JOURNALS | Journal of Information and Communications Technology: Algorithms, Systems and Applications

Volume 2 Issue 4 (December 2026)

DOI: https://doi.org/10.64189/ict.26320

J. Inf. Commun. Technol. Algorithms, Syst. Appl., 2026, 2(4), 26320

Research Article

OPEN ACCESS

Integrated Agro-Diagnostic Framework with Deep Learning for Crop Disease Management

V. Lokeswara Reddy,[1,*] V. Srilatha,[2,*] V. Ganga Devi,[2] P. Shiva Durga,[2] R. Tharun[2] and T. Harsha Vardhan Reddy[2]

[1] *Department of Computer Science and Engineering, K.S.R.M. College of Engineering (Autonomous), Kadapa, Andhra Pradesh 516003, India*

[2] *Department of Agricultural Sciences, Acharya N. G. Ranga Agricultural University (ANGRAU), Bapatla, Andhra Pradesh 522101, India*

*Corresponding Author

V. Lokeswara Reddy

Vlreddy74@gmail.com

V. Srilatha

v.srilatha@angrau.ac.in

Received: 08 August 2026

Revised: 01 October 2026

Accepted: 07 October 2026

Published Online: 09 October 2026

Citation

V. L. Reddy, V. Srilatha, V. G. Devi, P. S. Durga, R. Tharun, T. H. V. Reddy, Integrated agro-diagnostic framework with deep learning for crop disease management, *Journal of Information and Communications Technology: Algorithms, Systems and Applications*, 2026, 2(4), 26320, doi: https://doi.org/10.64189/ict.26320.

Open Access

This article is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License, which permits the non-commercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as appropriate credit to the original author(s) and the source is given by providing a link to the Creative Commons License and changes need to be indicated if there are any. The images or other third-party material in this article are included in the article's Creative Commons License, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons License and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this License, visit: https://creativecommons.org/licenses/by-nc/4.0/

© The Author(s) 2026

Abstract

Plant diseases are a constant issue in food production, resulting in worldwide losses of 20–40% on an annual basis and more than $147 billion in damages. In India, more than 150 million farmers have an annual contribution of nearly 18% to the country's GDP, and timely diagnosis of diseases is not easily accessible in rural areas because of the very limited availability of trained plant pathologists, which leads to the loss of yields in the field ranging between 25% and 50%. Most of the existing CNN-based systems are only concerned with the detection of diseases, and farmers are not provided with any practical suggestions for treatment or input procurement. In response, we present a lightweight CNN network called AgriNet, trained on the PlantVillage dataset of 61,486 labeled leaf images that represent 38 different classes for 14 different crops, and present an Integrated Agro-Diagnostic and Remediation Framework (IADRF) to address this. Under an identical experimental setup (same dataset, data split and training protocol), AgriNet is compared with Xception, ResNet-50 and Inception V3 in terms of accuracy, precision, recall, F1-score, number of parameters and inference latency. The proposed model not only reaches high accuracy of 98.68% but also yields a macro-averaged F1-score of 98.47% while using far fewer trainable parameters than the conventional CNN architectures. The prototype IADRF combines disease predictions with a structured collection of information regarding disease prevention and management and presents it through a user-friendly web interface on a mobile device. The reported latency results prove the possibility of calculations on the server side and in low resource areas, but direct deployment of the smartphone and validation at the field level remains a topic for further research.

***Keywords*:** Convolutional neural network (CNN); Deep learning; PlantVillage dataset; Decision support system; Precision agriculture; Plant leaf disease detection.

1. Introduction

The Indian economy is primarily based on agriculture, employing over 150 million farmers and contributing about 18% to the national GDP. Nevertheless, plant diseases continue to be among the most enduring challenges to agricultural output worldwide, with yearly crop losses of 20–40% and estimated economic losses exceeding 147 billion dollars globally.[1] Staple crops such as wheat, rice, maize, and vegetables are attacked by fungal, bacterial, and viral pathogens that result in large-scale yield degradation and food insecurity. Early and precise diagnosis of plant diseases is required for effective crop management. Farmers in developing countries like India, especially rural farmers, still largely rely on visual checking to detect disease symptoms. This is an inefficient, time-consuming, and subjective process, particularly since numerous diseases present similar visual symptoms in their early stages.[2] This is further compounded by the unavailability of plant pathologists, which can cause a delay in diagnosis resulting in as much as 50% loss in the field. In recent years, deep learning, in particular Convolutional Neural Networks (CNNs), has achieved remarkable results in image classification[3,4] and is thus a highly attractive approach to automate plant disease classification. Current CNN-based systems can accurately identify diseased leaves,[5–7] but they are limited in that they merely label the disease but do not offer any helpful information to the farmer. Although a farmer may learn that an apple tree has black rot, the farmer also needs information on which drugs to apply, how to prevent the disease, and where to buy the medicine. To address these limitations, in this paper, we propose an Integrated Agro-Diagnostic and Remediation Framework (IADRF). The IADRF leverages a lightweight CNN (AgriNet) to detect diseases and adds a decision support system that offers prevention and treatment advice and links to e-commerce platforms. This approach connects disease identification with actionable information for farmers, thereby facilitating sustainable and timely disease management. The following Fig. 1 shows the original plant photographs (upper panels) alongside their segmented versions (lower panels).

1.1 Research gaps

- Although there have been significant advances in deep learning-based plant disease classification, four major gaps still exist. The first is that most studies are concerned with accuracy of classification on the benchmark data sets and do not assess the limitations in deployment and computational efficiency.[5,6]
- Secondly, performance is reported on controlled set of images which may not necessarily be the same on more challenging backgrounds, illumination change, occlusion and image noise in the field.[2,5]
- Thirdly, the categorization of diseases and decision support for agriculture have been treated separately.[7]
- Fourth, relatively little research has been directed towards the use of relatively light diagnostic and practical preventive and therapeutic guidelines in a single convenient computer-friendly user manual for farmers.

The gaps are the foundation for the development of IADRF, with AgriNet providing a compact visual disease classification and a structured decision-support layer, that translates the predicted disease to actionable agricultural information.

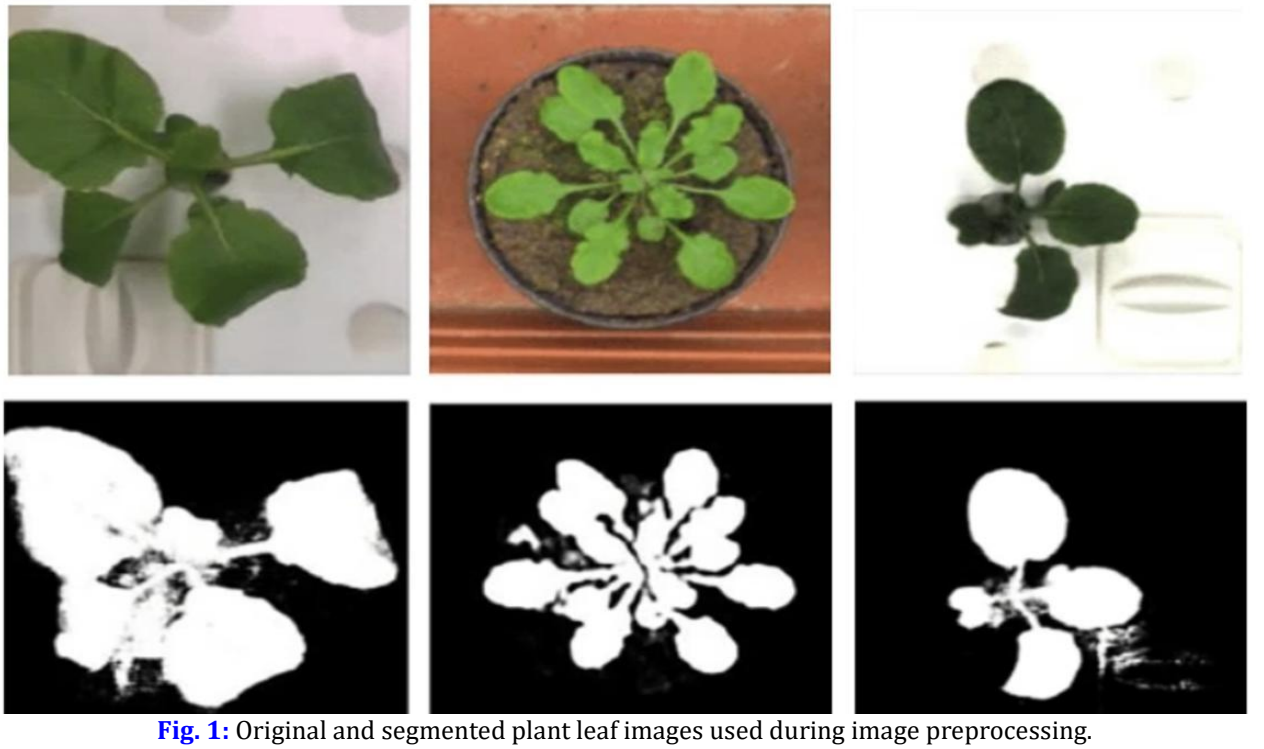

Fig. 1: Original and segmented plant leaf images used during image preprocessing.

1.2 Research contributions

- Inspired by the success of CNNs in image classification, a lightweight CNN architecture, AgriNet, is designed to classify multi-class plant disease images and is optimised using depthwise separable convolution, batch normalization, dropout and global average pooling.[8–11]
- The Integrated Agro-Diagnostic and Remediation Framework (IADRF) is designed to integrate the image-based disease identification system with structured agricultural decision support system.
- The proposed model is compared with three widely used models, namely Xception, ResNet-50 and Inception V3, in terms of accuracy, precision, recall, F1-score, computational parameters and inference latency.[12–14]
- A system-level analysis is provided for investigating the relationship between classification performance and efficiency of computations.
- The restrictions of the controlled benchmark data sets are clearly stated, and the purpose of the framework is to be used as a first-tier decision support system, not a replacement for field agronomic information.

2. Related work

In this section, key contributions towards plant disease detection using deep and machine learning have been reviewed, showing this technology's methodology, accuracy, and shortcomings. Krizhevsky *et al.* proposed the first deep convolutional neural network (CNN), called AlexNet, which exposes the capability of deep learning for massive image classification problems.[4] He *et al.* presented the Spatial Pyramid Pooling (SPP-Net) that enables CNNs to handle images of varying sizes.[15] Mohanty *et al.* applied deep CNNs to the PlantVillage dataset and achieved high accuracy in the controlled environment, but the accuracy dropped in the field.[5] Hu *et al.* used feature fusion with D–S evidence theory and ANN to classify plant diseases, with moderate accuracy but no scalability.[16] Esteva *et al.* successfully used CNNs for medical image classification at the level of experts, which indicated that deep learning is effective in detecting diseases.[17] Alehegn applied SVM in the classification of maize leaf disease, but the approach could not be applied to other crops.[18] Zhang *et al.* used domain knowledge with deep neural networks to improve classification.[19] A survey by Beyene *et al*. showed that data sets are small, environmental conditions are variable, and present systems are not executed in real time.[20] Zhang *et al*. used deep learning for grouping of agricultural machinery and showed the adaptability of CNN models to agriculture.[21] Bhuvaneswari *et al*. classified plant leaf diseases using an LNet-based deep learning model.[22] Eghbali and Hajihosseini investigated CNN-based large-scale image classification and noted the important role of deep structures in improving classification accuracy.[23] Ezat *et al.* developed a deep learning-based multiclass classification with performance improvement on various categories.[24] Rastogi *et al*. evolved a CNN-based method for early detection of plant diseases using leaf images.[25] This research addressed efficient models suitable for deployment in real agricultural settings, which showed better detection accuracy than conventional image processing techniques. Mahmood Ur Rehman *et al.* proposed a systematic approach to detect diseases in multiple crops through CNNs, with a focus on transfer learning approaches to tackle the problem of limited labeled field data.[26] They assessed the performance of the model in field conditions and suggested approaches to enhance its generalizability. Sunil and Jaidhar involved using a scalable and lightweight CNN model, EfficientNetV2, for identifying cardamom diseases, a lucrative crop in South Asian regions.[27] The research achieved high accuracy with minimal computational burden and could be deployed on mobile devices in the field. Despite the above advancements, most of the current methods focus only on the classification of diseases, and only a few provide decision support such as treatment recommendations, preventive steps, and input procurement. Further, few systems are designed for real-time functionality under resource-limited settings. The IADRF system proposed here overcomes these limitations by integrating very high-accuracy classification with decision support.

3. Proposed system

The Integrated Agro-Diagnostic and Remediation Framework (IADRF) aims to achieve two complementary goals: automated visual diagnosis of plant leaf diseases using smartphone images and transmission of actionable guidance in real time to assist with farming decisions based on the plant diseases identified. The following Fig. 2 represents the system architecture of the proposed IADRF framework.

3.1 System contributions

The most important aspects of the proposed framework include:

- AgriNet, a lightweight task-specific CNN architecture for efficient multi-class plant disease classification.
- A rule-based decision support module that maps each disease class detected to a curated database with prevention guidelines, organic and chemical treatment options, and fertilizer or pesticide recommendations.
- A Flask-based web application, which can be accessed from a mobile device, takes in leaf photos through a regular phone camera and provides

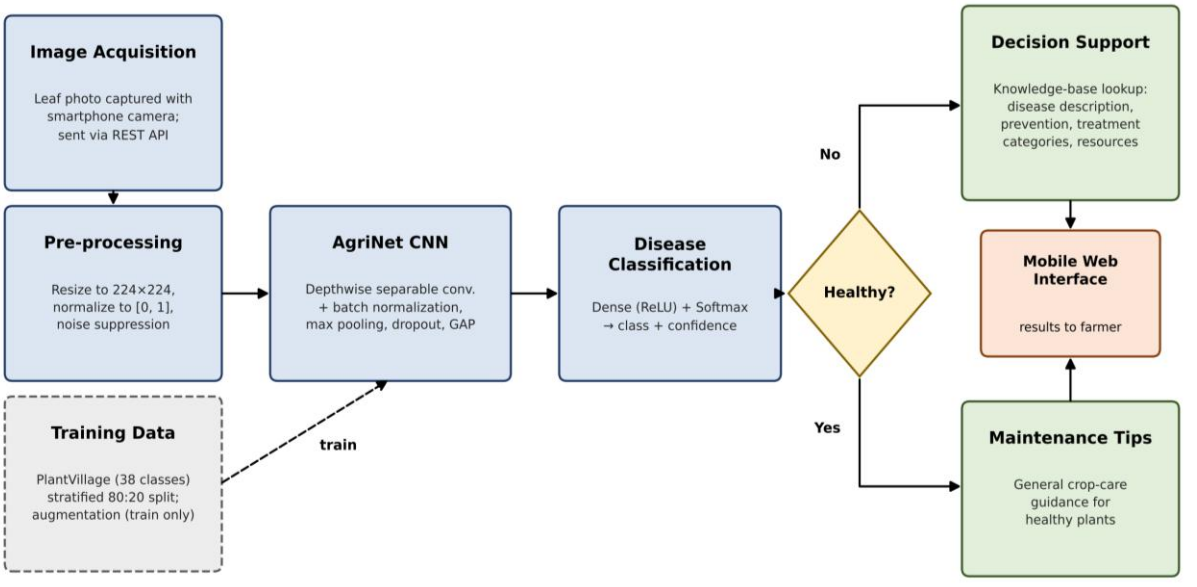

Fig. 2: Architecture of the proposed Integrated Agro-Diagnostic and Remediation Framework (IADRF).

classifications and confidence scores within 1–2 seconds using a CPU.

- A full end-to-end pipeline that removes the need for domain experts to perform first-level diagnosis in rural and semi-urban settings.

3.1.1 Architectural novelty and design rationale

The novel aspect of AgriNet is the interplay between the depthwise separable convolution, batch normalization and global average pooling. These components are individually well established in lightweight CNNs;[8,9,11] the novelty of AgriNet therefore lies not in the individual components but in their task-specific configuration into a compact architecture suited to multi-crop plant disease classification, and in its incorporation into the proposed agro-diagnostic decision-support pipeline. AgriNet aims to perform feature extraction with a small number of learnable parameters while retaining discriminative power across the 38 disease and healthy classes. The overall goals for the classification performance and computational efficiency and integration with an actionable agricultural decision-support layer are considered in architectural design, based on this. The novelty of the system is further substantiated by comparing the classification accuracy and general functionality of the system with current plant diseases detection approaches.

3.2 System architecture

The IADRF pipeline is outlined in five stages-each performing a specific task to turn an input into an output that can be acted upon.

3.2.1 Image acquisition

Any standard smartphone camera is used by a farmer to capture an image of a symptomatic leaf. The captured image is sent to the server-side pipeline with a REST API call through the mobile web interface.

3.2.2 Pre-processing module

The received image is resized to 224×224, and then, per channel, it is normalized in the range [0, 1], and noise is suppressed. This phase is also used during training to apply augmentation techniques like brightness change, flipping, rotation, and Gaussian noise to increase the model's generalization.

3.2.3 Feature extraction (AgriNet CNN)

A preprocessed image is fed to the AgriNet architecture, which extracts discriminative hierarchical features through the stacked depthwise separable convolutional blocks, with batch normalization and max pooling applied as listed in Table 1. Global average pooling in the penultimate layer averages the spatial feature maps into a single vector for each filter, which is less prone to overfitting than fully connected models.

3.2.4 Disease classification

A feature vector is fed into two dense layers, one using the ReLU activation and another using SoftMax activation. The result is a probability distribution of over 38 classes (26 disease classes and 12 healthy classes across 14 crop species). The class that has the maximum probability is returned as a prediction, and a confidence score is also returned.

3.2.5 Decision support and output

If a disease is found (that is, the predicted class is not "Healthy"), the decision support and output stage looks up a structured knowledge base, indexed by the predicted class. It fetches and presents: (a) disease description and spread mechanism, (b) preventive cultural practices, (c)

general treatment categories and preventive-management information compiled from expert advice and selected agricultural information sources, to support application decisions; and (d) links to authoritative agricultural information and extension resources. For the healthy case, general crop maintenance tips are shown. This architecture allows the system to be more than a classifier-it becomes a diagnostic and advisory system, available via a single mobile interface without the need for experts.

3.2.6 Responsible decision support and safety considerations

The decision support component should deliver information and support for the decision maker without diagnosing the agricultural problem and without giving regulatory advice. The disease prediction obtained from AgriNet is thus regarded as a preliminary diagnosis-marker. Treatment data is derived from a structured knowledge base and needs to be understood in the context of crops, severity, environment, product label and relevant agricultural legislation. The framework doesn't automatically approve pesticides usage or legally binding rates of application. Chemical products users are advised to apply chemical products in compliance with approved labels and competent agricultural extension staffs' recommendations, where appropriate. This design reduces the risk of a model prediction being used as a substitute for professional agronomic decision-making.

3.3 Detection algorithm

Algorithm 1: Plant_Disease_Detection
Input: Leaf Image captured via mobile device
Output: Disease Type, Confidence Score, Prevention Guidelines, Treatment Recommendations, E-commerce Links

1. Capture leaf image using mobile device camera.
2. Preprocess image: resize to 224 × 224 pixels, normalize pixel values to [0, 1], apply noise removal and data augmentation (rotation, flipping, brightness adjustment).
3. Load the trained AgriNet CNN model.
4. Extract hierarchical features from the preprocessed image using convolutional and pooling layers.
5. Classify the image into a disease class or healthy category; compute confidence score.
6. IF Disease_Type ≠ Healthy THEN
   • Retrieve disease-specific prevention guidelines
   • Retrieve fertilizer and pesticide recommendations
   • Generate links to agricultural e-commerce platforms
   ELSE Display healthy crop maintenance tips.
   END IF
7. Display results to farmer via mobile web interface.
8. Return: Disease Type, Confidence Score, Recommendations

4. Methodology

The approach involves a systematic process: data preparation, image processing, model development, model training, and prediction with a decision support system.

4.1 Data collection

We used the PlantVillage dataset for training. It is a public benchmark[5,28] that includes 61,486 high-quality images of leaves in 38 classes of 14 crop species (such as tomato, apple, grape, potato, cherry, peach, corn, and strawberry). These include both diseased and healthy leaves. The following Fig. 3 represents the overall system workflow for plant disease detection.

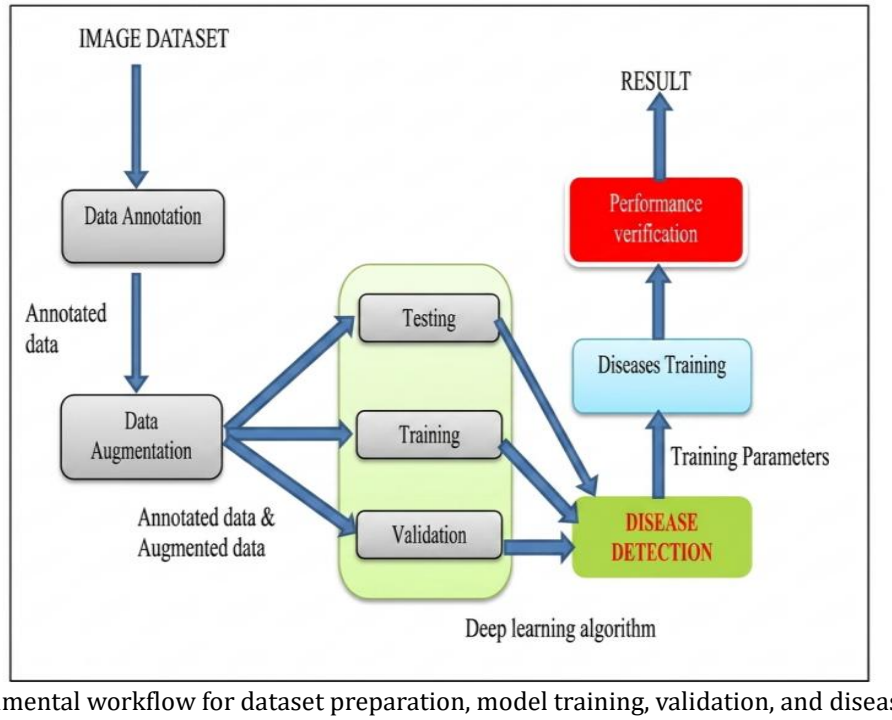

Fig. 3: Experimental workflow for dataset preparation, model training, validation, and disease prediction.

The dataset encompasses major categories of diseases, including fungal (e.g., powdery mildew, early blight), bacterial (e.g., bacterial spot), and viral (e.g., tomato yellow leaf curl virus) pathogens. To complement the Plant Village dataset and include more images from real field conditions, leaf images were collected from agricultural stations and publicly available agricultural datasets of crops common in the Indian subcontinent, such as sugarcane, guava, brinjal, and chili.

4.2 Image Preprocessing

The images were normalized to 224×224 pixels to maintain consistency in the dataset and to make the data suitable for feeding to a CNN. The pixel values were normalized to the range [0, 1] by dividing by 255 to increase the model's speed of learning and stability. The training dataset was augmented to increase diversity and prevent overfitting.[29] Augmentation techniques included:

Random rotation (up to 30 degrees)
Vertical and horizontal flipping
Brightness and contrast adjustment
Affine transformations (shear and zoom)
Gaussian noise addition

Augmented images were created using OpenCV and Python scripts. Augmentation was done offline prior to training. These images were saved at 256×256 and resized to the 224×224 input size for the model.

4.3 Model Architecture (AgriNet)

AgriNet is a new lightweight CNN for efficient classification of leaf diseases. It uses depthwise separable convolutions to achieve efficient computational performance while maintaining feature learning ability.[8,12] Each convolutional layer is succeeded by batch normalization to help with training stability, and global average pooling is applied to prevent overfitting.[9,11] Fig. 4 shows sample leaf images from the dataset and Fig. 5 shows sample augmented images used for model training.

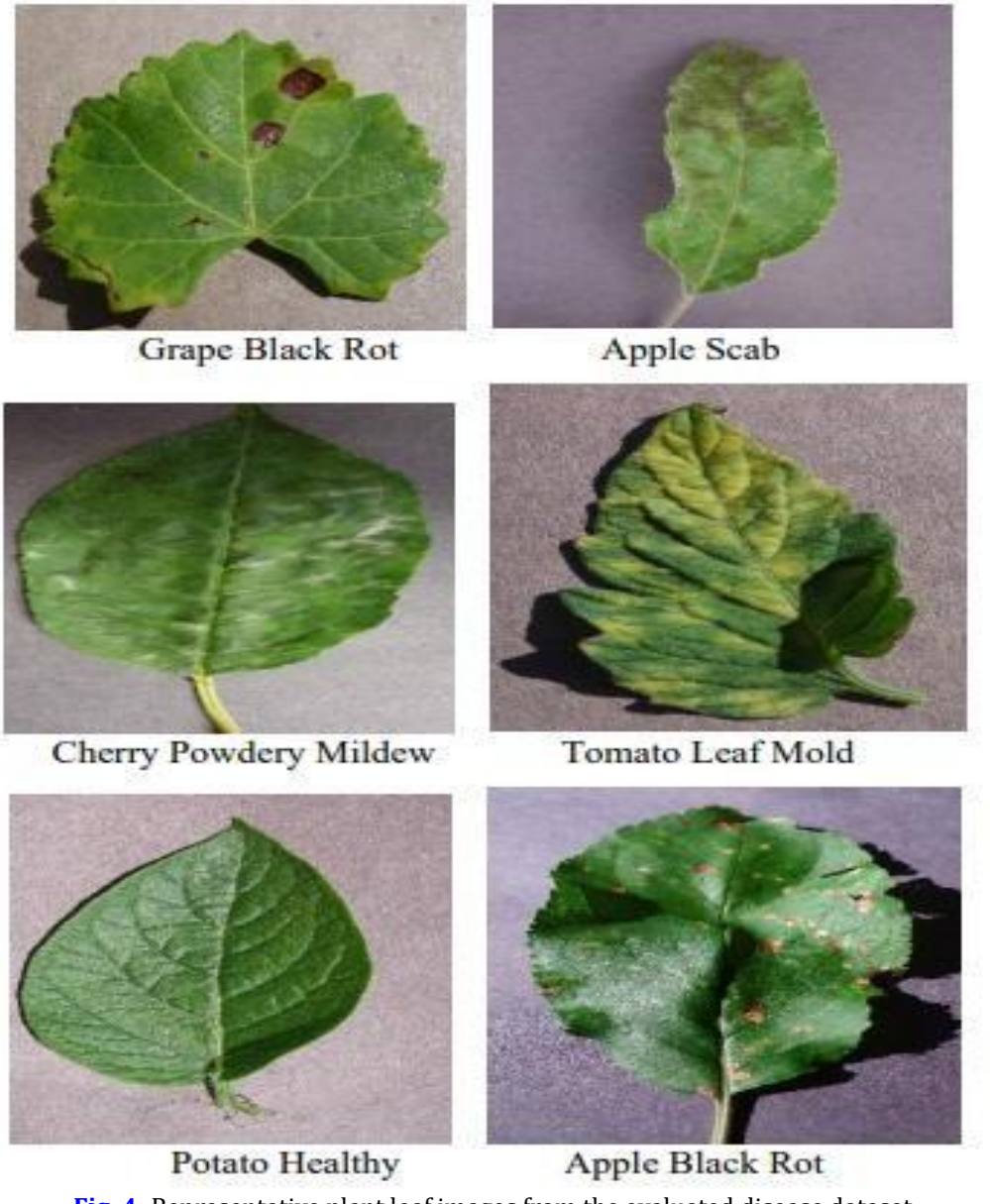

Fig. 4: Representative plant leaf images from the evaluated disease dataset.

Fig. 5: Examples of augmented leaf images generated using rotation, flipping, brightness/contrast adjustment, affine transformation, and Gaussian noise.

The final layer's Softmax function is defined as:

$$P(y = i) = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \quad (1)$$

where z_i is the output (logit) of the final dense layer for class i and $C = 38$ is the number of classes.
The loss function used for training is cross-entropy:

$$L = -\sum_{i=1}^{C} y_i \log(\hat{y}_i) \quad (2)$$

where y_i is the one-hot encoded ground-truth label and $\hat{y}_i$ is the predicted probability for class i. The loss is averaged over the images in each mini-batch.
Two dropout layers (rates of 0.25 and 0.5) are added for improved model stability. The full model comprises several blocks of convolutional layers with dense layers, as outlined in Table 1.

4.3.1 Model training

TensorFlow 2.x with the Keras API framework was used to implement the model, which was trained on a machine with an NVIDIA graphics processing unit (GPU). The data was split into training and testing sets (80–20) using stratified sampling to preserve class proportions. Fifteen epochs were run using the Adam optimizer[30] with an initial learning rate of 0.001 and categorical cross-entropy loss. Early stopping and reduction of the learning rate on plateau were also performed as callbacks during training.

The proposed AgriNet model's training loss converged at 0.0320, which is significantly lower than Xception (0.1066), ResNet-50 (1.1376), and Inception V3 (0.5003). This indicates that the proposed design was successful and the training setup was suitable for the task. The extra images collected from field conditions were further used with the training data to improve generalization.

Table 1: AgriNet model layer configuration

Layer	Output Dimension	Trainable Parameters
Separable Conv2D	218 × 218 × 32	155
Batch Normalization	218 × 218 × 32	128
Max Pooling	109 × 109 × 32	0
Separable Conv2D	107 × 107 × 64	2,400
Separable Conv2D	105 × 105 × 128	8,896
Dropout (0.25)	105 × 105 × 128	0
Max Pooling	52 × 52 × 128	0
Separable Conv2D	22 × 22 × 256	68,096
Separable Conv2D	20 × 20 × 512	133,888
Dropout (0.50)	20 × 20 × 512	0
Global Average Pooling	512	0
Dense (ReLU)	1024	525,312
Dense (Softmax)	38	38,950

5. Experimental setup

The experimental environment and organization details are provided below to secure consistency.

Table 2: Experimental configuration

Parameter	Details
Dataset	PlantVillage (61,486 images, 38 classes)
Image Input Size	224 × 224 × 3 (RGB)
Batch Size	32
Framework	TensorFlow 2.x with Keras
Loss Function	Categorical Cross-Entropy
Epochs	15
Train/Test Split	80% Training / 20% Testing (stratified)
Regularization	Dropout (0.25, 0.5), Batch Normalization
GPU	NVIDIA GPU (CUDA-enabled)
Programming Language	Python 3.8
Optimizer	Adam (lr = 0.001)
Web Backend	Flask (REST API)
Frontend	HTML5, CSS3, JavaScript

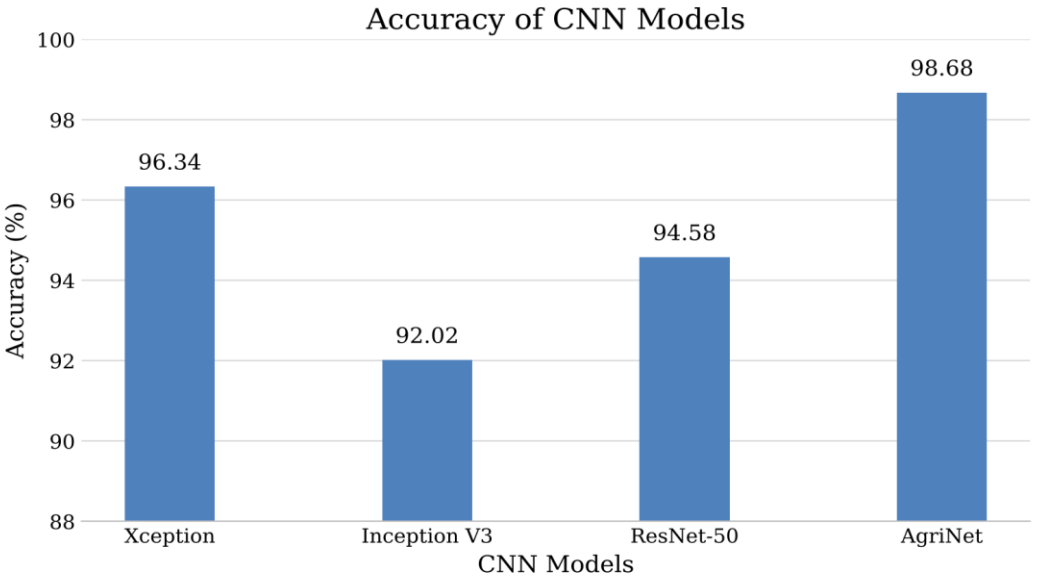

Fig. 6: Accuracy comparison of AgriNet and baseline CNN architectures.

5.1 Reproducibility and experimental protocol

To ensure reproducibility, all experiments were run with the same random seed 42. Stratified sampling was used to make the distribution of classes in each subset of the data set like that of the original data set. After partitioning the dataset, data augmentation was applied only to the training subset, so as not to have the same image in the augmented images in the evaluation dataset. Random rotation, flipping (both horizontal and vertical), brightness and contrast changes, affine transformations and Gaussian noise were used in the augmentation pipeline. All the models were trained with the same input resolution, batch size, optimizer, learning rate, number of training epochs, loss function and data split. The proposed AgriNet has around 805,001 trainable parameters. To allow the specific software/hardware configuration to be replicated, the software and hardware are reported in Table 2. Fig. 6 shows the accuracy comparison of various CNN models.

6. Results and discussion

The AgriNet model was compared with three popular CNN models—Xception, ResNet-50, and Inception V3—to test their classification accuracy under identical experimental conditions by training on the PlantVillage dataset for 15 epochs.

6.1 Classification accuracy

Table 3 shows the classification accuracy of the different CNN models. The proposed AgriNet model outperforms other architectures with an accuracy of 98.68%, followed by Xception (96.34%), ResNet-50 (94.58%), and Inception V3 (92.02%).

6.2 Precision, recall, and F1-score

In addition to accuracy, the model was tested for the precision, recall, and F1 scores for each class. Table 4 shows the overall (macro) average performances for all architectures. Our proposed AgriNet model works best among all the other models in all three of the metrics, which indicates that the model also performs well on the smaller (minority) disease classes.

Table 3: Comparative analysis of architectures

S. No	Architecture	Accuracy (%)	Epochs
1	Xception	96.34%	15
2	Inception V3	92.02%	15
3	ResNet-50	94.58%	15
4	AgriNet (Proposed System)	98.68%	15

All metrics in Tables 3 and 4 were computed from the predictions of each model on the same held-out stratified test set (20% of the data), using the same evaluation procedure for every architecture.[31] Accuracy is the proportion of correctly classified test images:

$$\text{Accuracy} = \frac{\sum_{c=1}^{C} \text{TP}_c}{N} \tag{3}$$

where N is the number of test images and TP_c is the number of correctly classified images of class c. Per-class precision and recall are defined as:

$$P_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c}, \qquad R_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c} \tag{4}$$

and the values in Table 4 are macro averages over the C = 38 classes, with the F1-score obtained as the harmonic mean of macro precision and macro recall:

$$P_{\text{macro}} = \frac{1}{C}\sum_{c=1}^{C} P_c, \qquad R_{\text{macro}} = \frac{1}{C}\sum_{c=1}^{C} R_c \tag{5}$$

$$F1 = \frac{2 \cdot P_{\text{macro}} \cdot R_{\text{macro}}}{P_{\text{macro}} + R_{\text{macro}}} \quad (6)$$

Because accuracy pools all test images, it is equivalent to the micro-averaged (sample-weighted) recall, whereas macro averaging gives each class equal weight irrespective of its size. As the PlantVillage classes are imbalanced, macro-averaged precision, recall and F1-score are expected to be slightly lower than the overall accuracy; the small differences between Table 3 and Table 4 therefore arise from this difference in averaging and not from different test data or evaluation protocols.

Table 4: Comparison of precision, recall, and F1-score metrics

Architecture	Precision (%)	Recall (%)	F1-Score (%)
AgriNet (Proposed)	98.51%	98.43%	98.47%
Xception	96.12%	95.89%	95.99%
ResNet-50	94.20%	93.76%	93.98%
Inception V3	91.80%	91.35%	91.57%

6.3 Loss function analysis

The final training loss values achieved by all tested architectures show that the proposed AgriNet attains the lowest loss of 0.0320, followed by Xception (0.1066), Inception V3 (0.5003), and ResNet-50 (1.1376). A low training loss alone does not establish the absence of overfitting; it is the consistency between the low training loss and the held-out test performance reported in Tables 3 and 4 that indicates good generalization. The low loss of AgriNet is attributed to the use of depthwise separable convolutions, which reduce over-parameterization, and global average pooling, which provides stronger regularization than fully connected layers.

6.4 Confusion matrix analysis

The confusion matrix of AgriNet, computed on the same held-out stratified test set used for Tables 3 and 4, is shown in Fig. 7, with strong dominance along the diagonal, indicating correct predictions. Minor misclassifications happen when there are some similarities in the disease categories, such as between early and late blight in tomato, and between healthy and mildly infected leaves—challenges that human experts face as well. The overall diagonal dominance of the confusion matrix confirms that the model effectively learns discriminative features for each disease group.

6.5 System performance discussion

The proposed AgriNet architecture achieves substantially better results than Xception and ResNet-50 despite having far fewer parameters (805,001 vs. millions in ResNet and Xception). This demonstrates that the architectural decisions-depthwise separable convolutions, batch normalization, and global average pooling—are particularly effective for plant disease classification. The lightweight design also makes the model potentially suitable for resource-constrained deployment, subject to dedicated mobile-device validation and low-resource server environments, which is essential for rural agricultural applications. The measured inference latency results show that AgriNet has

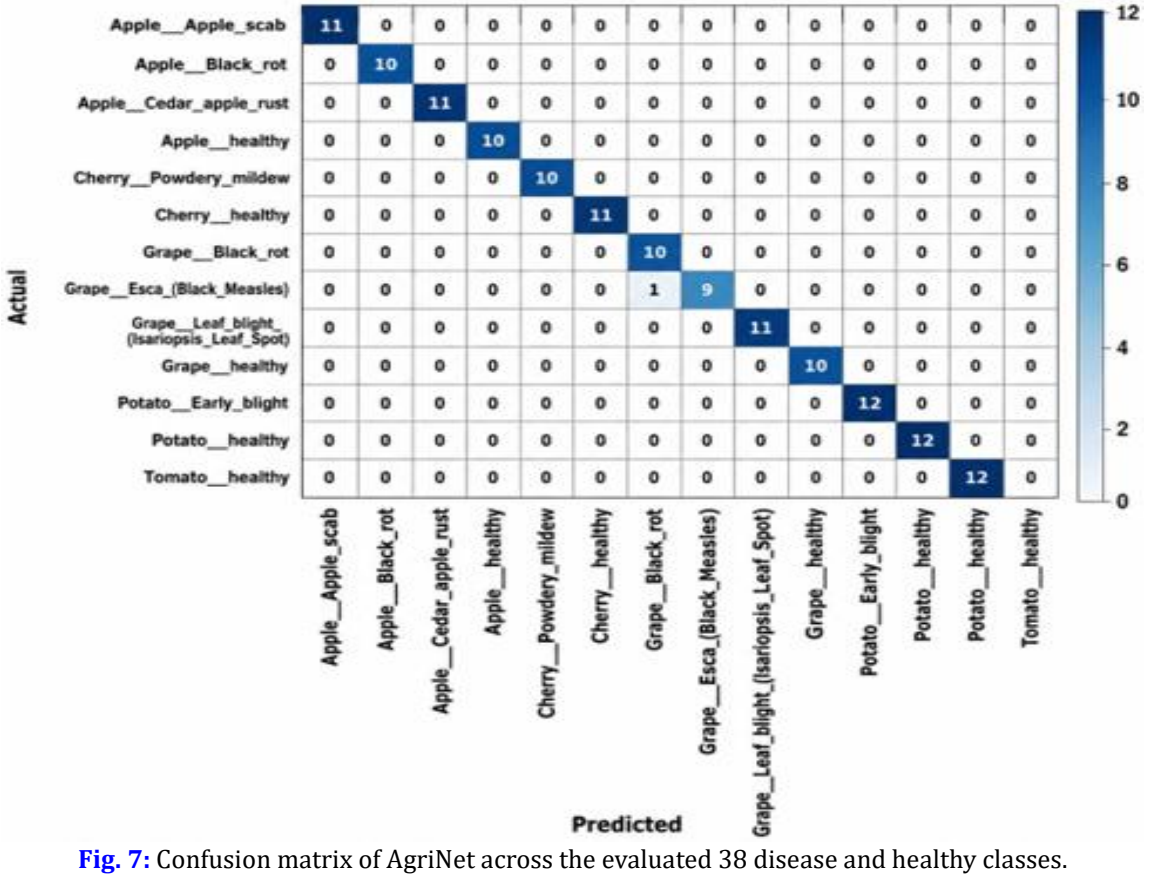

Fig. 7: Confusion matrix of AgriNet across the evaluated 38 disease and healthy classes.

properties that are suitable for deploying on a resource constrained system. These CPU/GPU inference measures reflect the CPU/GPU performance, but they don't necessarily reflect the performance of smartphones. The deployment of mobile devices would also be dependent on the processor power, memory, operating system, setting up of the network, image model conversion, acquisition conditions, *etc*., of the images. Accordingly, the feasibility of the tested server/CPU/GPU is being checked, and immediate deployment validation in the field is considered as the next step which would need to be carried out with representative smartphones. The integration of the decision support module provides practical value beyond disease classification alone. By combining high-precision disease detection with treatment recommendations and e-commerce connectivity, IADRF offers a comprehensive end-to-end solution for farm-level disease management. The mean inference time per image is approximately 0.3 seconds on GPU and 1–2 seconds on CPU, confirming that the proposed AgriNet model is suitable for real-time deployment in mobile and web-based applications.

6.6 Beyond accuracy: System-level contribution

Evaluation of the contribution of the proposed framework does not depend on accuracy of classification only. The effectiveness of the system is evaluated through the accuracy of the system (AgriNet: 98.68% accuracy), the F1-score (AgriNet: 98.47% F1-score) and other features such as predictive capability, model compactness, computational efficiency, inference latency, integration with agricultural decision support. The proposed architecture is a lot simpler in terms of the number of parameters used in comparison with the conventional large CNN architecture and the overall architecture embeds disease classification together with disease-specific preventive information and treatment guidance. The proposed contribution is therefore not only focused on accuracy but is targeted towards system-level integration of lightweight visual diagnosis and decision support. In comparative assessment, both model level metrics and end-to-end functionality should be assessed.

7. Limitations

Though the proposed AgriNet model performs very well on the PlantVillage dataset, this dataset is primarily composed of images taken under controlled conditions. In actual agricultural settings, factors such as uneven lighting, complex backgrounds, occlusions, and image noise may influence model performance.[2,5] Additionally, the model relies solely on image-based features and does not capture environmental conditions such as weather or soil conditions, which can also influence disease occurrence. Future research will explore datasets collected in the field and different data modalities.

8. Conclusion and future work

This paper introduced the Integrated Agro-Diagnostic and Remediation Framework (IADRF), which is a deep learning-based platform for plant leaf disease diagnosis and remediation guidance for farmers. The proposed CNN AgriNet, trained on the PlantVillage dataset (61,486 images, 38 classes), attained an accuracy of 98.68%, an F1-score of 98.47%, and the minimum training loss (0.0320), outperforming the other three models (Xception (96.34%), ResNet-50 (94.58%), and Inception V3 (92.02%)) under the same conditions. In contrast to the traditional systems that only classify the disease, IADRF incorporates a decision support module that provides disease-specific prevention guidelines, fertilizer and pesticide recommendations, and links to e-commerce platforms for agriculture, all through a mobile-friendly web-based application. The framework offers a straightforward approach to an initial screening for the first level of disease and structured agricultural information via an interface that can be accessed on a mobile device. It is not a substitute for agronomic diagnosis, particularly in the selection of the various treatment options and disease management in the field. Going forward, the focus will be on four areas: (1) making use of regional field-collected images of Indian crops for better generalization to real-world scenarios; (2) incorporation of soil and weather information in a multimodal diagnosis framework; (3) federated on-device learning to learn from farmer data while keeping privacy intact; and (4) development of multilingual interfaces to cater to non-English-speaking farming communities.

CRediT Author Contribution Statement

V. Lokeswara Reddy: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – Original draft. **V. Srilatha**: Methodology, Supervision, Project administration, Validation, Writing – Review and editing. **V. Ganga Dev:** Data curation, Investigation, Resources, Writing – Review and editing. **P. Shiva Durga:** Data curation, Investigation, Resources, Writing – review and editing. **R. Tharun**: Data curation, Investigation, Resources, Writing – Review and editing. **T. Harsha Vardhan Reddy**: Methodology, Validation, Writing – Review and editing. All authors have read and approved the final version of the manuscript.

Funding Statement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data Availability Statement

The PlantVillage dataset analysed in this study is publicly available(https://github.com/spMohanty/PlantVillage-Dataset). The additional field images, the trained AgriNet

model weights, the training and evaluation code, and the complete confusion-matrix data are available from the corresponding author upon reasonable request.

Conflict of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Artificial Intelligence (AI) Use Disclosure

The authors declare that artificial intelligence (AI)-assisted tools were used only for language refinement, grammar improvement, and manuscript structuring purposes during the preparation of this work. All technical content, experimental implementation, results, and interpretations were independently developed and verified by the authors, who take full responsibility for the content of this publication.

Supporting Information

Not applicable.

References

[1] S. Savary, L. Willocquet, S. J. Pethybridge, P. Esker, N. McRoberts, A. Nelson, The global burden of pathogens and pests on major food crops, *Nature Ecology & Evolution*, 2019, **3**, 430–439, doi: 10.1038/s41559-018-0793-y.

[2] J. G. A. Barbedo, Factors influencing the use of deep learning for plant disease recognition, *Biosystems Engineering*, 2018, **172**, 84–91, doi: 10.1016/j.biosystemseng.2018.05.013.

[3] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, *Nature*, 2015, **521**, 436–444, doi: 10.1038/nature14539.

[4] A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet classification with deep convolutional neural networks, Advances in Neural Information Processing Systems, Curran Associates, Inc., 2012, **25**, 1097–1105.

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

[6] K. P. Ferentinos, Deep learning models for plant disease detection and diagnosis, *Computers and Electronics in Agriculture*, 2018, **145**, 311–318, doi: 10.1016/j.compag.2018.01.009.

[7] A. Kamilaris, F. X. Prenafeta-Boldú, Deep learning in agriculture: A survey, *Computers and Electronics in Agriculture*, 2018, **147**, 70–90, doi: 10.1016/j.compag.2018.02.016.

[8] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, MobileNets: Efficient convolutional neural networks for mobile vision applications, *arXiv preprint*, 2017, arXiv:1704.04861, doi: 10.48550/arXiv.1704.04861.

[9] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, Proceedings of the 32nd International Conference on Machine Learning, PMLR, 2015, **37**, 448–456.

[10] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, *Journal of Machine Learning Research*, 2014, **15**, 1929–1958.

[11] M. Lin, Q. Chen, S. Yan, Network in network, *arXiv preprint*, 2013, arXiv:1312.4400, doi: 10.48550/arXiv.1312.4400.

[12] F. Chollet, Xception: Deep learning with depthwise separable convolutions, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2017, 1251–1258, doi: 10.1109/CVPR.2017.195.

[13] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2016, 770–778, doi: 10.1109/CVPR.2016.90.

[14] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2016, 2818–2826, doi: 10.1109/CVPR.2016.308.

[15] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, **37**, 1904–1916, doi: 10.1109/TPAMI.2015.2389824.

[16] M. Hu, X. Bu, X. Sun, Z. Yu, Y. Zheng, Rape plant disease recognition method of multi-feature fusion based on D–S evidence theory, *Mathematical and Computational Applications*, 2017, **22**, 18, doi: 10.3390/mca22010018.

[17] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, S. Thrun, Dermatologist-level classification of skin cancer with deep neural networks, *Nature*, 2017, **542**, 115–118, doi: 10.1038/nature21056.

[18] E. Alehegn, Maize leaf diseases recognition and classification based on imaging and machine learning techniques, Ph.D. dissertation, BAHIR DAR UNIVERSITY, 2020, https://ir.bdu.edu.et/bitstream/handle/123456789/10759/Enquhone_Alehegn_Oct_2017_

Final_Thesis.pdf?sequence=1&isAllowed=y

[19] X. Zhang, S. Wang, J. Liu, C. Tao, Towards improving diagnosis of skin diseases by combining deep neural network and human knowledge, *BMC Medical Informatics and Decision Making*, 2018, **18**, 59, doi: 10.1186/s12911-018-0631-9.

[20] H. Beyene, D. N. A. Joshi, D. K. Kotecha, Plant diseases prediction using image processing and machine learning techniques: A survey, *International Journal of Computer Applications*, 2018, **1,** 2250-1797, doi: 10.26808/rs.ca.i8v1.23.

[21] Z. Zhang, H. Liu, Z. Meng, J. Chen, Deep learning-based automatic recognition network of agricultural machinery images, *Computers and Electronics in Agriculture*, 2019, **166**, 104978, doi: 10.1016/j.compag.2019.104978.

[22] S. Bhuvaneswari, R. Surendiran, R. Aarthi, M. Thangamani, R. B. Lingisetti, Disease detection in plant leaf using LNet based on deep learning, *International Journal of Engineering Trends and Technology*, 2022, **70**(9), 64–75, doi: 10.14445/22315381/IJETT-V70I9P207.

[23] H. Eghbali, N. Hajihosseini, Deep convolutional neural network (CNN) for large-scale image classification, *SSRN Electronic Journal*, 2019, doi: 10.2139/ssrn.3476258.

[24] W. A. Ezat, M. M. Dessouky, N. A. Ismail, Multi-class image classification using deep learning algorithm, *Journal of Physics: Conference Series*, 2020, **1447**, 012021, doi: 10.1088/1742-6596/1447/1/012021.

[25] A. Rastogi, R. Arora, S. Sharma, Leaf disease detection and grading using computer vision technology and fuzzy logic, Proceedings of the 2nd International Conference on Signal Processing and Integrated Networks, *IEEE*, 2015, 500–505, doi: 10.1109/SPIN.2015.7095350.

[26] M. M. ur Rehman, J. Liu, A. Nijabat, M. Faheem, W. Wang, S. Zhao, Leveraging convolutional neural networks for disease detection in vegetables: A comprehensive review, *Agronomy*, 2024, **14**, 2231, doi: 10.3390/agronomy14102231.

[27] C. K. Sunil, C. D. Jaidhar, N. Patil, Cardamom plant disease detection approach using EfficientNetV2, *IEEE Access*, 2022, **10**, 789–804, doi: 10.1109/ACCESS.2021.3138920.

[28] D. P. Hughes, M. Salathé, An open access repository of images on plant health to enable the development of mobile disease diagnostics, *arXiv preprint*, 2015, arXiv:1511.08060, doi: 10.48550/arXiv.1511.08060.

[29] C. Shorten, T. M. Khoshgoftaar, A survey on image data augmentation for deep learning, *Journal of Big Data*, 2019, **6**, 60, doi: 10.1186/s40537-019-0197-0.

[30] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, *arXiv preprint*, 2014, arXiv:1412.6980, doi: 10.48550/arXiv.1412.6980.

[31] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, *Information Processing & Management*, 2009, **45**, 427–437, doi: 10.1016/j.ipm.2009.03.002.

Publisher Note: The views, statements, and data in all publications solely belong to the authors and contributors. GR Scholastic is not responsible for any injury resulting from the ideas, methods, or products mentioned. GR Scholastic remains neutral regarding jurisdictional claims in published maps and institutional affiliations.