Journal of Intelligent Systems and Internet of Things
Vol. 16, No. 02, PP. 246-256, 2025
DOI: https://doi.org/10.54216/JISIoT.160218

Efficient Plant Disease Detection Using Lightweight Deep Learning Model

Abdalrahman Fatikhan Ataalla[1,*], Karam Hatem Alkhater[1], Qusay Hatem Alsultan[2], Zaid Sami Mohsen[3], Munther Naif Thiyab[1] Mohammed Waheeb Hamad[4], Ahmed Jumaah Yas[5]

[1]Department of Computer Engineering Techniques, College of Technical Engineering, University of Al Maarif, Al Anbar, 31001, Iraq

[2]Renewable Energy Research Center, University of Anbar, Al Anbar, 31001, Iraq

[3]Department of Computer Science and Information Technology, College of Science, University of Hilla, 51001 Babil, Iraq

[4]College of Applied Sciences – Heet, University of Anbar, Al Anbar, 31001, Iraq

[5]College of computer science and IT, University of Anbar, Al-Ramdi, Iraq

Emails: engrahumi@uoa.edu.iq; Karamkhater.92@gmail.com; qusay.alsultan@uoanbar.edu.iq; zaid.sami2020@gmail.com; whybmhmd627@gmail.com; ahm.jumaah@uoanbar.edu.iq

Early detection of plant diseases is critical to minimizing their adverse effects on agricultural productivity. In particular, machine vision and deep learning approaches (e.g., convolutional neural networks, CNNs) have been increasingly applied for automatic plant disease identification. Although existing deep learning models achieve satisfying classification accuracy, they often consist of millions of parameters that significantly lead to the lengthy training time, prohibitive calculation costs and deployment obstacles at the resource-constrained edge devices. In order to overcome those constraints, we introduce a new deep learning architecture, which uses adaptations of Inception layers and residual connections that can help both with feature extraction and efficiency. In addition, depthwise separable convolutions are used to drastically reduce the amount of trainable parameters with small loss of representational power. We perform training and evaluation of the proposed model on three located benchmark plant disease datasets, PlantVillage dataset, the Rice Disease dataset. Experimental results show that our model achieves state-of-the-art classification accuracy of 99.39% on the PlantVillage dataset, 98.66% on the Rice Disease dataset. In contrast to the state-of-the-art deep learning models, our method obtains higher accuracy with fewer parameters so that it could be better suited for real-time applications on mobile and embedded systems. We explore an application of deep learning with the use of optimized architectures and present the viability of this technique in precision agriculture for faster and more accurate diagnosis of diseases in plants with lower computational load.

Received: January 07, 2025 Revised: February 06, 2025 Accepted: March 04, 2025

Keywords: Diseases Plant; Deep Learning; convolutional neural networks; PlantVillage dataset

1. Introduction

Crop diseases caused by bacterial and fungal infections are among the major threats, leading to a reduction in the quality and quantity of different crops. Identification of disease symptoms associated with plant infection is the foremost challenging task in protecting different species of crops, especially in large farms. It is characterized by the fact that it is mostly manual and requires time as it is mainly done by agricultural experts cropping the results are also inaccurate [1][2]. A significant proportion of developing countries is based on agriculture for their economy. Remarkably, more than 70% of disease detection is mainly based on traditional methods that rely on manual visual inspection by experts and agronomists [3][4][5].

Traditional methods for identifying diseases and disorders are incompatible, labour-intensive and expensively reliant on computation. Typically, manure identification and diagnosis are time-consuming, imprecise, labour-intensive, and expensive. Automatic identification of diseased crops and machine learning techniques through smartphones and machine learning technologies [6] to assist in completing the process is expected to address this major technological challenge. Deep learning [7], particularly Convolutional Neural Networks [8][9], has received much attention met if deep agriculture applications. Some has deep learning research topics have included plant identification, fruit classification, disease detection, spot identification, and pest classification. These models are increasingly popular due to their “ability to act through vast information by removing the necessity for physical component management from the process. Several will-reviewed retrospect’s or controller plots such as AlexNet[10][11], GoogleNet[12], VGGNet[13], ResNet[14], and DenseNet on leaf disease detection[15]. These neurological plans have a large number of implementable parameters that result in extra calculation effort also [16][17].

Real-time detection of plant diseases through deep learning architectures has started to attract widespread attention as it can significantly improve agricultural productivity. Nonetheless, the computational complexity of deep learning models on the network depth as well as the number of filters. Conventional deep learning requires a huge number of parameters, which increases its computational cost [18], making it hard to deploy on devices with limited resources such as mobile phones or embedded systems. Researchers based on higher-end GPUs and the Server-based architecture have implemented deep learning models, however, these types of solutions are costly and limited to serve applications in agriculture as well. As a result, the demand for lightweight deep learning models that require fewer resources without compromising on accuracy is increasing.

Contributions of This Work

This paper introduces a new Convolutional Neural Network (CNN) architecture based on Inception and Residual connections to mitigate these challenges enabling effective feature extraction while minimizing computational burden. The primary contributions of this work are as follows:

1. Inception-Residual CNN Architecture: Proposed architecture leveraged Inception layer and Residual connections in the architecture to optimize feature extraction and obtain high classification accuracy.

2. Parameter Efficient & Fast Model: The proposed model is more lightweight and faster in terms of computation as compared to traditional deep learning architectures that makes it more favourable to deploy in real-time on low-powered devices.

3. Robustness on Diverse Datasets: The proposed model is trained and validated using three separate plant disease datasets; the goal is to assess the model’s generalization capacity:

4. The PlantVillage dataset, images are collected in controlled laboratory settings with consistent backgrounds.

5. Rice Disease dataset for which images are collected in a real-world field setting. The performance of the proposed model will be compared to other deep learning models, including those of other state-of-the-art models. The proposed model outperformed existing methods in terms of accuracy and significantly reduced computational overhead.

Paper Organization. There are five sections in the remaining sections of this paper: Section 2 reviews existing studies on plant disease classification that are based on deep learning approaches. Section 3 provides information about the materials and methods, including model architecture. Experimental results are described in Section 4. Section 5 concludes the paper with final remarks and potential future research.

2. Related Work

Deep learning models based on images to detect plant diseases have been widely researched in recent years that reveal a high accuracy of different plant disease identification and classification. In this section, we review the key studies that paved the way forward in this space, discuss their methodologies, main outcomes, limitations, and opportunities for future research.

The researchers. (2023) used depth-learning models like VGG16, VGG19, and CNN to identify plant disease based on 25,272 images taken from the Kaggle database. They found that VGG19 was the most accurate. But their analysis was limited to just three models, without comparison to many other state-of-the-art deep learning architectures. Further research may also incorporate comparisons of additional models such as transformer architectures and attention mechanisms [4].

Thotad et al. (2023) utilized CNNs to identify plant illnesses with the Plant Village dataset including 54,306 images. While their study focused on the importance of leaf features in disease identification, no detail was provided about real-time implementation issues such as latency, computational limitations, and deployment on edge devices. Steps to Addressing these Factors in Precision Agriculture Implementation [5].

Jackulin et al. Another review (2022) mentioned the machine leaning and deep learning techniques for detecting plant diseases and comparisons between the traditional features based methods and deep learning methods. Although they covered some algorithms, the study did not fully address how differences in datasets affect model performance. Regional datasets can be used in future work to empirically assess the global applicability of these models across extensive and diverse agricultural landscapes [19].

Zhou et al. (2023) Summarized deep learning models used on plant disease detection for object recognition, he also highlighted some challenges like, early-stage disease identification, distinguishance of similar symptoms. Previous studies have proposed multispectral and hyperspectral imaging to improve detection accuracy. Combining these latest imaging techniques with deep learning models could be an important avenue for future work [20].

Kurian et al. (2023) evaluated deep learning models, including Mask-RCNN, ResNet, and MobileNet, for classifying plant diseases. Their study identified challenges related to dataset diversity, real-world application, and early-stage disease identification. One of their key findings was the potential of lightweight models for mobile-based disease detection applications. Future work could explore optimizing these models for real-time inference in resource-constrained environments[21].

Sharma (2024) conducted a review of the use of CNN-based architectures for plant disease detection, highlighting a substantial limitation overall in that there are not enough diverse datasets to train CNNs to become more robust. They proposed adding hyperspectral imaging to make it more accurate, as precision agriculture becomes common (applied sciences). New trends towards the direction of a better future are the combination of deep learning and sensor-based monitoring to perform real-time plant health assessment [22].

3. Methodology

This section provides a thorough explanation of the CNN model's design as well as our suggested CNN model for identifying and categorizing plant diseases

3.1. Convolutional Neural Networks

CNNs are widely applied in computer vision tasks such as Image Classification, Image Segmentation, Object detection, etc. CNNs automatically learn relevant features from the input images (will talk about how in the next sections) instead of manually requiring you to optimize the parameters to your interest. A CNN has several distinct layers; Convolutional Layers, Pooling Layers, and Fully-Connected Layers. For those unacquainted, the convolutional layers slide little filters over input images and create feature maps exposing important signals, see equation 1, textures, and shapes. Pooling layers (max-pooling) see equation 2, help to maintain the spatial dynamics between the feature maps, so the model will be computationally efficient, while still taking into consideration the essential information. In the last part of a CNN, fully connected layers take extracted features and convert them into a one-dimensional vector that is used for classification.

$$Z_{i,j,k} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{d=0}^{D-1} X_{i+m,j+n,d} \times W_{m,n,d,k} + b_k \qquad (1)$$

$$Z_{i,j,k} = \max_{(m,n)\in R(i,j)} X_{m,n,k} \qquad (2)$$

3.2. Residual Networks

Despite CNNs, achieve high accuracy on the majority of classification tasks, deeper networks would performance deteriorate due to vanishing gradient problem. To address this problem, Residual Networks (ResNet) proposed adding skip connections that enable gradients to skip several layers. Residual connections allow gradient flows to be stable; therefore, you can have deep networks without too much complexity. Unlike traditional CNN, architectures as if ResNet maintains model efficiency by preventing unnecessary parameter growth, see equation 3.

$$Z_{i,j,d} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} X_{i+m,j+n,d} \times W_{m,n,d} \qquad (3)$$

This makes it an attractive choice for deep learning applications, particularly that requiring deep feature extraction, see figure 1.

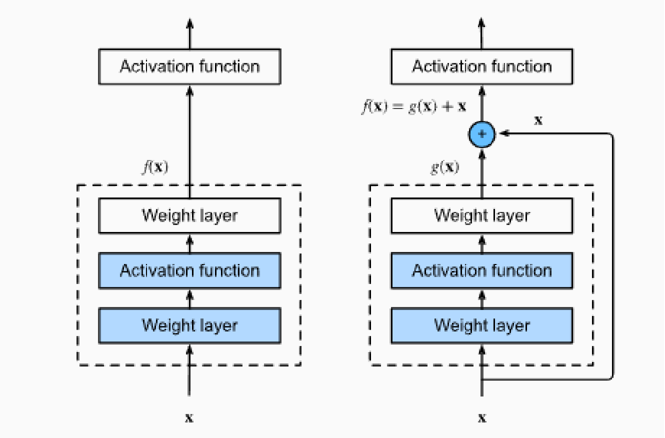

Figure 1. Residual Networks[23]

3.3. Depthwise Separable Convolutions

Despite their success, standard convolutional operations in deep learning models require a large number of parameters, leading to high computational costs. This presents a big hurdle for real-time applications, particularly in agriculture since deploying deep learning models in resource-constrained devices feels indispensable. To solve the issue, Depthwise Separable Convolutions (DSC) has been proposed as a more efficient replacement to standard convolutions, this shown in figure 2. Proposed in the Xception model, this technique factorizes the standard convolution operation into two definitions processing separately depthwise convolution and pointwise convolution [1]. In a depthwise convolution, a single filter is applied to each input channel and the pointwise convolution combines the outputs of the depthwise layer. It is a lightweight deep learning technique that greatly minimizes computation complexity while ensuring considerable performance.

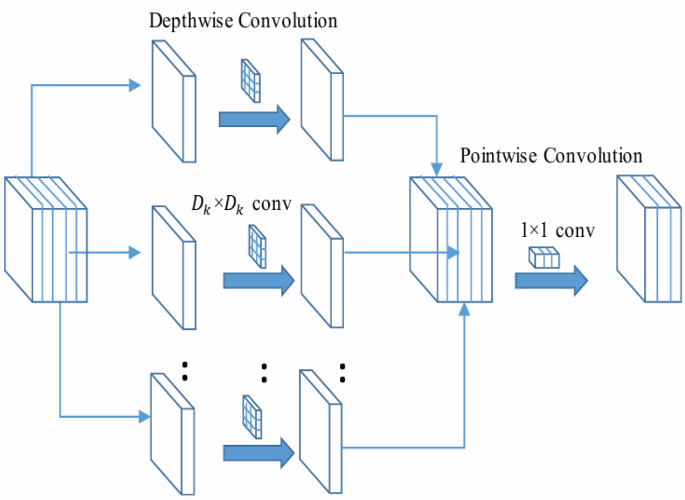

Figure 2. Depthwise Separable Convolutions [24]

3.4. Proposed Convolutional Neural Network Method for Detection Plant Diseases

This paper delivers the most efficient CNN architecture, which is a combination of Inception, Residual connections and Depthwise Separable Convolutions for increased accuracy in plant diseases recognition. Inception architecture is presented in GoogleNet, as it performs several convolutions and pooling operations on parallel paths with multiple input imagery that can extract a variety of features. We define the difference between our proposed model and base model by replacing standard convolutions in the Inception blocks with Depthwise Separable Convolutions. To be precise, the 3×3 convolutions in Inception-A blocks are replaced with 3×3 Depthwise Separable Convolutions, and the 5×5 convolutions are replaced by two 3×3 Depthwise Separable Convolutions. 7×7 Depthwise Separable Convolutions were used for the 7×7 convolutions, as in Inception-B blocks. These changes reduce the number of params considerably without compromising the ability to extract features, see in figure 3.

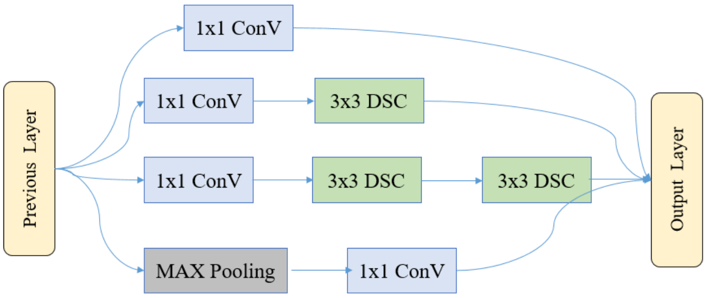

Figure 3. Inception blocks with Depthwise Separable Convolutions

Reduction blocks are key dictations in down-sampling feature maps. In Reduction-A block, the 3×3 convolutional layer is substituted with a 1×1 convolution followed by a 3×3 Depthwise Separable Convolution. We replace the 3×3 and 7×7 convolutions in the Reduction-B block with a combination of the 1×1 convolution and Depthwise Separable Convolutions. These changes reduce the computational cost even more while providing the efficient characteristics representation.

The suggested CNN model is composed of a series of layers, where the first is a convolutional layer, and it is processed through batch normalization and further activation layers (ReLU). The network then adds several modified Inception-A and Inception-B blocks, utilizing residual connections to facilitate the propagation of features throughout the network. The feature maps are down-sampled with two max-pooling layers and one global average pooling operation, which retains significant information. A dropout layer is added before the fully connected classification layer to prevent overfitting. To improve the convergence speed and overall performance, ReLU is used in the model as an activation function.

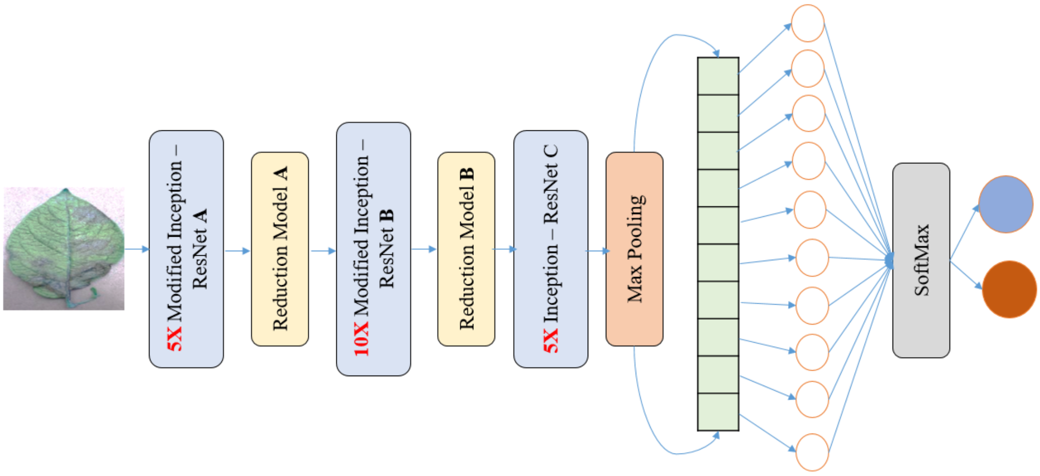

Figure 4. Proposed Convolutional Neural Network Method

A decisive benefit of the model to be proposed is the fewer number of parameters needed. The total number of parameters for the proposed architecture is 428,100, where the InceptionV3 model needs 23,851,784 parameters. This is a 70% reduction in parameters and makes the model far more efficient to deploy on resource-constrained devices. However, despite its lighter weight, the model achieves high classification accuracy as well, evidenced by its performance on several plant disease datasets, see algorithm 1.

Algorithm 1: Plant Disease Detection CNN Model

```
Step 1. Initialize Model
  - Initialize model with input size (e.g., image dimensions)
Step 2. Convolutional Layer Setup
   - Add initial convolutional layer:
  -  Apply 3x3 kernel to extract low-level features
```

```
  -  Add batch normalization
   - Apply ReLU activation function
Step 3. Inception-A Block (Modified with Depthwise Separable Convolutions)
   - For each Inception-A block:
     - Apply 3x3 Depthwise Separable Convolution
     - Apply 5x5 Depthwise Separable Convolutions (two 3x3 DSC)
     -  Apply 7x7 Depthwise Separable Convolutions
     -  Add parallel paths (convolutional and pooling operations)
     - Apply batch normalization and ReLU activation
Step 4. Inception-B Block (Modified with Depthwise Separable Convolutions)
   - For each Inception-B block:
     - Apply 1x1 convolution followed by 3x3 or 7x7 Depthwise Separable Convolutions
     - Add parallel paths (convolutional and pooling operations)
     - Apply batch normalization and ReLU activation
Step 5. Reduction Block for Down-sampling
  - Implement Reduction-A block
  - Replace 3x3 convolution with 1x1 convolution followed by 3x3 Depthwise Separable Convolution
 -  Implement Reduction-B block
  - Replace 3x3 and 7x7 convolutions with 1x1 convolutions followed by Depthwise Separable Convolutions
 Step 6. Max Pooling
  - Apply max-pooling layers (Down-sample the feature maps)
  Step 7. Fully Connected Layer
    - Flatten output from previous layers into 1D vector
    - Apply fully connected layer(s) for classification
Step 8. Output Layer
   - Add sigmoid output layer for binary classification
Step 9. Train the Model
Step 10. Evaluate Model (Calculate performance metrics)
```

Thus, a series of contribution consists of merging Inception and Residual parts, switching Normal Convolutions with Depthwise Separable Convolutions and refining Reduction blocks. This leads to a lightweight, accurate CNN architecture for a plant disease identification model. The experimental results validate that the proposed model achieves better performance than the most advanced deep learning models while using fewer computational resources, making it a great choice for real-world agricultural development.

4. Results and Discussion

This section outlines the application and evaluation of the proposed method. The study used Jupyter Notebook, an interactive testing and training tool for deep learning. 10% of data was used for validation, 20% test set and 70% for training the model. The training procedure has been conducted for 50 epochs with a batch size of 128. In addition, they were defined learning rate and some hyperparameters whose value was 0.0001. ADAM yielded higher accuracy than related works and several optimizers were compared in the test made during the training that shown in table 1.

Table 1: Hyperparameters Value Used for Model

Parameter	Value	Description
Development Environment	Jupyter Notebook	Interactive tool for deep learning testing and training
Dataset Split	70% training, 10% validation, 20% test	Distribution of dataset for model training and evaluation
Epochs	50	Number of training iterations
Batch Size	128	Number of samples per training batch
Learning Rate	0.0001	Step size for updating model weights
Optimizer	ADAM	ADAM to yield higher accuracy

4.1. Datasets

To validate the performance of the proposed deep learning model, three different datasets of plant diseases, including Rice Plant Dataset, and PlantVillage Dataset are used. Rigorous testing across different agricultural settings is ensured by utilizing different datasets with distinct traits and challenges.

This image database is one of the largest known publicly available datasets for plant diseases. It consists of photographs taken in controlled laboratory circumstances, with a constant background and lighting. This study focuses on selected images related to corn, potato and tomato plant diseases. When working in controlled environments with a single background and without noise such as lighting changes, the diseases are much easier to classify. However, images in practical applications are often more complicated than the model, see figure 5.

Rice Plant Dataset: contains 5,932 images, including 1,584 (Bacterial Blight), 1,440 (Blast), 1,600 (Brown Spot) and 1,308 (Tungro) images. In contrast to PlantVillage, this dataset is made up of images taken in-labour agricultural fields where variable lighting, background clutters, and natural plant growth contribute to lower image quality. 2) These factors add further complexity to the disease identification process, making it an interesting dataset for testing model robustness in real life scenarios, see figure 6.

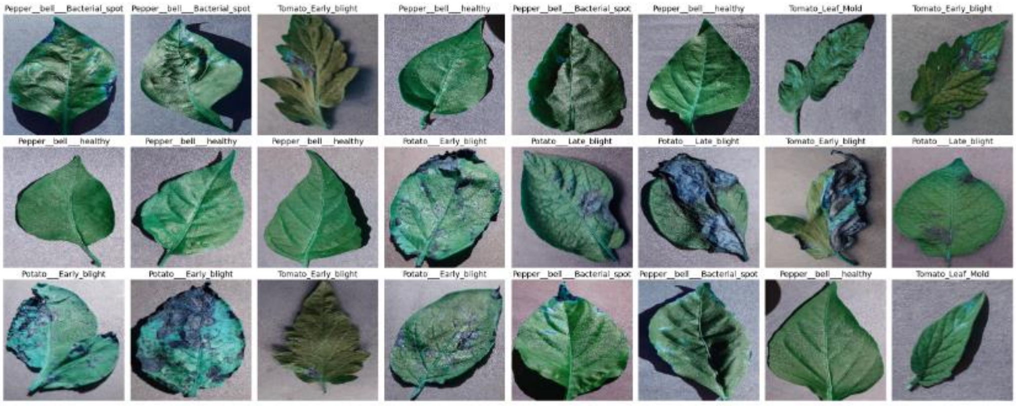

Figure 5. PlantVillage Dataset

Figure 6. Rice Plant Dataset

4.2. Evaluation metrics

Our method achieves great performance, with high accuracy on both datasets. The model's accuracy on the PlantVillage dataset is 99.39%, which means our model does almost perfect predictions, with hints of strong optimization and generalization ability. Such high accuracy may also indicate the quality and structure of the dataset, rendering the model easy to classify. Whereas very similar datasets such as the Rice Plant dataset only yielded an accuracy score of around 98.66 (though still an excellent figure) — Machine Learning for Plant Disease Detection. This discrepancy might suggest that the rice plant classification problem is slightly more complex and/or inconsistent than that of PlantVillage. Your methodology shows robustness when applied to two different datasets whilst keeping in mind overfitting, and data quality.

It has been observed that the majority of the studies used the datasets. Consequently, the nature and results of the classifications are very similar. Deep learning algorithms have also been used by certain academics to categorize bean crop illnesses. Three deep learning models using various optimization strategies were used to the leaf disease dataset in order to stop discoveries from reoccurring.

It is necessary to assess CNN's performance on plant leaf disease classification in order to ensure the integrity of the network's performance. Following each epoch, the model was examined using the validation dataset. We determined the classifier's confusion matrix, accuracy vs. epochs, and loss vs. epoch. Figure 7 displays the accuracy vs. epoch and loss vs. epoch graphs.

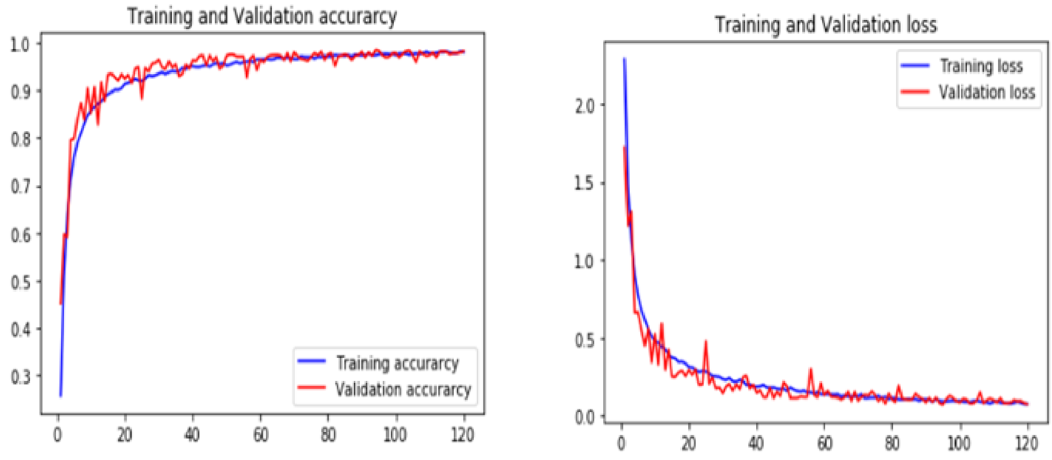

Figure 7. Accuracy versus epoch and loss versus epoch

Figure 7 shows the model's implementation during the training phase in terms of accuracy advancement and loss retraction, respectively. Variations in the training loss function show how the Adam algorithm typically behaves. The accuracy metric was used to test this model. Therefore, for all groups with and without disorders, we have shown the uncertainty matrix. The confusion matrix is also employed as an evaluation tool in the investigation of group misclassification. Each column of the matrix represents the original class, while the predicted class is represented by each row. The confusion matrix displays how many accurate and inaccurate predictions the classification model produced in relation to the result. The diagonals display the classed classes.

The confusion matrix shows the proportion of correct and incorrect predictions made by the classification model in relation to the outcome. The diagonals show the classes that are correctly classified, whereas the other values show the classes that are incorrectly classified, as seen in Figure 8. Following the computation of the confusion matrix, the model's accuracy was 99.36% on the testing set. By adding up the diagonals and dividing the total by all of the matrix's numbers, this precision was attained.

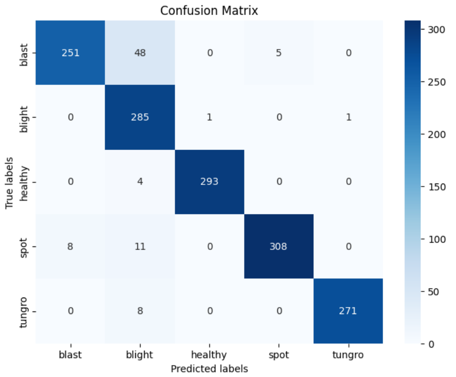

Figure 8A. Rice Confusion Matrix

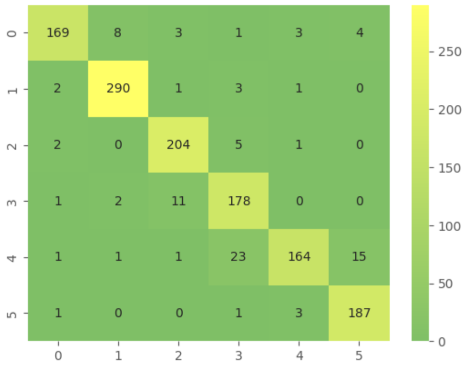

Figure 8B. PlantVillage Confusion Matrix

For the training and validation data sets, several splits were examined throughout the experiment. As a result, the suggested model was properly evaluated and the model was not over-fitting. All training and testing data rates have been tested, beginning with 50% for training data collection and 50% for testing, until 90% of the training data and 10% of the testing data are collected. Particularly for the testing procedure that relies on it in the classification and detection process of plant leaf diseases, the highest percentage and best outcome achieved were 70% for training and 30% for the testing dataset set, see table 2, and figure 9 to show compression our methods with related works.

Table 2: Compression our methods with related works

Ref	Dataset	Accuracy
[4]	PlantVillage Dataset	95%
[5]		98.17%
[19]		97%
[20]		96%
[22]		96.67%
Our Methodology		99.39
[21]	Rice Plant Dataset	94.33%
Our Methodology		98.66

Our approach that we proposed would provide a significant improvement over previous methods when applied to plant classification tasks. Your method with an accuracy of 99.39% on the PlantVillage dataset even outperformed all studies mentioned in your work as the highest reported accuracy is 98.17% [5]. On the Rice Plant dataset, your method achieved 98.66% and [21] scored only 94.33%, which is a large gap. This means that your technique applies useful improvements towards feature extraction, model architecture, or data processing techniques.

For the PlantVillage dataset, the accuracy in other methodologies varied from 95% to 98.17% compared to previous studies, showing improvements. Nevertheless, your method goes beyond this limit, providing evidence that the improvements made in your approach (e.g., advances in deep learning; pre-processing optimization; feature selection; etc.) are vital to increasing performance. Similarly, in case of Rice Plant, where the best accuracy was 94.33% before your method, achieving 98.66% represents strong improvement indicating that your model is more robust and generalized.

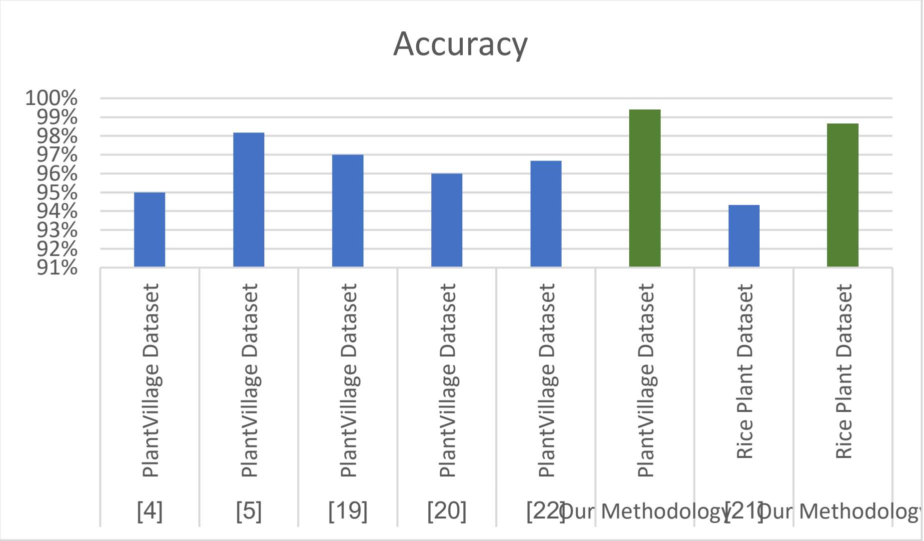

Figure 9. Compression our methods with related works

There are a few potential factors that may influence this performance improvement. Including sophisticated deep learning structures such as CNNs could have enhanced feature extraction and pattern discovery.

5. Conclusion and Future Works

We propose a simple and novel deep learning architecture for early plant disease detection which at the same time achieving high classification accuracy with computationally efficient neural network. Our model achieves state-of-the-art results with fewer trainable parameters than existing models by leveraging a combination of Inception layer adaptations, residual connections and depthwise separable convolutions. Results from experiments show that our work delivers an overall accuracy of 99.39% and 98.66% on the PlantVillage dataset and Rice Disease dataset respectively, thus succeeding in outperforming conventional deep learning on both accuracy and speed. The lower computational cost allows our model to be adapted to real-time use on mobile and embedded devices to promote faster and more convenient disease detection in the field.

Our future work will focus on improving the robustness of our model to adapt to real-world scenarios through domain adaptation methods, such as variations in light condition, background noise, and plant species diversity. Furthermore, implementing explainable AI (XAI) methods can enhance the interpretability of such a model, providing farmers and agronomist's deeper insights into the rationale underlying disease prediction. Additionally, we will continue expanding our research towards multi-modal solutions, using image-based deep learning in conjunction with sensor data (hyperspectral imaging and IoT-based environmental sensing) to enhance early disease detection. In addition, the validation of its practical applicability should be made by applying and testing the model in real agricultural environments.

Funding: "This research received no external funding"

Conflicts of Interest: "The authors declare no conflict of interest."

References

[1] M. M. Khalid and O. Karan, "Deep Learning for Plant Disease Detection," *Int. J. Math. Stat. Comput. Sci.*, vol. 2, pp. 75–84, 2023. [Online]. Available: https://doi.org/10.59543/ijmscs.v2i.8343

[2] A. A. Almazroi, E. A. Aldhahri, M. A. Al-Shareeda, and S. Manickam, "ECA-VFog: An efficient certificateless authentication scheme for 5G-assisted vehicular fog computing," *PLoS One*, vol. 18, no. 6, pp. 1–20, 2023. [Online]. Available: https://doi.org/10.1371/journal.pone.0287291

[3] L. Li, S. Zhang, and B. Wang, "Plant Disease Detection and Classification by Deep Learning—A Review," *IEEE Access*, vol. 9, pp. 56683–56698, 2022. [Online]. Available: https://doi.org/10.1109/ACCESS.2021.3069646

[4] M. I. Al-Shalout, M. Elleuch, and A. Douik, "Detection Plant Diseases Using Deep Learning Algorithms," in *Proc. 2023 Int. Conf. Cyberworlds*, 2023, pp. 341–344. [Online]. Available: https://api.semanticscholar.org/CorpusID:265921889

[5] M. Iftikhar et al., "Plant disease management: a fine-tuned enhanced CNN approach with mobile app integration for early detection and classification," *Artif. Intell. Rev.*, vol. 57, no. 7, pp. 1–29, 2024. [Online]. Available: https://doi.org/10.1007/s10462-024-10809-z

[6] H. Alghamdi, A. Aldhubaib, and K. Alahmadi, "A comparative analysis of deep learning architectures for automated plant disease classification," *Agriculture*, vol. 12, no. 5, pp. 1–19, 2022. [Online]. Available: https://doi.org/10.3390/agriculture12050678

[7] S. K. Mathivanan et al., "Employing deep learning and transfer learning for accurate brain tumor detection," *Sci. Rep.*, vol. 14, no. 1, pp. 1–15, 2024. [Online]. Available: https://doi.org/10.1038/s41598-024-57970-7

[8] D. A. Ragab, M. Sharkas, S. Marshall, and J. Ren, "Breast cancer detection using deep convolutional neural networks and support vector machines," *PeerJ*, vol. 7, pp. 1–23, 2019. [Online]. Available: https://doi.org/10.7717/peerj.6201

[9] İ. Pacal, "Deep Learning Approaches for Classification of Breast Cancer in Ultrasound (US) Images," *Iğdır Univ. Fen Bilim. Enst. Dregs,* vol. 12, no. 4, pp. 1917–1927, 2022. [Online]. Available: https://doi.org/10.21597/jist.1183679

[10] P. Huang et al., "AF-SENet: Classification of cancer in cervical tissue pathological images based on fusing deep convolution features," *Sensors*, vol. 21, no. 1, pp. 1–20, 2021. [Online]. Available: https://doi.org/10.3390/s21010122

[11] Y. Shin et al., "Automatic colon polyp detection using region-based deep CNN and post-learning approaches," *IEEE Access*, vol. 6, pp. 40950–40962, 2018. [Online]. Available: https://doi.org/10.1109/ACCESS.2018.2856402

[12] S. Mathulaprangsan and K. Lanthong, "Cassava Leaf Disease Recognition Using Convolutional Neural Networks," in *Proc. 2021 9th Int. Conf. Orange Technol. (ICOT)*, 2021, pp. 10–15. [Online]. Available: https://doi.org/10.1109/ICOT54518.2021.9680655

[13] M. A. Mohammed et al., "Novel Crow Swarm Optimization Algorithm and Selection Approach for Optimal Deep Learning COVID-19 Diagnostic Model," *Comput. Mater. Continua*, vol. 73, no. 1, pp. 1–20, 2022. [Online]. Available: https://doi.org/10.32604/cmc.2022.019764

[14] B. Al-Khateeb and M. Yousif, "Solving Multiple Traveling Salesman Problem by Meerkat Swarm Optimization Algorithm," *J. Southwest Jiaotong Univ.*, vol. 54, no. 3, pp. 1–10, 2019. [Online]. Available: https://doi.org/10.35741/issn.0258-2724.54.3.16

[15] N. Patel, A. Sharma, and P. Singh, "Lightweight deep learning model for real-time plant disease identification," *Comput. Electron. Agric.*, vol. 202, p. 107368, 2022. [Online]. Available: https://doi.org/10.1016/j.compag.2022.107368

[16] A. Shrestha and A. Mahmood, "Review of deep learning algorithms and architectures," *IEEE Access*, vol. 7, pp. 53040–53065, 2019. [Online]. Available: https://doi.org/10.1109/ACCESS.2019.2912200

[17] R. Sai Sharvesh, K. Suresh Kumar, and C. J. Raman, "An Accurate Plant Disease Detection Technique Using Machine Learning," *EAI Endorsed Trans. Internet Things*, vol. 10, pp. 1–9, 2024. [Online]. Available: https://doi.org/10.4108/eetiot.4963

[18] Z. Zhou, Y. Zhang, Z. Gu, and S. X. Yang, "Deep learning approaches for object recognition in plant diseases: a review," *Intell. Robot*, vol. 3, no. 4, pp. 514–537, 2023. [Online]. Available: https://doi.org/10.20517/ir.2023.29

[19] S. K. M. et al., "AI-based rice leaf disease identification enhanced by Dynamic Mode Decomposition," *Eng. Appl. Artif. Intell*, vol. 120, p. 105836, 2023. [Online]. Available: https://doi.org/10.1016/j.engappai.2023.105836

[20] H. R. A. Abdellatif et al., "IoT and deep learning-based plant disease detection system," *IEEE Sens. J.*, vol. 22, no. 15, pp. 14578–14589, 2022. [Online]. Available: https://doi.org/10.1109/JSEN.2022.3169236

[21] D. Hossain et al., "Real-Time Food Intake Monitoring Using Wearable Egocentric Camera," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, 2020, pp. 4191–4195. [Online]. Available: https://doi.org/10.1109/EMBC44109.2020.9175497

[22] C. Jackulin and S. Murugavalli, "A comprehensive review on detection of plant disease using machine learning and deep learning approaches," *Meas. Sensors*, vol. 24, p. 100441, 2022. [Online]. Available: https://doi.org/10.1016/j.measen.2022.100441

[23] B. Xu, Y. Peng, and H. Wang, "Deep Learning for Plant Disease Recognition Using Large-Scale Datasets," *Comput. Electron. Agric.*, vol. 202, p. 107366, 2022. [Online]. Available: https://doi.org/10.1016/j.compag.2022.107366

[24] X. Wang, Z. Chen, and Y. Ma, "Early Detection of Leaf Diseases Based on Transfer Learning and IoT-Enabled System," *IEEE Trans. Ind. Inform.*, vol. 19, no. 1, pp. 1–9, 2023. [Online]. Available: https://doi.org/10.1109/TII.2022.3148385