



Optimized CNN-Based Transfer Learning Approach with ReduceLROnPlateau for Early Potato Leaf Disease Classification

Andi Sunyoto^{1*}, Yoga Pristyanto², Anggit Ferdita Nugraha³

^{1,2,3}Faculty of Computer Science, Universitas Amikom Yogyakarta, Yogyakarta, Indonesia

*yoga.pristyanto@amikom.ac.id

Abstract

Potato leaf disease classification is important for supporting early disease detection and improving crop management. This study proposes an optimized CNN-based transfer learning approach for classifying potato leaf images into three classes: Early Blight, Healthy, and Late Blight. Six pre-trained architectures—InceptionV3, Xception, DenseNet121, EfficientNetB0, ResNet50, and MobileNet were evaluated using ImageNet weights. Each model was modified with a custom classification head consisting of Global Average Pooling, a dense layer with 1,024 neurons, dropout, and a Softmax output layer. The models were evaluated using 5-fold stratified cross-validation. A frozen baseline configuration was first established, followed by fine-tuning configurations that varied learning rate, dropout rate, number of unfrozen layers, and the use of ReduceLROnPlateau. Data augmentation was applied only to the training subset in each fold. Performance was assessed using accuracy, AUC, validation loss, confusion matrix, ROC curves, training time, inference time, and GFLOPs. The results showed that optimized fine-tuning improved the average accuracy from 0.9876 to 0.9975 and reduced the average validation loss from 0.0316 to 0.0025. The Xception with the HP3 configuration achieved the best performance, with a mean accuracy of 0.999444 and an AUC of 0.999997. However, EfficientNetB0 achieved a high accuracy of 0.998333 with only 0.80 GFLOPs per image, making it more suitable for efficiency-oriented deployment. These findings demonstrate that fine-tuned transfer learning combined with adaptive learning rate scheduling is effective for potato leaf disease classification. Because the optimized configurations differ from the frozen baseline in the number of unfrozen layers, the dropout rate, the learning rate, and the use of ReduceLROnPlateau simultaneously, the reported gain reflects the combined effect of these components rather than the isolated contribution of ReduceLROnPlateau. The results were obtained on PlantVillage images acquired under controlled conditions and should therefore be regarded as an upper reference rather than an estimate of performance under field condition.

Keywords: early blight; late blight; potato leaf disease; reducelonplateau; transfer learning

How to Cite: A. Sunyoto, Y. Pristyanto, and A. F. Nugraha, "Optimized CNN-Based Transfer Learning Approach with ReduceLROnPlateau for Early Potato Leaf Disease Classification", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 10, no. 5, pp. 1265–1276, Oct. 2026, doi: [10.29207/resti.v10i5.7533](https://doi.org/10.29207/resti.v10i5.7533).

Permalink/DOI: <https://doi.org/10.29207/resti.v10i5.7533>

Received: January 28, 2026

Accepted: September 5, 2026

Available Online: October 2, 2026

*This is an open-access article under the [CC BY 4.0 License](https://creativecommons.org/licenses/by/4.0/)
Published by [Ikatan Ahli Informatika Indonesia](https://www.iaii.or.id/)*

1. Introduction

Modern agricultural farm technology continues to evolve rapidly, and agricultural production is highly dependent on the technologies applied in farming practices. Potato farming is one of the most important agricultural activities, as potatoes provide a rich source of carbohydrates, proteins, minerals, and dietary fiber compared to many cereal crops. However, potato productivity can decrease significantly due to diseases that attack potato plants. These diseases commonly affect leaves and fruits, but this study focuses specifically on potato leaf diseases. In traditional agricultural practices, farmers often rely on plant

pathologists to accurately detect and diagnose plant diseases [1]. In our daily lives, it is also a good source of carbohydrates.

Potatoes are an important food crop cultivated and consumed worldwide [2]. However, the combination of population growth, urbanization, and limited agricultural land creates increasing pressure on food production. Crop diseases are one of the major factors affecting agricultural productivity and can cause considerable economic losses. Potato plants are vulnerable to various diseases, particularly those affecting their leaves, with early blight and late blight being among the most significant causes of yield

reduction. Disease identification is commonly performed through visual examination, but relying on human expertise has practical limitations, including high labor requirements, time consumption, and the limited availability of plant disease specialists in some regions [3].

Timely and reliable detection of potato diseases is essential for limiting crop losses and reducing the risk of disease transmission. Conventional inspection based on visual assessment is labor-intensive, dependent on expert knowledge, and may be difficult to apply consistently over large cultivation areas. Delayed diagnosis can further allow infections to spread before appropriate control measures are implemented. Consequently, automated computer-based systems have emerged as useful tools for supporting farmers in making faster and more informed decisions [4]. In particular, artificial intelligence (AI) can facilitate early identification of leaf diseases and support more effective crop protection.

The application of AI, machine learning (ML), and computer vision (CV) has led to numerous image-based approaches for plant disease classification. Earlier methods commonly relied on handcrafted feature extraction followed by conventional classifiers. For instance, [5] combined Kuan filtering and Hough transformation to obtain color, shape, and texture features before applying a boosting-based classifier to PlantVillage images. Another approach [6] derived texture information using the Gray Level Co-Occurrence Matrix (GLCM) after grayscale conversion and subsequently performed classification with Support Vector Machine (SVM). Random Forest has also been applied as an alternative classifier for image-derived plant disease features [7].

Deep learning (DL) has become increasingly important in agricultural research, particularly for tasks involving image-based analysis [8]. Its integration with modern computational technologies has supported the development of intelligent agricultural systems capable of addressing complex tasks such as feature learning, representation transformation, pattern recognition, and image classification [9]. Among these techniques, Convolutional Neural Networks (CNNs) are widely used for plant disease identification because they can learn relevant visual features directly from raw images without relying entirely on handcrafted descriptors. Their application has been reported in various agricultural tasks, including fruit counting, blossom detection, and leaf-related analysis estimation [10], [11]. Studies comparing different CNN architectures across multiple datasets have also reported improvements over conventional approaches, although model performance remains dependent on both network architecture and dataset characteristics. CNN-based methods have further been explored for identifying specific plant diseases, including mosaic disease and cassava brown streak virus [12].

Recent advances in artificial intelligence have further accelerated the development of smart agriculture by integrating essential AI technologies, including mobile devices, autonomous agents, robotics, sensing systems, computer vision, and environmental interaction. The integration of heterogeneous data sources and collaborative systems has led to the adoption of semantic technologies to support agricultural decision-making [13], [14]. Furthermore, the need for reliable predictions and effective planning in agricultural management has emphasized the importance of interdisciplinary collaboration between agricultural experts and artificial intelligence researchers [13].

Although substantial advances have been made, several limitations reported in earlier studies still require attention. The PlantVillage dataset, for example, is widely adopted as a benchmark for plant disease classification, but its potato subset contains only 152 healthy leaf images, compared with 1,000 images each for early blight and late blight [15]. This distribution creates a considerable class imbalance that may bias model learning and affect classification reliability. Another concern is overfitting, which is particularly relevant when CNN models are trained with limited or imbalanced data. In such cases, a model may become overly adapted to the training samples, producing high training accuracy while failing to generalize effectively to previously unseen images [16].

To address the challenges posed by small and imbalanced datasets, several techniques have been proposed, with data augmentation among the most widely adopted approaches. Data augmentation increases the amount of training data by generating new samples through various image transformations, thereby improving model generalization and reducing overfitting [17]. This technique is particularly effective for increasing the minority class sample size in imbalanced datasets.

This study addresses potato leaf blight classification, with particular emphasis on the challenges posed by limited and imbalanced datasets, as also reported in previous studies [4], [6], [7], [18]. To mitigate these limitations, data augmentation is employed to increase the diversity of training samples, while deep learning models such as ResNet50 [24], VGG16 and VGG19 [1], [19], and ANN [20] have been explored to improve disease classification and reduce the risk of overfitting. This study aims to develop a system that assists farmers or agricultural managers in identifying potato leaf diseases using potato leaf image data. Unlike previous CNN-based studies that focused solely on improving accuracy, this study proposes an enhanced transfer learning framework that combines customized data augmentation and adaptive learning rate scheduling to address class imbalance and overfitting in small agricultural datasets. Additionally, the model's efficiency is analyzed for potential deployment in mobile and edge-based agricultural systems.

This study presents an optimized transfer learning framework for potato leaf disease classification by integrating fine-tuning strategies, hyperparameter optimization, and comprehensive performance evaluation. Unlike previous studies that primarily emphasize classification accuracy, this work systematically investigates the effects of transfer learning optimization on predictive performance, computational complexity, and model generalization using a unified experimental protocol. The main contributions of this study are summarized as follows:

An optimized transfer learning framework is proposed by integrating partial fine-tuning, grid-search hyperparameter tuning, and adaptive learning rate using ReduceLROnPlateau. The framework systematically evaluates the effects of learning rate, dropout rate, and the number of unfrozen layers on classification performance. Because these components were varied jointly rather than ablated individually, their effects are reported as a combined contribution.

A comprehensive comparative evaluation is conducted on six pre-trained CNN architectures (MobileNet, EfficientNetB0, DenseNet121, ResNet50, InceptionV3, and Xception) using 5-fold stratified cross-validation. Model performance is assessed using multiple evaluation metrics, including accuracy, AUC, confusion matrix, ROC curve, GFLOPs, training time, inference time, and statistical significance testing.

The proposed optimization framework achieves superior classification performance on the PlantVillage potato leaf disease dataset. Among the evaluated models, the fine-tuned Xception architecture with ReduceLROnPlateau achieved the highest mean accuracy of 99.94% and a mean AUC of 99.99%, outperforming several previously reported methods for three-class potato leaf disease classification. This result was obtained on the controlled-acquisition PlantVillage dataset, and the comparison with previously reported methods is indicative rather than strictly equivalent, since the referenced studies differ in dataset source, number of target classes, preprocessing, and evaluation protocol.

An accuracy complexity trade-off analysis is presented to support practical model selection. While Xception achieved the highest predictive performance, EfficientNetB0 demonstrated the lowest computational complexity with competitive classification accuracy, making it a suitable candidate for deployment on resource-constrained devices such as mobile and edge-computing platforms. This indication is derived from theoretical computational complexity; on-device latency was not measured in this study.

The remainder of this paper is structured into four sections. Section 1 introduces the research background, motivation, problem formulation, objectives, and contributions. Section 2 details the dataset, the proposed CNN-based transfer learning framework, hyperparameter tuning process, and experimental

procedures. Section 3 reports the experimental findings, model performance, comparative results, and corresponding discussion. Finally, Section 4 summarizes the main findings and contributions and suggests potential directions for future research.

2. Methods

Figure 1 summarizes the experimental procedure adopted in this study. The workflow starts with the potato leaf image dataset and proceeds to image preprocessing before the data are used for transfer learning. Six pre-trained CNN architectures—InceptionV3, Xception, DenseNet121, EfficientNetB0, ResNet50, and MobileNet—were investigated as base models. Each architecture was tested under five hyperparameter settings, consisting of a frozen baseline and four fine-tuning schemes that varied in learning rate, dropout, and the number of trainable layers. The experiments employed stratified 5-fold cross-validation to obtain robust performance estimates while maintaining the class distribution across folds. Model performance was subsequently evaluated using accuracy, AUC, confusion matrices, ROC curves, GFLOPs, training time, and statistical tests.

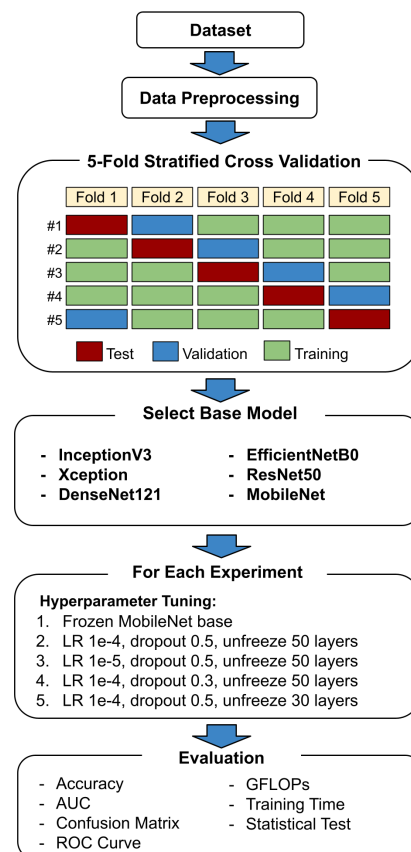


Figure 1. Overall experimental workflow for the proposed CNN-based transfer learning approach with hyperparameter tuning and 5-fold stratified cross-validation.

2.1 Dataset

This study used the potato leaf dataset from PlantVillage [15], a non-profit project developed in

collaboration with Penn State University (US) and EPFL (Switzerland). The dataset includes color JPG files with a resolution of 256x256. The total number of all 3,600 images, along with their distribution, is shown in Table 1.

Table 1. Summary of PlantVillage Potato Leaf Disease Dataset

Label class	Sample	Ratio
Early Blight	1,200	33.33%
Late Blight	1,200	33.33%
Healthy	1,200	33.33%
Total samples	3,600	100%

Table 2. Three illustrations of the class's three most typical potato leaves taken from the PlantVillage Dataset (Healthy, early blight, and late blight).

Class	Sample 1	Sample 2	Sample 3
Healthy potato leaves			
Early blight potato leaf			
Late blight potato leaf			

As presented in Table 2, the symptoms of Early Blight are characterized by yellowish to bright yellow-green halos surrounding or overlapping the affected spots. As the lesions enlarge, they may become more extensive and cause widespread yellowing, followed by premature leaf death. In contrast, Late Blight initially appears as small, irregularly shaped lesions ranging from light green to dark green. Under cool and humid conditions, these lesions can develop rapidly into large brown to purple-black areas. Severe infection may result in the destruction of entire leaflets and can progress through the petioles to the stems, potentially causing the above-ground parts of the plant to die. Healthy leaves, on the other hand, exhibit a fresh and normal appearance without visible dark lesions or disease-related spots.

2.2 Data Preprocessing

Image resizing is an essential preprocessing operation in computer vision, as it standardizes the spatial dimensions of input images before they are provided to a deep learning model. By reducing the image dimensions, the number of pixels that must be processed by the network can also be decreased, which may lead to lower computational requirements and shorter training times. Larger images contain substantially more pixels and therefore require greater computational effort during model training. In this study, all images were resized according to the input requirements of the respective architectures. A resolution of 224×224 pixels was used for MobileNet, EfficientNetB0, DenseNet121, ResNet50, and VGG16, whereas InceptionV3 and Xception used images with a resolution of 299×299 pixels. The preprocessing pipeline was implemented using Keras-based deep learning models.

Data augmentation was employed to enhance the generalization capability of the models and mitigate potential overfitting. The augmentation procedure was performed only on the training data within each cross-validation iteration, while the corresponding validation data were kept intact to ensure an unbiased evaluation. The applied transformations consisted of random rotations up to 20° , horizontal and vertical translations of up to 20% of the image width and height, respectively, a shear transformation with a factor of 0.2, and random zooming with a maximum range of 20%. In addition, horizontal flipping was performed to increase image diversity and enhance invariance to object orientation. Newly generated pixel regions resulting from these transformations were filled using nearest-neighbor interpolation. By applying these augmentations only to the training data in each cross-validation fold, the model was exposed to a broader range of image variations while ensuring that validation performance was evaluated on unaltered data, thereby providing a more reliable assessment of generalization capability.

2.3 Select Base Model

The selected architectures cover a broad spectrum of deep learning approaches, including lightweight models optimized for computational efficiency (MobileNet and EfficientNetB0), architectures that balance accuracy and efficiency (DenseNet121 and Xception), and feature-rich networks designed for enhanced information propagation and multi-scale feature extraction (ResNet50 and InceptionV3). This diversity enables a comprehensive comparison of classification performance, computational complexity, and generalization capability across different transfer learning architectures. These architectures represent diverse network designs and computational characteristics commonly used in image classification research. Furthermore, all models were initialized with weights pre-trained on the ImageNet dataset, enabling the transfer of learned visual features to the target classification task.

To avoid ambiguity in the remainder of this manuscript, three terms are defined explicitly and are used consistently throughout. The term original pretrained architecture refers to each convolutional network in its published form together with its ImageNet weights. In this study the original architectures are used solely as a source of pretrained feature extractors and are never evaluated in their unmodified form, because their 1,000-class classification heads are not applicable to the three-class task addressed here. The term frozen baseline refers to the configuration in which all layers of the pretrained convolutional base are frozen and only the custom classification head described below is trained. The term optimized model refers to the configuration in which the upper layers of the same convolutional base are unfrozen and trained jointly with the same classification head, under the hyperparameter settings listed in Table 3. The frozen baseline and the

optimized model therefore share an identical architecture and differ only in which layers are updated during training and in the associated training hyperparameters. Accordingly, every comparison reported in this study is a comparison between the frozen baseline and an optimized configuration, and not a comparison against the original pretrained architectures.

Table 3. Grid-search hyperparameter settings for transfer learning optimization.

Experiment	Learning Rate	Dropout	Unfrozen Layers	ReduceLR OnPlateau	Dense Units
Baseline	1×10^{-4}	0.5	0	No	1024
Frozen					
HP1	1×10^{-4}	0.5	50	Yes	1024
HP2	1×10^{-5}	0.5	50	Yes	1024
HP3	1×10^{-4}	0.3	50	Yes	1024
HP4	1×10^{-4}	0.5	30	Yes	1024

For each transfer learning architecture, the original output layer was replaced with a customized classification head. This head comprises Global Average Pooling (GAP), a fully connected layer containing 1,024 neurons with ReLU activation, a dropout layer for regularization, and a Softmax layer that produces predictions for the three target categories: Early Blight, Healthy, and Late Blight. The GAP operation is formulated in Equation 1.

$$G_k = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W F_k(i, j) \quad (1)$$

$F_k(i, j)$ represents the activation value at spatial location (i, j) of the k -th feature map, and H and W denote the height and width of the feature map, respectively. The ReLU activation function is expressed as Equation 2.

$$f(x) = \max(0, x) \quad (2)$$

The final Softmax layer converts the network outputs into class probabilities according to Equation 3.

$$P(y = i) = \frac{e^{z_i}}{\sum_{j=1}^C e^{z_j}} \quad (3)$$

z_i denotes the output score for class i , and C represents the total number of target classes.

2.4 Validation Performance

Figure 2 illustrates the 5-fold cross-validation procedure adopted in this study. The dataset consists of 3,600 images equally distributed among three classes, with 1,200 images per class. The dataset was partitioned into five stratified folds, each containing 720 images (240 images per class). In each iteration, three folds were used for training (2,160 images), one fold for validation (720 images), and the remaining fold for testing (720 images). The folds were rotated across five iterations, ensuring that each image was used for testing exactly once and for validation once during the evaluation process. This strategy provides a comprehensive assessment of model performance by using all available data for both training and evaluation, thereby reducing the variance associated with a single

train–test split. The results were obtained by averaging the performance metrics across all five iterations.

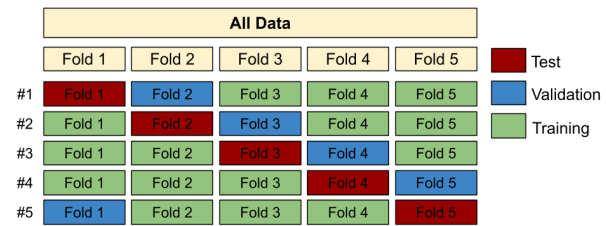


Figure 2. Data partitioning strategy for 5-fold cross-validation, showing the rotation of training (green), validation (blue), and test (red) sets across the five folds.

2.5 Hyperparameter Tuning

To identify the most effective transfer learning configuration, a series of hyperparameter tuning experiments was conducted by varying several key model parameters, including the learning rate, dropout rate, the number of unfrozen layers in the pre-trained network, and the use of adaptive learning rate scheduling. A baseline configuration was first established using a fully frozen pre-trained convolutional base. Subsequently, additional experiments were designed to investigate the impact of fine-tuning different portions of the network while maintaining a fixed dense layer size of 1,024 neurons.

The hyperparameter search space consisted of learning rates of 1×10^{-4} and 1×10^{-5} , dropout rates of 0.5 and 0.3, and fine-tuning configurations with either 30 or 50 unfrozen layers. In addition, selected experiments employed adaptive learning rate reduction through the ReduceLROnPlateau strategy. The complete set of hyperparameter configurations evaluated in this study is presented in Table 3. The optimal configuration was selected based on the highest mean validation accuracy and the lowest validation loss across the cross-validation folds.

Hyperparameter tuning was conducted using a predefined manual grid-search strategy. Instead of using an automatic search algorithm such as greedy search, random search, or Bayesian optimization, several fixed experimental configurations were manually defined and evaluated. Each configuration varied key parameters, including the learning rate, dropout rate, number of unfrozen layers, and the use of ReduceLROnPlateau. The configurations were then trained and evaluated using 5-fold stratified cross-validation. The optimal configuration was selected by ranking the experimental results by the highest mean validation accuracy and, in cases of comparable validation accuracy, the lowest mean validation loss.

The use of a predefined manual grid nevertheless constrains the search to a small and discrete region of the hyperparameter space, and configurations outside this grid may yield better performance. This design was adopted because the objective of this study was to characterise the effect of a limited number of interpretable design decisions, namely the learning rate,

the dropout rate, and the depth of fine-tuning, rather than to identify a globally optimal configuration, and because each configuration required five complete cross-validation runs for each of the six architectures, amounting to 150 independent training runs. Automated search strategies such as random search or Bayesian optimization would explore the space more thoroughly and are recommended for future work when a larger computational budget is available.

2.6 Training Configuration

The training process used 5-fold stratified cross-validation to preserve the class distribution across all folds. In each iteration, three folds were used for training, one for validation, and one for testing. Data augmentation was applied exclusively to the training subset through random rotation, translation, shearing, zooming, and horizontal flipping, while the validation and test subsets remained unchanged.

All models were trained for 50 epochs with a batch size of 32 using the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-7}$. The optimization objective was to minimize the categorical cross-entropy loss function as shown in Equation 4.

$$L = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{ic} \log(\hat{y}_{ic}) \quad (4)$$

N is the total number of samples, C is the number of classes, y_{ic} is the ground-truth label, and $\hat{y}_{ic}$ is the predicted probability for class c .

For configurations employing adaptive learning-rate scheduling, ReduceLROnPlateau was applied with validation loss as the monitoring criterion. The learning rate was decreased by 50% whenever the validation loss failed to improve for three successive epochs, subject to a minimum learning rate of 1×10^{-7} .

2.7 Evaluation Metrics

A comprehensive evaluation framework was employed to assess both the predictive performance and computational efficiency of the transfer learning models. Classification accuracy was used as the primary performance measure, representing the proportion of test samples assigned to their correct classes. In addition, the Area Under the Receiver Operating Characteristic Curve (AUC-ROC) was computed using a one-versus-rest approach to examine the models' capability to discriminate among the three potato leaf disease categories. Confusion matrices and classification reports were generated for every cross-validation fold to provide a more detailed view of class-specific predictions, including correctly and incorrectly classified samples.

Model stability and generalization were examined using stratified 5-fold cross-validation. For each evaluation metric, the mean and standard deviation obtained across the five folds were reported. ROC curves were additionally constructed to illustrate the relationship between the true-positive rate and false-positive rate at different classification thresholds. Computational cost

was evaluated through training duration, inference time, and the number of floating-point operations expressed in GFLOPs. Statistical significance was assessed using a one-sample t-test with a baseline accuracy of 0.5, while paired t-tests were conducted to compare different hyperparameter configurations. These analyses were used to determine whether the differences observed between experimental settings were statistically meaningful. Collectively, the classification, computational, and statistical measures enabled a systematic comparison of the transfer learning architectures and their corresponding hyperparameter configurations.

3. Results and Discussions

The experimental design combined transfer learning, hyperparameter tuning, and 5-fold stratified cross-validation to ensure robust, reliable evaluation. Six pretrained CNN architectures, Xception, ResNet50, EfficientNetB0, MobileNet, DensNet121, and InceptionV3, were evaluated using five hyperparameter configurations consisting of one baseline setting and four optimized settings that varied the learning rate, dropout rate, and number of unfrozen layers. Each hyperparameter configuration was assessed using 5-fold stratified cross-validation, with three folds for training, one for validation, and one for testing in each iteration. Consequently, each base model required $5 \times 5 = 25$ independent training and evaluation runs. Considering all six transfer learning architectures, the complete experimental framework involved $6 \times 5 \times 5 = 150$ independent experiments, providing a comprehensive comparison of model architectures and hyperparameter settings while minimizing the influence of dataset partitioning bias.

Before the results are presented, one characteristic of the evaluation data should be stated explicitly, as it conditions the interpretation of every accuracy value reported in this section. The PlantVillage images were acquired under controlled laboratory conditions: each image contains a single detached leaf photographed against a uniform background under consistent illumination, without occlusion, overlapping foliage, or variation in viewing distance. These conditions substantially reduce intra-class visual variability and remove most of the nuisance factors that a field-deployed classifier would encounter, which is the principal reason why near-ceiling accuracies are attainable on this dataset. The accuracies reported below should therefore be read as performance on this controlled distribution and as an upper reference for field deployment, rather than as an estimate of performance under real agricultural conditions. This limitation is revisited when the best fine-tuned models are discussed and in the Conclusion.

3.1 The Effect of ReduceLROnPlateau

The effect of the optimized training strategy with ReduceLROnPlateau was evaluated by comparing each frozen baseline model with its best optimized

configuration. The optimized configurations improved the average accuracy from 0.9876 to 0.9975, corresponding to an average accuracy gain of approximately 0.0100. As shown in Figure 3, all six transfer learning architectures achieved positive accuracy gains after optimization. The largest improvement was obtained by InceptionV3, with an accuracy gain of 0.0236, followed by Xception with a gain of 0.0150. These results indicate that the optimization strategy was particularly beneficial for architectures with relatively lower frozen baseline performance. As defined in Section 2, the frozen baseline and the optimized configurations share an identical architecture and differ in four training settings simultaneously: the number of unfrozen layers, the dropout rate, the learning rate, and the use of adaptive learning-rate scheduling. The gains reported in this subsection are therefore attributable to the combination of these settings rather than to ReduceLRonPlateau considered in isolation.

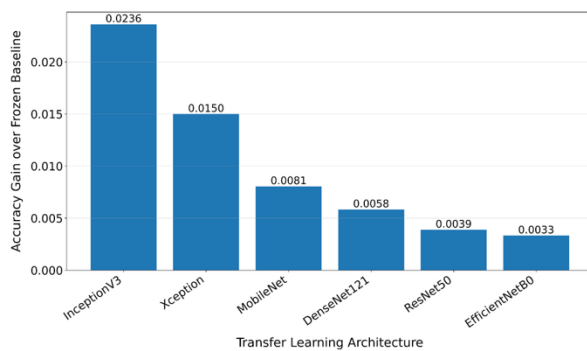


Figure 3. Accuracy gain after optimized fine-tuning with ReduceLRonPlateau compared with the frozen baseline model.

Figure 3 shows that InceptionV3 and Xception benefited the most from the optimized configuration, while ResNet50 and EfficientNetB0 showed smaller gains. This does not necessarily indicate weak optimization performance for ResNet50 and EfficientNetB0; instead, both models already achieved high baseline accuracy, leaving less room for further improvement. Therefore, the observed accuracy gains suggest that partial fine-tuning, combined with adaptive learning rate scheduling, helped the models learn more disease-specific features than the frozen baseline architecture.

The validation loss results provide additional evidence of the positive effect of the optimized configuration. As illustrated in Figure 4, all architectures showed substantial reductions in validation loss after optimization. Across the six architectures, the average validation loss decreased from 0.0316 in the frozen baseline models to 0.0025 in the optimized configurations, representing an average reduction of approximately 91.78%. Xception achieved the highest validation loss reduction at 97.8%, followed by EfficientNetB0 at 95.7% and DenseNet121 at 92.3%. These reductions indicate that the optimized models produced more stable and confident predictions during validation.

Figure 4 also shows that even models with smaller accuracy gains still experienced large decreases in validation loss. For example, EfficientNetB0 had one of the smallest accuracy gains, yet its validation loss decreased by 95.7%. This suggests that the optimization process improved not only the final classification accuracy but also the confidence and stability of the prediction probabilities. In this context, validation loss is useful because it reflects the quality of probability estimation, whereas accuracy only measures whether the final predicted class is correct.

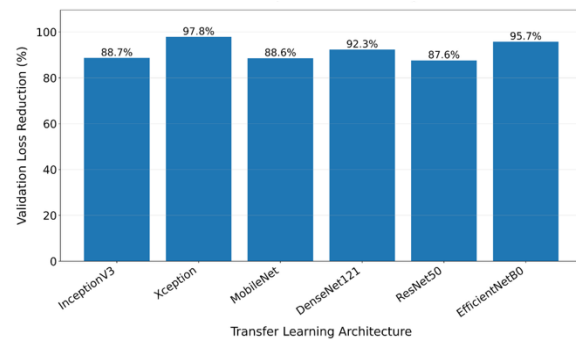


Figure 4. Validation loss reduction after optimized fine-tuning with ReduceLRonPlateau compared with the frozen baseline model.

Overall, the results in Figures 3 and 4 demonstrate that the optimized fine-tuning strategy with ReduceLRonPlateau improved both classification performance and training stability. ReduceLRonPlateau reduced the learning rate when validation loss stopped improving, allowing the model to continue optimizing with smaller weight updates. However, because ReduceLRonPlateau was applied together with partial fine-tuning of the upper layers, the observed improvement should be interpreted as the combined effect of adaptive learning rate scheduling and fine-tuning rather than the isolated effect of ReduceLRonPlateau alone. An ablation isolating ReduceLRonPlateau while holding the number of unfrozen layers, the dropout rate, and the learning rate fixed was not performed in this study, and is required before any independent contribution can be attributed to the scheduler.

3.2 Best Fine-Tuning Models

Among the evaluated fine-tuning configurations, the best overall performance was achieved by Xception with the HP3 configuration, which applied a learning rate of 1×10^{-4} , dropout of 0.3, fine-tuning of the last 50 layers, and ReduceLRonPlateau. This configuration achieved a mean test accuracy of 0.999444, a standard deviation of 0.000761, and a mean AUC of 0.999997 across 5-fold stratified cross-validation. These results indicate that Xception with partial fine-tuning achieved the strongest classification performance among the tested architectures, suggesting that its depthwise separable convolutional structure was highly effective at extracting discriminative features from potato leaf disease images. As noted at the beginning of this section, an accuracy of this magnitude reflects the controlled acquisition conditions of the PlantVillage

dataset as much as the capability of the model. Under field conditions, where illumination varies, backgrounds are cluttered, several leaves may appear in a single image, and disease severity differs across plants, a lower accuracy should be expected. The present results therefore establish the performance attainable on controlled imagery and do not by themselves demonstrate readiness for operational deployment.

Although Xception achieved the highest mean accuracy, ResNet50 also showed highly competitive performance. The ResNet50 HP3 configuration achieved a mean accuracy of 0.998889 and a lower mean validation loss of 0.000841. This indicates that Xception provided the best classification accuracy, while ResNet50 demonstrated very stable validation performance. Therefore, Xception HP3 can be considered the best-performing fine-tuned model in terms of accuracy and AUC, whereas ResNet50 HP3 may be considered a strong alternative when validation loss and computational efficiency are also considered.

As illustrated in Figure 5, Xception achieved the highest fine-tuning performance among the evaluated architectures, with a mean test accuracy of 0.999444. This result indicates that Xception was the most effective architecture after partial fine-tuning and adaptive learning rate scheduling. ResNet50 achieved the second-highest performance with a mean accuracy of 0.998889, followed by EfficientNetB0 with 0.998333. Although all fine-tuned models achieved high accuracy, Xception showed the strongest overall classification capability.

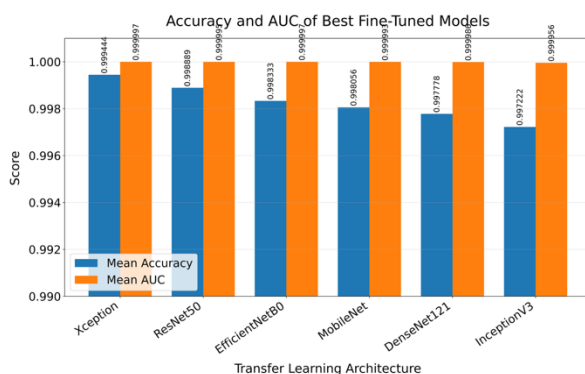


Figure 5. Accuracy and AUC of Best Fine-Tuned Models

The comparison between accuracy and AUC in Figure 5 further confirms the strong discriminative ability of the fine-tuned models. Xception achieved the highest mean accuracy and an AUC of 0.999997, indicating it was highly effective at distinguishing the three potato leaf classes. ResNet50 and EfficientNetB0 also achieved AUC values close to 1.0000, showing that their probability-based class separation was also very strong. Overall, the fine-tuning results demonstrate that Xception HP3 achieved the best performance, while ResNet50 and EfficientNetB0 remained strong alternatives.

The one-versus-rest ROC curves described in Section 2 were computed for each class in every cross-validation fold. Because the curves for all architectures lie within a very narrow band near the ideal operating point, plotting them would not be visually informative; their behaviour is therefore reported numerically through the per-class AUC values discussed below rather than as a separate figure. Because the AUC is threshold-independent, values approaching unity indicate that the separation between each class and the remaining two is preserved across the entire range of decision thresholds, and not only at the default threshold used to compute accuracy. For Xception HP3, the mean AUC of 0.999997 implies that the corresponding ROC curves lie almost exactly along the ideal operating path, so that nearly all positive samples of a given class are recovered at a negligible false-positive rate. Read together with the accuracy results, this indicates that the reported accuracy is not an artefact of a favourable decision threshold. The residual departure from the ideal curve is concentrated in the Early Blight and Late Blight pair, which is expected because the two diseases produce lesions with overlapping colour and texture characteristics during the early stage of infection, whereas healthy leaves are separated from both diseased classes without measurable overlap.

In summary, the best fine-tuned model identified in this study is Xception trained under the HP3 configuration, in which the learning rate was set to 1×10^{-4} , the dropout rate to 0.3, and the last 50 convolutional layers were unfrozen and trained jointly with the classification head, with adaptive learning-rate scheduling through ReduceLROnPlateau enabled. As noted above, these four settings were applied together rather than individually, so the resulting performance is attributable to their combination.

3.3 Computational Complexity

Based on Figure 6, the combined analysis of GFLOPs and mean accuracy demonstrates the trade-off between predictive performance and computational complexity. Xception achieved the highest mean accuracy of 0.999444; however, it also required the highest computational cost, with 16.77 GFLOPs per image. This indicates that Xception is the most accurate fine-tuned model, but its computational demand may limit its suitability for resource-constrained deployment.

EfficientNetB0 achieved the best efficiency-oriented performance, with a mean accuracy of 0.998333 and only 0.80 GFLOPs per image. Although its accuracy was slightly lower than Xception, the difference was very small, while the computational cost was substantially lower. Compared with EfficientNetB0, Xception improved accuracy by only 0.001111 but required approximately 21 times more GFLOPs. Therefore, EfficientNetB0 offers the most favorable accuracy–complexity trade-off, particularly for applications that require fast, lightweight inference.

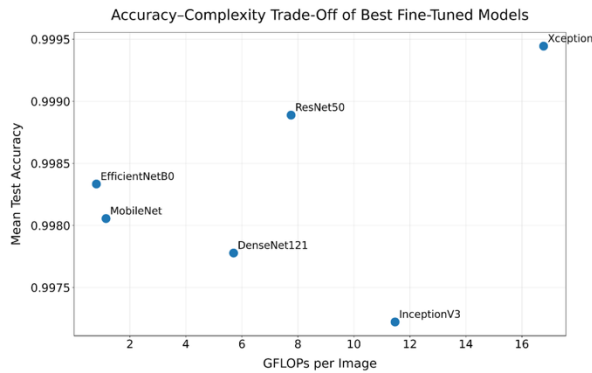


Figure 6. Accuracy-complexity trade-off of the best fine-tuned models.

ResNet50 represented a balanced alternative between accuracy and computational complexity. It achieved a mean accuracy of 0.998889 with 7.76 GFLOPs per image, placing it between EfficientNetB0 and Xception. This suggests that ResNet50 is suitable when a higher accuracy than EfficientNetB0 is desired, but the very high computational cost of Xception is not preferred. Overall, Xception is the best model for maximum accuracy, EfficientNetB0 is the best for computational efficiency, and ResNet50 provides a strong compromise between the two.

Two qualifications apply to this analysis. First, GFLOPs and the inference times reported here were measured in the training environment and represent a theoretical proxy for computational cost. No physical on-device latency measurement was performed in this study, and factors such as memory bandwidth, operator support on the target runtime, quantisation, and thermal throttling can cause measured on-device latency to deviate substantially from what the GFLOPs figures suggest. Second, the recommendation of EfficientNetB0 for edge deployment should therefore be understood as a complexity-based indication that requires confirmation through on-device benchmarking on the intended hardware, which is left to future work.

3.4 Comparison with Existing Studies

Table 5 presents a comparison between the proposed approach and previously reported methods for potato leaf disease classification. The reviewed studies have explored a wide range of classification techniques, from conventional machine learning algorithms, including SVM, k-NN, ANN, and Random Forest, to various deep learning architectures such as AlexNet, VGG16, VGG19, ResNet50, MobileNet, Mask R-CNN, and customized CNN models. The PlantVillage dataset is the most frequently adopted benchmark, particularly for experiments involving three disease categories. Other studies have employed alternative data sources, including self-generated datasets, Kaggle collections, and the AI Challenger Global AI Contest dataset, with the number of target classes varying between two, three, and five.

Compared with previous PlantVillage-based studies, the proposed Xception model, combined with fine-

tuning and ReduceLROnPlateau, achieved the highest reported performance, achieving 99.94% accuracy in three-class classification. This result is higher than the performance reported by VGG19 fine-tuning in [21] (97.8%), the self-built CNN model in [4] (96.75%), and the ANN-based method in [21] (96.5%). The proposed model also outperformed the study in [25], which evaluated VGG16, VGG19, ResNet50, MobileNet, and fine-tuned VGG16 and reported a best accuracy of 97.89% on the PlantVillage dataset. These results indicate that the proposed optimization strategy was more effective in improving feature adaptation for potato leaf disease classification.

Table 5. Comparison with Existing Work. The reported accuracies were obtained under differing datasets, class distributions, preprocessing procedures, and evaluation protocols, and are therefore indicative rather than strictly comparable.

Author	Best Method	Dataset	Acc.
[21]	VGG19 for fine-tuning	PlantVillage	97.8%
[4]	Self-build CNN	PlantVillage	96.75%
[22]	Own architecture	PlantVillage	92.00%
[23]	SVM, k-NN, ANN, RF	Global AI	97.40%
[7]	RF	Own	97.00%
[24]	Mask R-CNN	Own	81.40%
[20]	VGG16, VGG19	PlantVillage	91.00%
[25]	VGG16 fine-tuned	PlantVillage	97.89%
[18]	ANN	PlantVillage	96.5%
[26]	Own Sequential mode	Dataquest	97.00%
[27]	SVM	PlantVillage	87.60%
[1]	AlexNet, VGG-16, VGG-19	Kaggle competition	98.00%
Proposed	Xception+ReduceLROnPlateau	PlantVillage Dataset	99.94%

The improvement can be attributed to transfer learning with the Xception architecture, partial fine-tuning of the upper convolutional layers, and adaptive learning rate scheduling with ReduceLROnPlateau. Unlike frozen transfer learning, which only trains the classification head, the proposed fine-tuning strategy allowed the model to adapt high-level visual features to disease-specific patterns, such as lesion shape, leaf discoloration, and texture differences. ReduceLROnPlateau further improved the training process by reducing the learning rate when validation loss stopped improving, allowing the model to continue optimizing with smaller, more stable weight updates.

Although several previous studies also achieved high accuracy, direct comparisons should be interpreted with caution because of differences in dataset sources, numbers of classes, preprocessing methods, model architectures, and evaluation protocols. For example, [1] achieved 97.40% using traditional classifiers on the AI Challenger Global AI Contest dataset with five classes, while [7] achieved 97.00% using Random Forest on a generated dataset with two classes. Similarly, [28] used Mask R-CNN on a generated dataset and obtained 81.40% for two classes. These differences show that dataset characteristics and classification complexity strongly influence model performance. Nevertheless, among the studies using the PlantVillage dataset for three-class potato disease classification, the proposed method achieved the

strongest performance. Accordingly, Table 5 should be read as an indication of the relative standing of the proposed approach rather than as a strictly controlled benchmark, since none of the referenced results were reproduced under the experimental protocol used in this study.

Overall, the comparison demonstrates that the proposed Xception + ReduceLROnPlateau approach provides a competitive and effective solution for potato leaf disease classification. The achieved accuracy of 99.94% indicates that the optimized transfer learning framework can outperform several existing CNN-based and traditional machine learning approaches. In addition, the use of cross-validation, hyperparameter tuning, and computational analysis provides a more comprehensive evaluation of the proposed model than studies that report only classification accuracy.

4. Conclusions

This study investigated an optimized CNN-based transfer learning framework for potato leaf disease classification using six pre-trained architectures: MobileNet, EfficientNetB0, DenseNet121, ResNet50, InceptionV3, and Xception. A stratified 5-fold cross-validation scheme was employed to obtain reliable performance estimates while maintaining class proportions. Hyperparameter configurations involving learning rate, dropout, the number of trainable layers, and ReduceLROnPlateau were systematically evaluated. Fine-tuned models generally outperformed their frozen counterparts, demonstrating that adapting selected layers improved the extraction of disease-related features.

Among the evaluated models, Xception with HP3 produced the highest performance, achieving a mean test accuracy of 0.999444 and a mean AUC of 0.999997 using a learning rate of 1×10^{-4} , 0.3 dropout, and fine-tuning of the final 50 layers with ReduceLROnPlateau. ResNet50 achieved a comparable mean accuracy of 0.998889 with lower computational cost. Overall, the results indicate that combining selective fine-tuning with appropriate hyperparameter settings can substantially enhance potato leaf disease classification. However, the observed improvement cannot be attributed to ReduceLROnPlateau alone because the optimized configurations simultaneously differed from the frozen baseline in learning rate, dropout, and the number of unfrozen layers.

The computational analysis using GFLOPs showed that the most accurate model was not necessarily the most efficient. Xception achieved the highest accuracy but incurred the highest computational cost, at 16.77 GFLOPs per image. In contrast, EfficientNetB0 achieved a mean accuracy of 0.998333 with only 0.80 GFLOPs per image, making it the most suitable architecture for efficiency-oriented deployment. MobileNet also demonstrated lightweight performance, requiring only 1.15 GFLOPs per image and achieving a mean accuracy of 0.998056. Therefore, Xception is

recommended when maximum classification accuracy is the main priority, while EfficientNetB0 is recommended when computational efficiency and deployment feasibility are more important.

Overall, the results demonstrate that the proposed transfer learning optimization strategy, which combines partial fine-tuning, dropout regularization, data augmentation, and ReduceLROnPlateau, is effective for potato leaf disease classification. The study also highlights the importance of evaluating both accuracy and computational complexity when selecting a model for practical agricultural applications. Future work may extend this study by testing the proposed models on field-collected images, evaluating performance under different lighting and background conditions, and deploying the most efficient model on mobile or edge-computing devices for real-time disease diagnosis.

Three limitations qualify these conclusions. First, all results were obtained on PlantVillage images acquired under controlled laboratory conditions, so the reported accuracies represent an upper reference rather than an estimate of performance in the field, and validation on field-collected imagery is required before deployment claims can be made. Second, the contribution of ReduceLROnPlateau was not isolated through a dedicated ablation, and the reported gains reflect the joint effect of partial fine-tuning, dropout configuration, learning-rate selection, and adaptive scheduling. Third, hyperparameters were explored through a predefined manual grid, and no on-device latency measurement was carried out to support the edge-deployment recommendation. Future work should accordingly address these points by evaluating the models on field-collected images under varying lighting and background conditions, conducting an isolated ablation of the learning-rate scheduler, employing automated hyperparameter search such as random search or Bayesian optimization, and benchmarking on-device latency for the most efficient architecture on the intended edge hardware.

Declarations

AI Usage Statement

The authors used GPT exclusively for language refinement and readability enhancement. All scientific content, analysis, interpretation, and conclusions are solely the authors' responsibility.

Author Contributions

Andi Sunyoto: conceptualization and formulation of the research idea, research design and methodology, algorithm implementation, programming, model development, model optimization, research supervision, coordination of research activities, data analysis and interpretation, and critical review of the manuscript; Yoga Pristyanto: experimental design and execution, model evaluation, comparison of experimental results, validation of model performance, manuscript preparation, manuscript revision, and final

proofreading; Anggit Ferdita Nugraha: literature review, dataset preparation and preprocessing, data cleaning and organization, experimental data analysis, and critical review of the manuscript.

Funding Statement

The authors gratefully acknowledge the Research Group of Computer Vision and Intelligence System (CVIS), Universitas Amikom Yogyakarta, for its support throughout this study. This research was supported by the Directorate General of Higher Education, Research and Technology, Ministry of Education, Culture, Research, and Technology of the Republic of Indonesia, under grant number 036/E5/PG.02.00/2022.

Conflict of Interest Statement

The authors declare no conflict of interest.

Ethical Clearance

This study did not involve human participants, animal subjects, or any procedures requiring ethical approval. Therefore, no ethical clearance was required.

Data and Code Availability

The PlantVillage potato leaf dataset analysed in this study is publicly available from the source cited in [15], and no new data were generated in this study. The implementation code and the experimental result files supporting the findings of this study are openly available in a public GitHub repository at <https://github.com/myandisun/potato-leaf-classification-01>.

References

- [1] N. E. M. Khalifa, M. H. N. Taha, L. M. A. El-Maged, and A. E. Hassanien, "Artificial Intelligence in Potato Leaf Disease Classification: A Deep Learning Approach," 2021, pp. 63–79. doi: [10.1007/978-3-030-59338-4_4](https://doi.org/10.1007/978-3-030-59338-4_4).
- [2] T.-Y. Lee, J.-Y. Yu, Y.-C. Chang, and J.-M. Yang, "Health Detection for Potato Leaf with Convolutional Neural Network," in *2020 Indo – Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)*, IEEE, Feb. 2020, pp. 289–293. doi: [10.1109/Indo-TaiwanICAN48429.2020.9181312](https://doi.org/10.1109/Indo-TaiwanICAN48429.2020.9181312).
- [3] R. Balodi, S. Bisht, A. Ghatak, and K. H. Rao, "Plant Disease Diagnosis: Technological Advancements and Challenges," *Indian Phytopathol.*, vol. 70, no. 3, Sep. 2017, doi: [10.24838/ip.2017.v70.i3.72487](https://doi.org/10.24838/ip.2017.v70.i3.72487).
- [4] U. Barman, D. Sahu, G. G. Barman, and J. Das, "Comparative Assessment of Deep Learning to Detect the Leaf Diseases of Potato based on Data Augmentation," in *2020 International Conference on Computational Performance Evaluation (ComPE)*, IEEE, Jul. 2020, pp. 682–687. doi: [10.1109/ComPE49325.2020.9200015](https://doi.org/10.1109/ComPE49325.2020.9200015).
- [5] N. R. Deepa and N. Nagarajan, "Kuan noise filter with Hough transformation based reweighted linear program boost classification for plant leaf disease detection," *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 1, p. 409, 2023, doi: [10.1007/s12652-022-04156-6](https://doi.org/10.1007/s12652-022-04156-6).
- [6] M. Islam, Anh Dinh, K. Wahid, and P. Bhowmik, "Detection of potato diseases using image segmentation and multiclass support vector machine," in *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*, IEEE, Apr. 2017, pp. 1–4. doi: [10.1109/CCECE.2017.7946594](https://doi.org/10.1109/CCECE.2017.7946594).
- [7] Md. A. Iqbal and K. H. Talukder, "Detection of Potato Disease Using Image Segmentation and Machine Learning," in *2020 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*, IEEE, Aug. 2020, pp. 43–47. doi: [10.1109/WiSPNET48689.2020.9198563](https://doi.org/10.1109/WiSPNET48689.2020.9198563).
- [8] U. P. Singh, S. S. Chouhan, S. Jain, and S. Jain, "Multilayer Convolution Neural Network for the Classification of Mango Leaves Infected by Anthracnose Disease," *IEEE Access*, vol. 7, pp. 43721–43729, 2019, doi: [10.1109/ACCESS.2019.2907383](https://doi.org/10.1109/ACCESS.2019.2907383).
- [9] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Comput. Electron. Agric.*, vol. 145, Feb. 2018, doi: [10.1016/j.compag.2018.01.009](https://doi.org/10.1016/j.compag.2018.01.009).
- [10] J. Chen, X. Deng, Y. Wen, W. Chen, A. Zeb, and D. Zhang, "Weakly-supervised learning method for the recognition of potato leaf diseases," *Artif. Intell. Rev.*, vol. 56, no. 8, pp. 7985–8002, Sep. 2023, doi: [10.1007/s10462-022-10374-3](https://doi.org/10.1007/s10462-022-10374-3).
- [11] M. Agarwal, A. Singh, S. Arjaria, A. Sinha, and S. Gupta, "ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network," *Procedia Comput. Sci.*, vol. 167, pp. 293–301, 2020, doi: [10.1016/j.procs.2020.03.225](https://doi.org/10.1016/j.procs.2020.03.225).
- [12] G. Sambasivam and G. D. Opiyo, "A predictive machine learning application in agriculture: Cassava disease detection and classification dataset with imbalanced dataset using convolutional neural networks," *Egyptian Informatics Journal*, vol. 22, no. 1, Mar. 2021, doi: [10.1016/j.eij.2020.02.007](https://doi.org/10.1016/j.eij.2020.02.007).
- [13] A. Dengel, "Special Issue on Artificial Intelligence in Agriculture," *KI - Künstliche Intelligenz*, vol. 27, no. 4, pp. 309–311, Nov. 2013, doi: [10.1007/s13218-013-0275-y](https://doi.org/10.1007/s13218-013-0275-y).
- [14] D. Bonino and G. Procaccianti, "Exploiting semantic technologies in smart environments and grids: Emerging roles and case studies," *Sci. Comput. Program.*, vol. 95, pp. 112–134, Dec. 2014, doi: [10.1016/j.scico.2014.02.018](https://doi.org/10.1016/j.scico.2014.02.018).
- [15] David. P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," Nov. 2015, Accessed: Oct. 28, 2021. [Online]. Available: <https://arxiv.org/abs/1511.08060>
- [16] B. Wu, Z. Liu, Z. Yuan, G. Sun, and C. Wu, "Reducing Overfitting in Deep Convolutional Neural Networks Using Redundancy Regularizer," 2017, pp. 49–55. doi: [10.1007/978-3-319-68612-7_6](https://doi.org/10.1007/978-3-319-68612-7_6).
- [17] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: Fast and Flexible Image Augmentations," *Information*, vol. 11, no. 2, p. 125, Feb. 2020, doi: [10.3390/info11020125](https://doi.org/10.3390/info11020125).
- [18] K. Sanjeev, N. K. Gupta, W. J. Jeberson, and S. Paswan, "Early Prediction of Potato Leaf Diseases Using ANN Classifier," *Oriental journal of computer science and technology*, vol. 13, no. 0203, pp. 129–134, Jan. 2021, doi: [10.13005/ojcs13.0203.11](https://doi.org/10.13005/ojcs13.0203.11).
- [19] S. Kumar, S. Paswan, and R. Kumari, "Early prediction of potato tuber diseases using ann classifier," *Progressive Agriculture*, vol. 22, no. 2, pp. 162–165, 2022, doi: [10.5958/0976-4615.2021.00003.x](https://doi.org/10.5958/0976-4615.2021.00003.x).
- [20] R. A. Sholihati, I. A. Sulistijono, A. Risnumawan, and E. Kusumawati, "Potato Leaf Disease Classification Using Deep Learning Approach," in *2020 International Electronics Symposium (IES)*, IEEE, Sep. 2020, pp. 392–397. doi: [10.1109/IES50839.2020.9231784](https://doi.org/10.1109/IES50839.2020.9231784).
- [21] D. Tiwari, M. Ashish, N. Gangwar, A. Sharma, S. Patel, and S. Bhardwaj, "Potato Leaf Diseases Detection Using Deep Learning," in *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, IEEE, May 2020, pp. 461–466. doi: [10.1109/ICICCS48265.2020.9121067](https://doi.org/10.1109/ICICCS48265.2020.9121067).
- [22] A. J. Rozaqi and A. Sunyoto, "Identification of Disease in Potato Leaves Using Convolutional Neural Network (CNN) Algorithm," in *2020 3rd International Conference on Information and Communications Technology (ICOIACT)*, IEEE, Nov. 2020, pp. 72–76. doi: [10.1109/ICOIACT50329.2020.9332037](https://doi.org/10.1109/ICOIACT50329.2020.9332037).
- [23] C. Hou et al., "Recognition of early blight and late blight diseases on potato leaves based on graph cut segmentation," *J. Agric. Food Res.*, vol. 5, p. 100154, Sep. 2021, doi: [10.1016/J.JAFR.2021.100154](https://doi.org/10.1016/J.JAFR.2021.100154).
- [24] J. Johnson et al., "Enhanced Field-Based Detection of Potato Blight in Complex Backgrounds Using Deep Learning," *Plant*

- Phenomics*, vol. 2021, pp. 1–13, May 2021, doi: [10.34133/2021/9835724](https://doi.org/10.34133/2021/9835724).
- [25] K. K. Chakraborty, R. Mukherjee, C. Chakraborty, and K. Bora, “Automated recognition of optical image based potato leaf blight diseases using deep learning,” *Physiol. Mol. Plant Pathol.*, vol. 117, p. 101781, Jan. 2022, doi: [10.1016/J.PMPP.2021.101781](https://doi.org/10.1016/J.PMPP.2021.101781).
- [26] Md. K. R. Asif, Md. A. Rahman, and Most. H. Hena, “CNN based Disease Detection Approach on Potato Leaves,” in *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*, IEEE, Dec. 2020, pp. 428–432. doi: [10.1109/ICISS49785.2020.9316021](https://doi.org/10.1109/ICISS49785.2020.9316021).
- [27] D. Das, M. Singh, S. S. Mohanty, and S. Chakravarty, “Leaf Disease Detection using Support Vector Machine,” in *2020 International Conference on Communication and Signal Processing (ICCSP)*, IEEE, Jul. 2020, pp. 1036–1040. doi: [10.1109/ICCSP48568.2020.9182128](https://doi.org/10.1109/ICCSP48568.2020.9182128).
- [28] J. Johnson *et al.*, “Enhanced Field-Based Detection of Potato Blight in Complex Backgrounds Using Deep Learning,” *Plant Phenomics*, vol. 2021, p. 9835724, 2021, doi: <https://doi.org/10.34133/2021/9835724>.