Deep Learning-Based Plant Disease Detection Using Computer Vision

Gauri Jadhav[1], Aarti Pimpalkar[2], Mayuresh Gulame[3], Ashwini Shahapurkar[4]
[1,2,3,4]*MIT School of Computing, MIT Art, Design and Technology University, Pune*

***Abstract*—Plant diseases are a major challenge in agriculture, as they can affect crop quality and reduce agricultural productivity. Early identification of plant diseases is important for supporting effective crop management. This research presents a computer vision-based approach for automated plant disease classification using the EfficientNet-B0 architecture. The PlantVillage dataset, consisting of 54,305 RGB images from 14 crop species and 38 classification categories, is used for the experiment. The images are preprocessed through resizing, normalization, and data augmentation techniques to improve the consistency and diversity of training samples. A modified classification layer with dropout is used to produce predictions for the 38 categories. The dataset is divided into training, validation, and testing sets using stratified sampling. The recorded experimental evaluation on 8,146 test images reports an accuracy of 96.94%, a macro F1-score of 95.87%, and a weighted F1-score of 96.97%. These results indicate the potential of image-based deep learning approaches for plant disease classification. However, further evaluation using real-field images is required to assess the system's performance under practical agricultural conditions.**

***Index Terms*—computer vision, convolutional neural network, deep learning, EfficientNet-B0, image classification, Plant disease detection, PlantVillage dataset.**

I. INTRODUCTION

Agriculture plays an important role in food production and the economic development of many countries. However, plant diseases remain a major challenge because they can affect crop quality, reduce agricultural productivity, and cause economic losses. Early identification of plant diseases is important because it helps farmers take suitable actions before the infection spreads. Traditional disease identification mainly depends on visual inspection by farmers or agricultural experts. However, this process can be time-consuming and may require considerable experience, particularly when different diseases show similar symptoms [1].

Recent developments in computer vision and deep learning have created new opportunities for automated plant disease detection. Convolutional Neural Networks (CNNs) can automatically learn visual features such as colour, shape, edges, and texture from leaf images. These features help the model distinguish between healthy and diseased plant leaves. Previous research has demonstrated the effectiveness of deep learning techniques for identifying multiple plant diseases using image datasets [1]. Recent studies have also explored different deep learning architectures and their applications in plant disease classification [2].

Despite these developments, accurate plant disease classification remains challenging. Models may face difficulties when disease symptoms look similar, when the available training images are limited, or when images contain variations in lighting and background conditions. Research has also highlighted the importance of developing models that can perform effectively across different plant disease datasets [3]. Therefore, selecting a suitable deep learning architecture and evaluating its performance using an independent test dataset are important steps in developing a reliable classification system.

This research presents a deep learning-based plant disease classification system using the EfficientNet-B0 architecture and computer vision techniques. The PlantVillage dataset is used for model development and evaluation. The dataset contains 54,305 RGB leaf images belonging to 38 classification classes and representing 14 crop species. The input images are

resized to 224 × 224 pixels and normalized before being passed to the model. Random horizontal flipping, rotation, and colour adjustment are applied during training to introduce image variations.

The EfficientNet-B0 model is trained from random initialization using the PyTorch framework. A dropout layer with a probability of 0.30 and a fully connected classification layer are added to produce predictions for 38 classes. The model is trained using the AdamW optimizer with different learning rates for the feature extraction layers and classifier. Weighted cross-entropy is used as the loss function. The dataset is divided into training, validation, and testing subsets using a stratified splitting strategy.

The main objectives of this research are:

1) To develop a deep learning-based system for classifying healthy and diseased plant leaf images.
2) To apply image preprocessing and data augmentation techniques to prepare the dataset for model training.
3) To implement EfficientNet-B0 for multi-class plant disease classification.
4) To evaluate the model using accuracy, precision, recall, F1-score, and confusion matrix analysis.
5) To examine the classification results and identify areas for future improvement.

The proposed model is evaluated using an independent test dataset containing 8,146 images. The experimental results achieve an accuracy of 96.94%, a macro F1-score of 95.87%, and a weighted F1-score of 96.97%. These results demonstrate the model's ability to classify the selected plant disease categories within the PlantVillage dataset.

II. LITERATURE STUDY

Several researchers have explored deep learning techniques to improve the accuracy of automated plant disease detection. Different approaches, including transfer learning, modified CNN architectures, and multi-stage classification systems, have been developed to identify plant diseases from leaf images.

Saleem et al. [4] proposed a weight optimization-based transfer learning approach for detecting plant diseases in New Zealand vegetables. Their study used datasets such as PlantVillage, NZDLPlantDisease-v1, and DeepWeeds to optimize model weights. The proposed approach used a Faster R-CNN Inception ResNet-v2 architecture and achieved a mean Average Precision (mAP) of 91.33%. The study highlighted the importance of dataset diversity and optimized model weights in improving plant disease detection.

Wu et al. [5] developed an improved ResNet101 model for recognizing leaf diseases in woody fruit plants. Their approach incorporated techniques such as attention mechanisms, global average pooling, dropout, and regularization. The model was evaluated using 14,433 images covering 25 categories of healthy and diseased leaves. It achieved an accuracy of 85.90%, demonstrating the effectiveness of modifying existing CNN architectures for plant disease classification.

Jung et al. [6] proposed a deep learning-based multi-stage plant disease detection system. Their approach involved identifying plant categories, detecting diseases, and classifying specific disease types. The researchers evaluated different pretrained CNN architectures, including EfficientNet, VGG19, AlexNet, GoogLeNet, and ResNet50. Their experiments reported strong classification performance, including 99.40% accuracy for the final potato disease classification model and 97.09% for tomato disease classification. The study also examined the use of an unknown category to identify crops that were not included during model development.

The existing studies demonstrate that deep learning architectures can achieve promising results in plant disease identification. However, their performance depends on factors such as dataset characteristics, model architecture, training strategy, and image quality. Based on these studies, the present research implements EfficientNet-B0 for classifying 38 plant health and disease categories. Unlike the transfer learning approaches discussed above, the proposed model is trained from random initialization and evaluated using a separate test dataset.

TABLE I. SUMMARY OF ADDITIONAL RESEARCH ON PLANT DISEASE DETECTION

Source	Technique	Advantages	Limitations
[7]	CNN-based plant disease identification	Automatically extracts disease-related features from leaf images and supports classification across different plant disease categories	Performance may be affected by variations in real-world images and complex backgrounds.
[8]	Modified Faster R-CNN	Supports plant disease localization and classification through region-based feature extraction.	Region-based detection can require considerable computational resources.
[9]	Multi-stage pretrained CNN classification	Combines crop identification, disease detection, and disease classification into multiple stages.	Multiple processing stages can increase system complexity.
[10]	Machine learning and deep learning-based review	Provides an overview of existing techniques and discusses different approaches to plant disease detection.	Does not introduce a new classification model or experimental implementation.

III. DATASET DESCRIPTION

A. Dataset Overview

The PlantVillage dataset is used for developing and evaluating the proposed plant disease classification model. It contains images of healthy and diseased plant leaves collected from different crop species. The dataset consists of 54,305 RGB images belonging to 38 classification classes across 14 crop species.

Each image is associated with a specific crop and its corresponding health condition or disease category. The dataset supports multi-class image classification, allowing the proposed model to identify different plant disease categories rather than focusing on a single crop.

B. Image Preprocessing

Before training, all input images are resized to 224 × 224 pixels to maintain a consistent input size for the EfficientNet-B0 architecture. The images are converted into tensors and normalized using the ImageNet mean and standard deviation values.

The following data augmentation techniques are applied to the training images:

1) Random Horizontal Flip: Flips images horizontally to introduce variations in leaf orientation.
2) Random Rotation: Rotates images by up to 15 degrees to improve variation in image positioning.
3) Colour Adjustment: Changes brightness, contrast, and saturation to introduce variations in leaf appearance.

These transformations are applied during training, while validation and testing images undergo resizing and normalization without random augmentation.

C. Dataset Splitting

The dataset is divided into three subsets using stratified sampling to maintain class representation across the partitions. A fixed random seed of 42 is used to support reproducibility.

- Training set: 38,013 images (70%).
- Validation set: 8,146 images (15%).
- Testing set: 8,146 images (15%).

The training set is used to update the model parameters, while the validation set is used to monitor performance and select the best model checkpoint. The independent testing set is used for final performance evaluation.

TABLE II. DETAILS OF THE PLANTVILLAGE DATASET

Parameter	Description
Dataset name	Plant Village
Total images	54305
Crop species	14
Classification classes	38
Image format	RGB
Input image size	224 × 224 pixels
Training images	38,013
Validation images	8,146
Testing images	8,146
Splitting method	Stratified sampling
Random seed	42

Parameter	Description
Classification type	Multi-class classification

D. Dataset Characteristics

The PlantVillage dataset provides a useful collection of labelled leaf images for evaluating image-based plant disease classification techniques. However, most images were captured under controlled conditions. Therefore, the results obtained using this dataset may not fully represent real agricultural environments, where lighting variations, complex backgrounds, overlapping leaves, and different image-capturing conditions can affect classification performance.

IV. PROPOSED METHODOLOGY

The proposed system uses the EfficientNet-B0 convolutional neural network to classify plant leaf images into 38 categories. The methodology consists of image preprocessing, data augmentation, feature extraction, classification, model training, and performance evaluation. The complete workflow is presented in Fig. 1.

A. Overall System Architecture

The proposed system begins with RGB leaf images from the PlantVillage dataset. The dataset is divided into training, validation, and testing subsets using stratified sampling. Each image is resized and normalized before being passed to the EfficientNet-B0 model.

The model processes the input image and produces 38 output scores corresponding to the plant health and disease categories. A dropout layer is included to reduce overfitting, followed by a fully connected layer for final classification. During training, the validation set is used to identify the best model checkpoint, while the independent test set is used for final evaluation.

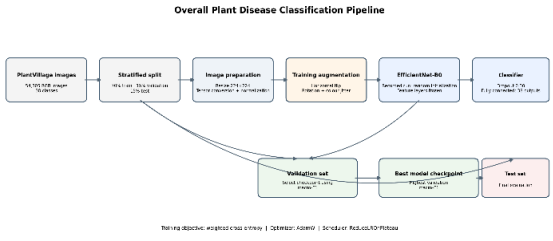


Fig. 1. Overall architecture of the proposed plant disease classification system.

B. Image Preprocessing and Augmentation

Image preprocessing is performed to ensure that all input images have a consistent format and size. Each RGB image is resized to 224 × 224 pixels and converted into a tensor. Normalization is then applied using the ImageNet mean and standard deviation values [11].

To increase the diversity of training samples, data augmentation is applied only to the training images. The techniques include:

1) Random Horizontal Flip: Horizontally flips images to introduce variations in leaf orientation.
2) Random Rotation: Rotates images randomly within a range of 15 degrees.
3) Color Jitter: Adjusts brightness, contrast, and saturation to introduce colour variations.

Validation and testing images are resized and normalized without random augmentation. This ensures that model evaluation is performed on images without additional random transformations.

C. EfficientNet-B0 Architecture

EfficientNet-B0 is used as the main convolutional neural network architecture for plant disease classification. It is designed to provide an efficient balance between model size and computational performance.

The architecture uses convolutional operations and mobile inverted bottleneck convolution (MBConv) blocks to learn visual patterns from input images. These operations help the network identify important characteristics such as leaf shape, texture, colour variations, and visible disease symptoms.

In the recorded implementation, EfficientNet-B0 is initialized without pretrained weights because the pretrained weights could not be loaded. The feature-extraction layers are kept frozen in the supplied model implementation. Therefore, the classification layer is trained using the dataset, while the randomly initialized feature-extraction layers remain unchanged.

D. Modified Classification Layer

The original classification layer of EfficientNet-B0 is replaced to support the 38 plant disease categories in the dataset. The modified classifier consists of two main components:

1) Dropout Layer: A dropout rate of 0.30 is applied to reduce overfitting during training.

2) Fully Connected Layer: A linear layer maps the extracted feature representation to 38 output classes.

The model produces 38 output scores for each input image. The class with the highest output score is selected as the predicted plant disease category [12].

E. Model Training and Optimization

The model is trained using the PyTorch framework. Weighted Cross-Entropy Loss is used as the loss function to account for differences in the number of images available across classes. The class weights are calculated using the square root of the ratio between the maximum class count and the individual class count.

$$L = -\frac{1}{N}\sum_{i=1}^{N} w_{y_i} \log\left(p_{i,y_i}\right) \quad (1)$$

The AdamW optimizer is configured with a learning rate of 0.001 for the classifier and 0.0001 for the feature-extraction parameter group. A weight decay value of 0.0001 is used for regularization. However, because the feature-extraction layers are frozen in the recorded implementation, their parameters are not updated during training.

A ReduceLROnPlateau learning rate scheduler is used to monitor validation macro F1-score. The learning rate is reduced by a factor of 0.3 when the monitored performance stops improving for two consecutive epochs.

The training configuration uses a batch size of 16 and a maximum of 10 epochs. A fixed random seed of 42 is used to support reproducibility [13].

F. Model Selection and Testing

The validation dataset is used to monitor the model's performance after each training epoch. Macro F1-score is selected as the main evaluation measure for choosing the best model checkpoint. The checkpoint with the highest validation macro F1-score is saved for final evaluation.

After training, the selected checkpoint is loaded and evaluated using the independent testing dataset containing 8,146 images. Accuracy, precision, recall, F1-score, and test loss are calculated to measure classification performance.

The confusion matrix and classification report are also generated to examine the model's predictions across the 38 plant disease categories. These evaluation results are discussed in the following sections.

V. EXPERIMENTAL SETUP AND RESULTS

A. Experimental Setup

The proposed plant disease classification model is implemented using Python and the PyTorch deep learning framework. Training and evaluation are performed using the NVIDIA GPU available in the experimental environment.

The model is trained using the PlantVillage dataset, which contains 54,305 images belonging to 38 classes. The images are resized to 224 × 224 pixels before being provided to the EfficientNet-B0 architecture.

TABLE III. EXPERIMENTAL CONFIGURATION

Parameter	Configuration
Deep learning framework	PyTorch
Model architecture	EfficientNet-B0
Dataset	PlantVillage
Total images	54,305
Number of classes	38
Input image size	224 × 224
Batch size	16
Number of epochs	10
Optimizer	AdamW
Loss function	Weighted Cross-Entropy Loss
Classifier learning rate	0.001
Feature parameter group learning rate	0.0001
Weight decay	0.0001
Random seed	42
Evaluation metric	Accuracy, Precision, Recall, F1-score

B. Training Performance

The model is trained for 10 epochs using the training dataset. Training loss, validation loss, training accuracy, and validation accuracy are recorded during each epoch to monitor the learning process.

The training loss decreases from 2.1272 in the first epoch to 0.1848 in the tenth epoch. Similarly, training accuracy increases from 43.54% to 94.36%. Validation accuracy reaches 96.78% in the final

epoch, while the validation macro F1-score reaches 95.70%.

These observations indicate that the recorded training process shows a reduction in loss and an improvement in classification performance over the epochs.

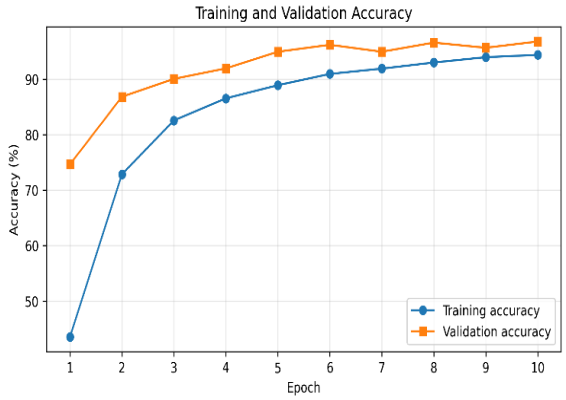


Fig. 2. Training and validation accuracy across 10 epochs.

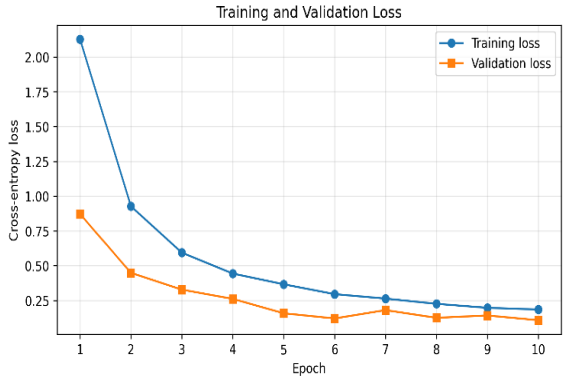


Fig. 3. Training and validation loss across 10 epochs

C. Test Dataset Evaluation

After training, the selected model checkpoint is evaluated using the independent test dataset containing 8,146 images. Accuracy, precision, recall, and F1-score are calculated to measure the classification performance.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (3)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (4)$$

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

TABLE IV. OVERALL TEST PERFORMANCE OF THE PROPOSED MODEL

Evaluation Metric	Test Result
Test accuracy	96.94%
Macro precision	95.01%
Macro recall	97.09%
Macro F1-score	95.87%
Weighted precision	97.12%
Weighted recall	96.94%
Weighted F1-score	96.97%
Test loss	0.0953
Number of test images	8,146

The model achieves an overall test accuracy of 96.94% and a macro F1-score of 95.87%. The weighted F1-score is 96.97%, indicating strong overall classification performance across the test samples. The macro F1-score also provides a class-balanced measure by giving equal importance to each of the 38 categories.

D. Class-Wise Classification Analysis

The classification report provides precision, recall, and F1-score for each of the 38 plant disease categories. This analysis helps identify how consistently the model classifies individual plant diseases.

For example, the model achieves an F1-score of 99.82% for Squash Powdery Mildew, 99.69% for Grape Leaf Blight, and 98.88% for Tomato Yellow Leaf Curl Virus. However, comparatively lower F1-scores are observed for Potato Healthy (77.19%), Corn Cercospora Leaf Spot and Gray Leaf Spot (85.23%), and Tomato Early Blight (87.59%).

These differences show that classification performance varies across disease categories. Similar visual symptoms and differences in the number of available samples may contribute to misclassification.

E. Confusion Matrix Analysis

A confusion matrix is used to examine the relationship between actual and predicted disease categories. Correct predictions appear along the diagonal, while values outside the diagonal represent incorrect classifications.

The confusion matrix provides a detailed view of the classification performance across all 38 categories. It helps identify disease categories that are frequently confused with other categories and provides additional information beyond the overall accuracy score.

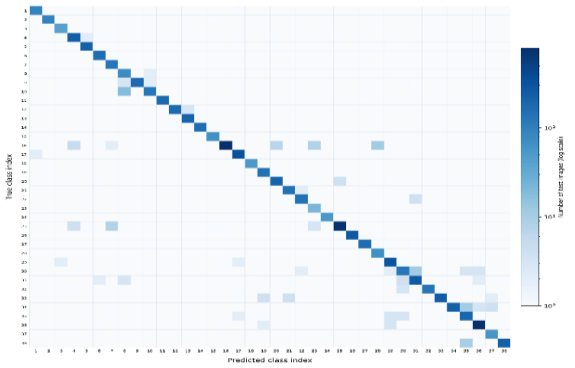


Fig. 4. Confusion matrix of the proposed model on the test dataset.



Fig. 5. Class index key.

F. Result Discussion

The experimental results show that the recorded model achieved 96.94% test accuracy and a macro F1-score of 95.87% on the PlantVillage test dataset. The weighted F1-score of 96.97% indicates strong performance when considering the distribution of test samples across the different categories.

The class-wise results show that some disease categories are classified more accurately than others. This highlights the importance of evaluating the model using multiple metrics rather than relying only on accuracy.

Although the results are promising on the PlantVillage dataset, further testing using images collected under natural field conditions is required to determine how well the model performs in real agricultural environments.

VI. CONCLUSION AND FUTURE WORK

A. Conclusion

This research presents a deep learning-based approach for plant disease classification using computer vision. The proposed system uses the EfficientNet-B0 architecture to classify plant leaf images into 38 categories belonging to 14 crop species. The PlantVillage dataset, containing 54,305 images, is used for model development and evaluation.

The methodology includes image resizing, normalization, data augmentation, and a modified classification layer with dropout. Weighted Cross-Entropy Loss and the AdamW optimizer are used during the training process. The model is evaluated using an independent test dataset containing 8,146 images.

The recorded experimental results show a test accuracy of 96.94%, a macro F1-score of 95.87%, and a weighted F1-score of 96.97%. The confusion matrix and classification report provide further insights into the model's performance across different plant disease categories.

Overall, the results demonstrate the potential of computer vision techniques for automated plant disease classification. However, the evaluation is based on the PlantVillage dataset, and further investigation is required to establish the system's reliability under real agricultural conditions.

B. Future Work

The following improvements can be considered in future development:

1) Real-Field Image Testing: Evaluate the model using plant leaf images captured under natural lighting and different environmental conditions.
2) Model Optimization: Improve the training process by enabling effective feature learning and exploring suitable pretrained weights.
3) Mobile Application Development: Develop a mobile application that allows farmers to capture leaf images and receive disease predictions.
4) Disease Severity Estimation: Extend the system to estimate the severity of detected plant diseases.
5) Additional Crop Support: Include more crop species and disease categories to improve the coverage of the system.

6) Model Comparison: Compare EfficientNet-B0 with other architectures such as MobileNetV3, ResNet, and DenseNet to identify suitable models for accuracy and deployment efficiency.
7) Real-Time Prediction: Explore real-time disease identification using mobile cameras or other image-capturing devices.

REFERENCES

[1] 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. no. 1419, 2016, doi: 10.3389/fpls.2016.01419. https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2016.01419/full#cite

[2] M. Shoaib, B. Shah, S. El-Sappagh, A. Ali, A. Ullah, F. Alenezi, T. Gechev, T. Hussain, and F. Ali, "An advanced deep learning models-based plant disease detection: A review of recent research," Frontiers in Plant Science, vol. 14, Art. no. 1158933, 2023, doi: 10.3389/fpls.2023.1158933. https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2023.1158933/full#cite

[3] M. Xu, S. Yoon, Y. Jeong, and D. S. Park, "Transfer learning for versatile plant disease recognition with limited data," Frontiers in Plant Science, vol. 13, Art. no. 1010981, 2022, doi: 10.3389/fpls.2022.1010981. https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2022.1010981/full#cite

[4] M. H. Saleem, J. Potgieter, and K. M. Arif, "A weight optimization-based transfer learning approach for plant disease detection of New Zealand vegetables," Frontiers in Plant Science, vol. 13, Art. no. 1008079, 2022, doi: 10.3389/fpls.2022.1008079. https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2022.1008079/full#cite

[5] Z. Wu, F. Jiang, and R. Cao, "Research on recognition method of leaf diseases of woody fruit plants based on transfer learning," Scientific Reports, vol. 12, Art. no. 15385, 2022, doi: 10.1038/s41598-022-18337-y. https://www.nature.com/articles/s41598-022-18337-y#citeas

[6] M. Jung, J. S. Song, A.-Y. Shin, B. Choi, S. Go, S.-Y. Kwon, J. Park, S. Park, and Y.-M. Kim, "Construction of deep learning-based disease detection model in plants," Scientific Reports, vol. 13, Art. no. 7331, 2023, doi: 10.1038/s41598-023-34549-2. https://www.nature.com/articles/s41598-023-34549-2#citeas

[7] V. Gautam, N. K. Trivedi, A. Singh, H. G. Mohamed, I. D. Noya, P. Kaur, and N. Goyal, "A transfer learning-based artificial intelligence model for leaf disease assessment," Sustainability, vol. 14, no. 20, Art. no. 13610, 2022, doi: 10.3390/su142013610. https://www.mdpi.com/2071-1050/14/20/13610

[8] Z. Al Sahili and M. Awad, "The power of transfer learning in agricultural applications: AgriNet," Frontiers in Plant Science, vol. 13, Art. no. 992700, 2022, doi: 10.3389/fpls.2022.992700. https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2022.992700/full#cite

[9] J. Arun Pandian, K. Kanchanadevi, N. R. Rajalakshmi, and G. Arulkumaran, "An improved deep residual convolutional neural network for plant leaf disease detection," Computational Intelligence and Neuroscience, vol. 2022, Art. no. 5102290, 2022, doi: 10.1155/2022/5102290. https://onlinelibrary.wiley.com/doi/10.1155/2022/5102290

[10] J. Boulent, S. Foucher, J. Théau, and P.-L. St-Charles, "Convolutional neural networks for the automatic identification of plant diseases," Frontiers in Plant Science, vol. 10, Art. no. 941, 2019, doi: 10.3389/fpls.2019.00941. https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2019.00941/full#cite

[11] A. P. Pimpalkar and J. Rajesh Prasad, "AgroVision: A Field-Acquired Grape Leaf Dataset with a Segmentation-Aware GhostNet Framework," 2026 1st International Conference on Innovation & Growth in Next-Gen Intelligent Technology & Engineering

(IGNITE), Pune, India, 2026, pp. 1-6, doi: 10.1109/IGNITE69592.2026.11706659.

[12] Pimpalkar, A. P. (2026). A Hybrid Lesion Segmentation and Deep Learning Framework for Intelligent Grape Leaf Disease Classification. Journal of Intelligent Decision Making and Information Science, 3(2), 234-261.

[13] Pimpalkar, A. P., & Prasad, J. R. (2025, July). Segmenting Plant Diseases Applying a Novel BIRCH-Based Approach. In 2025 International Conference on Information, Implementation, and Innovation in Technology (I2ITCON) (pp. 1-5). IEEE.