-25, 2023.