Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

Augmentation Strategy and Hyperparameter Optimization Using Optuna for Potato Leaf Disease Classification in Uncontrolled Environment

Harri Kurniawan Rofiqi*[1], Edi Noersasongko[2], Sri Winarno[3], M. Arief Soeleman[4]

[1,2,3,4]Informatics Engineering, Faculty of Computer Science, Universitas Dian Nuswantoro, Indonesia

Email: [1]p31202302576@mhs.dinus.ac.id

Received : Jun 17, 2025; Revised : Nov 6, 2025; Accepted : Nov 9, 2025; Published : Apr 15, 2026

Abstract

Image-based classification of potato leaf diseases presents a significant challenge, particularly when data are collected in uncontrolled field environments. While Convolutional Neural Networks (CNNs) and Computer Vision have been widely used for plant disease identification, most previous studies relied on laboratory datasets with uniform lighting and backgrounds, limiting their real-world applicability. This study proposes an integrated framework that combines data augmentation, class balancing using the Synthetic Minority Over-sampling Technique (SMOTE), and automated hyperparameter optimization through Optuna to enhance the robustness and accuracy of CNN-based models. A total of 3,076 high-resolution potato leaf images representing seven disease classes were evaluated across five CNN architectures and three training scenarios. The MobileNetV3-Large model achieved the best baseline performance with an accuracy of 0.863 and F1-score of 0.868, while Optuna-based optimization further improved performance to 0.895 accuracy, 0.913 precision, 0.906 recall, and 0.904 F1-score, demonstrating the effectiveness of adaptive optimization in improving model generalization. The integration of augmentation, SMOTE, and Optuna resulted in an intelligent and efficient system resilient to environmental variability, showing strong potential for automatic early detection of potato leaf diseases in real agricultural settings. This research contributes to the advancement of Informatics and Artificial Intelligence by promoting adaptive computer vision approaches for smart agriculture and real-world image-based diagnostic systems.

Keywords : *Augmentation, Convolutional Neural Network (CNN), Optuna, Potato Leaf Disease Classification, SMOTE.*

1. INTRODUCTION

1.1. Background

The highland areas of Central Java are not only renowned for their scenic landscapes but also serve as critical hubs for agricultural production. Potatoes stand out as one of the region's leading horticultural commodities, supported by ideal geographic and microclimatic conditions, including low temperatures and stable humidity. As a high-value crop, potato farming plays a pivotal role in sustaining local farmers' livelihoods and enhancing regional economic resilience. Remarkably, potato harvests from these plateau areas have successfully reached both national and international markets[1].

Despite its economic reasons, potato farming faces a major challenge: the crop's high susceptibility to various foliar diseases caused by bacteria, fungi, viruses, and pests. Early blight (*Alternaria solani*) and late blight (*Phytophthora infestans*) are among the most destructive potato diseases, capable of reducing crop yields by over 50% when not properly managed in a timely manner. [2].

Early disease detection is crucial to prevent widespread damage and to maintain the quality of the yield. However, traditional manual inspection methods are time-consuming, labor-intensive, and heavily

Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

reliant on the expertise of individual farmers, making them less effective, particularly in large-scale farming operations[3]. At the field level, similar limitations are noted that traditional visual inspection often leads to delayed or inaccurate diagnoses, thereby hindering effective disease management strategies [4].

In the context of the Industrial Revolution 4.0 and the digitalization of the agricultural sector, automated approaches based on artificial intelligence technologies offer a promising solution to accelerate the process of plant disease detection and diagnosis with higher accuracy and consistency. Recent studies emphasize that deep learning-based image analysis has significantly improved the accuracy and reliability of automated plant disease identification systems [5]-[6].

Recent advancements in artificial intelligence, particularly in Computer Vision and Convolutional Neural Networks (CNNs) have introduced promising solutions for image-based plant disease classification. CNNs have demonstrated strong capabilities in identifying complex visual patterns, including shape, texture, and color from leaf images [3]. However, most existing studies rely on datasets collected under controlled conditions, such as those provided by PlantVillage and Potato Leaf Diseases (PLD), where images feature clean backgrounds and uniform lighting. On the other hands, these datasets do not fully capture the variability and complexity of real-world agricultural environments[7]-[8]. Although these approaches have yielded high classification accuracy, they fall short of representing the real-world challenges encountered in natural agricultural settings.

In response to these limitations, a study conducted by Shabrina et al. successfully developed a dataset comprising 3,076 high-resolution images of potato leaves, collected directly from agricultural fields under uncontrolled, real-world conditions[9]. The dataset comprises seven distinct disease classes and presents realistic variability in background, lighting conditions, camera angles, and shooting distances mirroring the diversity encountered in actual field environments[10]. However, the uneven distribution of images across classes introduces a new challenge: class imbalance, which can hinder model performance and lead to biased predictions[11]-[12]. A literature review by Upadhyay et al. [13] also emphasizes that discrepancies in visual distribution between training and testing conditions are a major factor contributing to reduced model accuracy in real-world applications. Certain diseases such as those caused by fungi are more commonly observed, while others, like nematode infections, are significantly rarer, resulting in an imbalanced distribution of data across classes.

In this context, data augmentation emerges as a critical strategy for enhancing visual diversity, particularly within minority classes. Techniques such as random horizontal flipping, contrast and brightness adjustment, and rotation up to 90 degrees help improve model generalization without altering the essential visual features of the images [14]-[15]. In addition, synthetic oversampling methods such as SMOTE (Synthetic Minority Over-sampling Technique) are employed to balance class distribution without directly duplicating existing data. By generating new synthetic samples for underrepresented classes, SMOTE helps mitigate class imbalance and enhances the robustness of the classification model [16]-[17]. This strategy has proven effective in improving model performance, particularly for previously underrepresented classes[18].

Although several well-known CNN architectures such as EfficientNet, ResNet, and MobileNet have been evaluated on this dataset, the classification accuracy remains significantly lower compared to results obtained using controlled datasets[19]. For instance, the highest accuracy achieved on the field-acquired dataset was 73.63%, considerably lower than the 98.15% reported on the PlantVillage dataset [9]. This highlights that potato leaf disease classification in uncontrolled environments remains a significant challenge that requires specialized strategies. A follow-up study by Boukhlifa and Chibani introduced a lightweight CNN architecture incorporating depthwise separable convolutions and skip connections. When evaluated on the same dataset, this model demonstrated a substantial improvement in accuracy, reaching 80.45% and up to 87.82% [20], [19]. These findings underscore the importance of designing models that

Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

are both efficient and adaptable to visual noise and lighting variability commonly encountered in real-world agricultural settings.

Furthermore, in order to optimize CNN performance on complex datasets, hyperparameter tuning plays a critical role and also serves as an essential evaluation step. The use of Optuna or a define-by-run–based hyperparameter optimization framework enables efficient and adaptive exploration of parameter spaces, facilitating the development of more effective and robust models [21]. By leveraging features from the TensorFlow library such as early stopping, parallel search, and dynamic search spaces, Optuna has demonstrated its effectiveness in improving model accuracy across various domains, including image-based plant disease classification[22]-[23].

Therefore, this study focuses on enhancing the performance of potato leaf disease classification by combining data augmentation strategies with CNN and hyperparameter optimization using Optuna. The integration of these three components is expected to yield an accurate and adaptive potato leaf disease classification model that can be effectively implemented in real-world agricultural environments.

1.2. Literature Review

1.2.1. Previous Study

The study conducted by Shabrina et al. [9] revealed that data augmentation techniques were applied to address class imbalance within a dataset collected from uncontrolled environments, and the following table presents a summary of their research.

Table 1 Related Work

CNN Architectures	Original				Augmented			
	Test Accuracy	Precision	Recall	F1-Score	Test Accuracy	Precision	Recall	F1-Score
EfficientNetV2B3	0.736	0.742	0.736	0.730	0.723	0.737	0.723	0.719
MobileNetV3-Large	0.720	0.731	0.720	0.713	0.704	0.709	0.704	0.703
VGG-16	0.598	0.605	0.598	0.590	0.562	0.569	0.562	0.560
ResNet50	0.681	0.700	0.681	0.674	0.662	0.665	0.662	0.660
DenseNet121	0.591	0.605	0.591	0.591	0.585	0.585	0.585	0.584

Table 1 presents the comparison of CNN model performance before and after data augmentation. The results indicate that the augmentation process did not lead to a significant improvement in the performance of the tested deep learning models. For instance, the accuracy of the EfficientNetV2B3 model decreased from 0.7363 to 0.7235, while its F1-score dropped from 0.7302 to 0.7199 after augmentation. A similar trend was observed in the MobileNetV3-Large model, with accuracy declining from 0.7203 to 0.7042 and F1-score from 0.7131 to 0.7037. This performance decline may be attributed to the high complexity of the dataset, as well as the possibility that the applied augmentation process did not effectively improve model performance, causing the deep learning model to struggle in consistently recognizing patterns, even though the amount of data was increased through augmentation.

1.2.2. Uncontrolled Environment

Classifying potato leaf diseases in uncontrolled environments presents significant challenges that cannot be overlooked. Environmental variability such as changes in lighting conditions, diverse backgrounds, and differing camera angles introduces substantial visual “noise” that hinders accurate disease identification[24]. Barbedo [25] emphasized that plant images captured in natural environments are

significantly more complex than those obtained under controlled conditions. Field-acquired images often exhibit obstructions or overlap with other elements such as surrounding foliage, mulch, or foreign objects factors that further degrade model accuracy. In contrast to controlled image acquisition, which typically features clean backgrounds and uniform lighting, the dynamic nature of real-world environments introduces a higher level of complexity that machine learning algorithms must contend with[26].

1.2.3. Data Augmentation Strategy

Data augmentation is a critical technique in training deep learning models, particularly for image classification tasks, as it enhances data diversity and reduces the risk of overfitting. Transformations such as rotation, flipping, zooming, and lighting adjustments enable models to better recognize common visual variations encountered in real-world scenarios[15]. By enriching the visual representation in the training data, augmentation enables deep learning models to learn more generalized features, thereby maintaining stable performance when exposed to varying or suboptimal visual conditions[14], [27]. Mathematically, augmentation transformations such as image rotation can be described using the concept of two-dimensional linear transformations. The rotation of a point (x,y) in the image plane around a center point (typically the image center) by an angle θ can be computed using the following rotation matrix:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (1)$$

Where:

(x, y) : represents the original coordinates of the pixel
(x', y') : represents the new coordinates after the rotation is applied
θ : rotation angle in radians (positive values indicate counterclockwise rotation)

In the context of augmentation, this formula is used to generate variations in image orientation, enabling deep learning models to learn visual patterns from multiple perspectives. According to Gonzalez and Woods [28], Such transformations are essential for enriching the visual distribution of training data, particularly when images are obtained from uncontrolled environments. The implementation of this rotation formula has been widely adopted in libraries such as OpenCV and TensorFlow, where it forms an integral part of automated augmentation pipelines.

1.2.4. Normalization

Images typically contain pixel values ranging from 0 to 255. To align with the expectations of deep learning models, these values are normalized to a smaller range, commonly between 0 and 1 or standardized with a mean and standard deviation. This preprocessing step is crucial, as it accelerates model convergence during training and helps prevent issues arising from differences in data scale [29]-[30].

Most deep learning models such as VGG16, ResNet50, DenseNet121, MobileNetV3, and EfficientNetV2B3 utilize the preprocess_input function to normalize input images. This function adjusts pixel values according to the specific preprocessing standards used during the training of each model, ensuring compatibility and optimal performance during inference or fine-tuning [31]-[32],. This approach ensures that the distribution of input image data closely matches the distribution used during the model's original training, thereby facilitating more effective learning and improving convergence stability.

1.2.5. Balancing and SMOTE

An imbalanced class distribution often leads models to favor the majority class while neglecting minority classes. This bias adversely affects performance metrics such as recall and F1-score, resulting in

suboptimal generalization for underrepresented categories [33]-[34]. To address class imbalance, which can lead to bias toward the majority class, it is essential to apply balancing strategies during training. In this study, data balancing was first performed through image augmentation. This approach aimed not only to enrich the visual diversity of the training data, making the model more robust to environmental variations, but also to equalize the number of samples across classes. As a second strategy, the Synthetic Minority Over-sampling Technique (SMOTE) was employed to generate synthetic samples for minority classes by interpolating between feature vectors, thereby improving class representation without direct duplication [33]. SMOTE does not operate directly on raw image data, but rather on feature vectors extracted from the convolutional neural network (CNN). The technique generates new synthetic samples within the minority feature space by interpolating between neighboring vectors. This results in semantically meaningful representations that preserve the underlying characteristics of the minority class. Mathematically, the SMOTE interpolation process is formulated as follows [17]

$$x_{\text{new}} = x_{\text{i}} + \delta \,.\, (x_{\text{nn}} - \ x_{\text{i}}) \qquad (2)$$

Where:

xI = a feature vectors of two close instances from the minority class
$x$nn = a feature vector from one of the *k*-nearest neighbors of xix_ixi
δ = a random number in the range 0 and 1
$x$new = the newly synthesized data point

Recent approaches have even integrated SMOTE with contrastive learning[18] resulting in significant improvements in classification accuracy on imbalanced datasets.

1.2.6. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a class of deep learning architectures particularly effective for image recognition and visual analysis tasks, including image classification, object detection, and image segmentation. A typical CNN consists of three main components: convolutional layers, pooling layers, and fully connected layers, which work sequentially to extract hierarchical features from input data such as images. This technology has been widely applied in the field of plant disease identification, including the classification of diseases in snake gourd (Trichosanthes cucumerina) leaves [35]. Various popular CNN architectures offer distinct advantages; for instance, EfficientNetV2B3 is known for its computational efficiency while maintaining relatively high classification accuracy [36], MobileNetV3-Large achieves relatively high accuracy among lightweight models and is specifically optimized for deployment on mobile and edge devices [37], VGG-16, though architecturally simple, remains effective for feature extraction due to its deep and uniform convolutional structure [38], ResNet50 addresses the problem of accuracy degradation in deep networks by employing residual connections, which facilitate more effective gradient flow during training [32], and DenseNet121 enhances information flow and parameter efficiency by connecting each layer to all preceding layers, enabling feature reuse and mitigating the vanishing gradient problem [39].

2. RESEARCH METHODOLOGY

2.1. Dataset

This study utilizes a dataset specifically developed to capture the complexity of potato leaf diseases under real-world uncontrolled field conditions. A total of 3,076 high-resolution images were collected in August 2023 from potato farms located in three highland regions of Central Java, Indonesia; Wonosobo,

Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

Banjarnegara, and Magelang. The images were acquired using a range of smartphone cameras, contributing to considerable variability in lighting, perspective, and background composition. Expert annotators from the Department of Plant Protection conducted image validation and manually categorized the dataset into seven classes: virus, Phytophthora, nematode, fungus, bacteria, pest, and healthy. A key strength of this dataset lies in its realistic representation of complex field conditions, including visual noise, diverse backgrounds, and random leaf orientations. These characteristics make it more representative of real-world scenarios compared to widely used datasets available on platforms such as Kaggle, particularly PlantVillage and PLD which were collected under controlled laboratory environments.

2.2. Research Workflow

The research workflow is illustrated in the diagram as follows

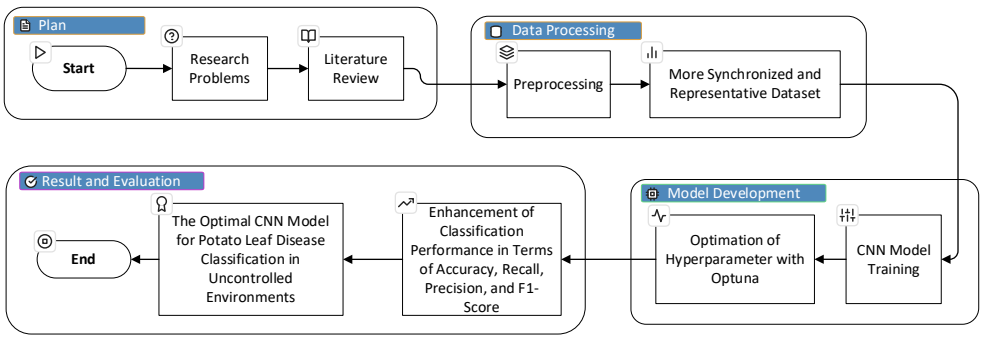

Fig 1. Research Workflow

This research workflow consists of four main stages: planning, data processing, model development, and results and evaluation. The planning stage includes problem formulation and literature review as the foundation for defining the research direction. Next, the data processing stage involves preprocessing, augmentation, and class balancing to obtain a more representative dataset. The model development stage includes training several CNN architectures and performing hyperparameter optimization using Optuna to determine the best configuration. Finally, the evaluation stage is conducted to measure performance improvements using accuracy, precision, recall, and F1-score metrics, resulting in an optimal CNN model for potato leaf disease classification under uncontrolled environmental conditions.

2.3. Proposed Method

To provide a comprehensive understanding of the potato leaf disease classification process, the following flowchart illustrates the key stages involved in the proposed system architecture. An overview of the proposed method is presented in the following figure;

The workflow begins with dataset acquisition from public sources, followed by a preprocessing phase that includes both data augmentation to increase visual diversity and normalization to standardize input values. Once preprocessing is completed, the dataset is split into training (80%), validation (10%), and testing (10%) subsets. Augmentation techniques are applied exclusively to the training set. The augmented training data is then used to train several CNN architectures namely EfficientNetV2B3, MobileNetV3-Large, VGG-16, ResNet50, and DenseNet121 to evaluate their performance in image classification. After identifying the architecture with the most promising results, Optuna is employed to perform automated hyperparameter optimization specifically on the best-performing model. Final evaluation is conducted using a confusion matrix and accuracy plots, leading to the selection of the most accurate and efficient image classification model for the given dataset.

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

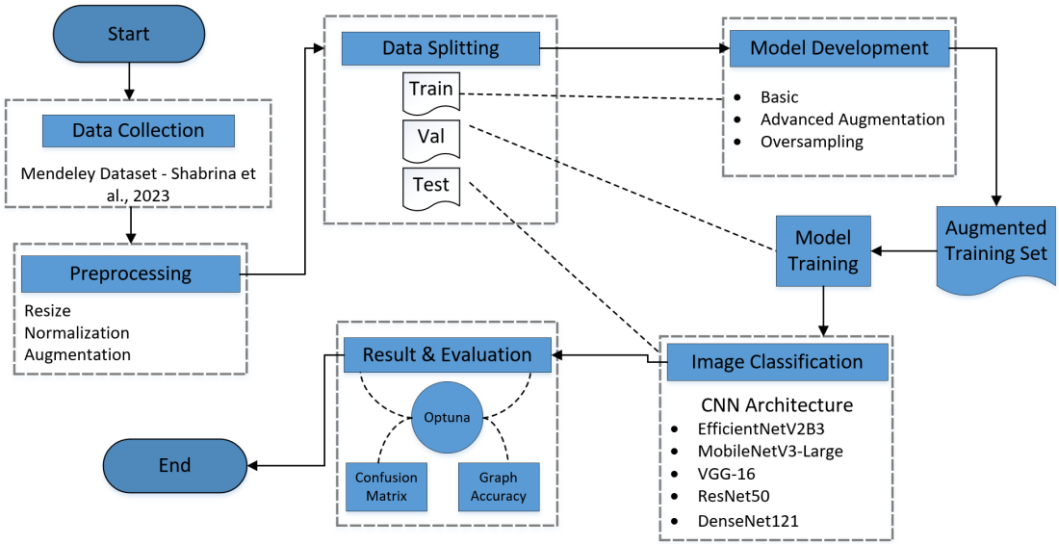

Fig 2. Proposed Method

2.4. Data Preprocessing

All images, regardless of their assignment to training, validation, or testing sets were resized to a fixed resolution of 224 × 224 pixels. Pixel values were then converted to float32 and normalized using the preprocess_input function from the Keras library, which aligns with the preprocessing scheme used during pre-training on the ImageNet dataset. This step is crucial to ensure compatibility between the input data distribution and the pre-trained weights utilized in the transfer learning process. To further enrich the dataset and improve model generalization, both geometric and photometric transformations were applied as part of data augmentation. These augmentations were performed prior to the data split, ensuring consistent variation across the training, validation, and test sets. Examples of the augmented images are presented in the following figure:

Fig 3. Original and Augmented Images of Nematode-Affected Plant Leaves

The augmentation techniques employed in this study include random horizontal flipping, rotations ranging from 90° to 270°, as well as controlled adjustments to brightness and contrast. These transformations introduce spatial variability and enhance the model's ability to recognize disease patterns despite changes in viewpoint or lighting conditions. To maintain numerical stability during training, all augmented outputs were subsequently clipped within a valid pixel value range.

2.5. Strategy Development

2.5.1. Scenario 1 : Basic Augmentation

In the first scenario, image augmentation was applied during the preprocessing stage, prior to splitting the dataset into training, validation, and test subsets. Augmentation was selectively performed on

Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

underrepresented classes that contained fewer images than the predefined target of 340 images per class. For instance, as illustrated in Figure 3, the "Nematode" class, which originally contained only 68 images, was augmented fivefold to reach the minimum target of 340 images. After augmentation, the dataset was split, with 80% allocated for training. Consequently, minority classes such as "Nematode" and "Healthy" (label number 2 and 5) were expanded to achieve the following class distribution as follows:

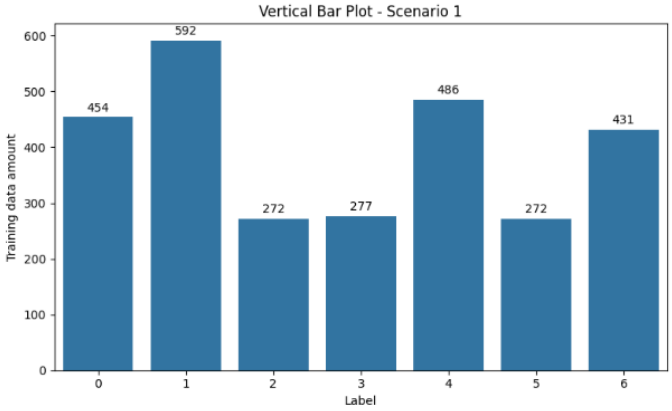

Fig 4. Training Data on Scenario 1

All augmentation processes were performed using the TensorFlow library to maintain valid pixel values and visual realism. The augmented images were saved in their respective class directories with unique filenames to prevent overwriting the original data. The final dataset consisted of 2,784 training samples and was used without additional augmentation, although class distribution remained imbalanced.

2.5.2. Scenario 2 : Advanced Augmentation

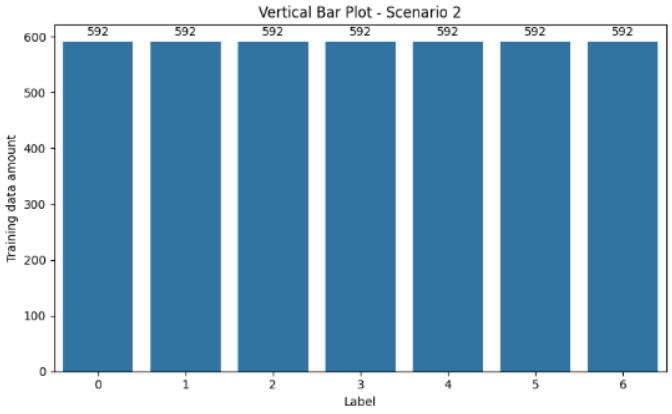

Fig 5. Training Data on Scenario 2

In the second scenario, augmentation was applied twice: initially during the preprocessing stage, and subsequently on the training subset after the dataset was split. This extended augmentation strategy aimed to address class imbalance by employing a function named apply_augmentation_balance. The process began by calculating the number of images in each class and identifying the class with the fewest samples. For underrepresented classes, additional augmented images were generated and added until the number of images in each class matched that of the class with the highest sample count. The visualization is presented in the following figure:

Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

As a result, each class was expanded to contain 592 images, indicating that the extended augmentation was carried out until class distribution became balanced, as illustrated in Fig. 5. This brought the total number of training images to 4,144. Such an approach effectively reduces model bias toward majority classes, while the validation and testing sets remained unaltered. Maintaining the integrity of the validation and testing data is crucial to ensure objective evaluation of model performance.

2.5.3. Scenario 3 : Oversampling with SMOTE

In the third scenario, the Synthetic Minority Over-sampling Technique (SMOTE) was applied exclusively to the training subset. Since SMOTE is inherently designed for two-dimensional tabular data, it cannot be directly applied to image datasets in tensor format such as tf.data.Dataset. Therefore, each image was first converted into a flattened numerical array, transforming the dataset into a two-dimensional matrix where each row represents an image and each column corresponds to a feature. Once the dataset was in tabular format, SMOTE was utilized to generate synthetic samples for the minority classes by interpolating between similar data points. The resulting balanced dataset comprising both original and synthetic feature arrays with their corresponding labels was then reshaped back into image format to align with the expected input structure of the CNN models. The visualization is presented in the following figure:

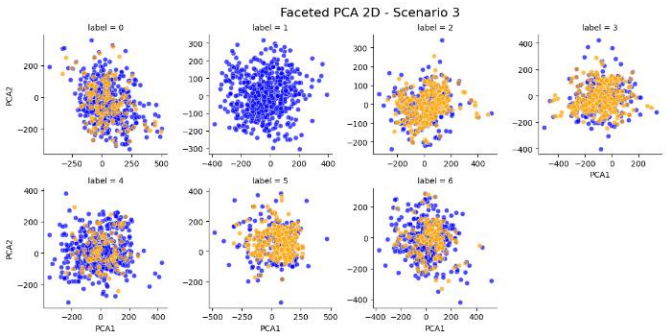

Fig 6. Visualization of Original and Synthetic Classes

The reshaped synthetic data were then integrated with the original training data and used collectively during the model training process.

2.6. Architecture Model

The classification model in this study was developed using a transfer learning approach, employing five distinct convolutional architectures: EfficientNetV2B3, MobileNetV3-Large, VGG-16, ResNet50, and DenseNet121. The selection of the five architectures was based on their diverse characteristics and varying levels of complexity, ranging from classical deep models such as VGG-16, ResNet50, and DenseNet121 to modern and computationally efficient architectures such as EfficientNetV2B3 and MobileNetV3-Large. Among these architectures, MobileNetV3-Large was selected as the primary model for optimization due to its balanced trade-off between accuracy, computational efficiency, and lightweight design. This architecture is specifically engineered to perform effectively on resource-constrained devices while maintaining high representational capacity. Its design incorporates squeeze-and-excitation blocks and the hard-swish activation function, enabling adaptive feature extraction and improved robustness against variations in lighting and background conditions commonly found in field-captured images.

For each architecture, the initial layers (base models) were frozen to preserve the pre-trained weights from ImageNet, allowing training to focus solely on the newly added classification layers. The added structure comprised a Global Average Pooling 2D layer to reduce spatial dimensions while retaining global

feature context, followed by a dense layer with 256 units and ReLU activation. To mitigate overfitting, a Dropout layer with a rate of 0.3 was incorporated. The output layer consisted of seven neurons corresponding to the number of target classes, with a softmax activation function to generate probabilistic outputs. The entire architecture was implemented using the Keras Sequential API, offering high flexibility for training and further hyperparameter tuning.

2.7. Optimizer and Evaluation

Model performance was comprehensively evaluated using both quantitative metrics and visual analysis. The training process employed the Adam optimizer, an adaptive optimization algorithm that estimates the first and second moments of the gradients to achieve efficient and stable convergence[40], In addition, the loss function used was sparse_categorical_crossentropy, which is suitable for multi-class classification problems with integer-encoded labels. To prevent overfitting, an EarlyStopping callback was implemented to halt training if the validation accuracy did not improve for five consecutive epochs. In addition, the ModelCheckpoint callback was employed to automatically save the model with the highest validation accuracy during training.

After training, the model was evaluated on the test set using a confusion matrix and evaluation metrics including precision, recall, and F1-score, all of which were summarized in a comprehensive classification report. These metrics provide insights into the model's performance on each class and are computed using the following formulas [41]:

$$\text{Test Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (4)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (5)$$

$$\text{F1-Score} = 2 \text{ x } \frac{(\text{Recall x Precision})}{(\text{Recall + Precision})} \quad (6)$$

All evaluation results were visualized to provide deeper insights into the model's learning dynamics and classification performance. Accuracy and loss curves were plotted for each training epoch to illustrate the convergence behavior and potential overfitting or under fitting trends. Additionally, a heat map of the confusion matrix was generated to offer a visual representation of class-wise prediction performance.

3. RESULT AND DISCUSSIONS

3.1. Result

The classification models developed using various CNN architectures demonstrated highly competitive performance. The results indicate that the first scenario yielded reasonably strong performance, as presented in row of Scenario 1. In the second scenario, overall model performance improved compared to previous findings outlined in the Related Work Table, demonstrating the effectiveness of additional augmentation in addressing class imbalance. The third scenario consistently produced enhancements in both accuracy and F1-score across nearly all tested CNN architectures. Among the five architectures, MobileNetV3-Large achieved the best overall performance in all scenarios, with its highest scores observed in the third scenario, recording an accuracy of 0.863248 and an F1-score of 0.867725.

The application of data augmentation was shown to enhance model generalization by enriching the visual variability of leaf images, thereby reducing dependency on specific lighting conditions or

Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

orientations. However, in Scenario 2, performance decreased due to excessive augmentation, which introduced over-regularization and feature distortion, making it difficult for the model to extract salient features effectively. Performance improvements were observed again in Scenario 3 when augmentation was combined with SMOTE, as this combination balanced the class distribution without introducing excessive noise, resulting in the most stable configuration across all CNN architectures. The experimental results for Scenarios 1, 2, and 3 are summarized in the following table:

Table 2. Experimental Results of CNN Architectures Across Three Training Scenarios

Scenario	CNN Architectures	Accuracy	Precision	Recall	F1-Score
1	DenseNet121	0,846154	0,847769	0,858316	0,851515
	EfficientNetV2B3	0,849003	0,855365	0,859009	0,854645
	MobileNetV3Large	0.863248	0.865010	0.870416	0.866260
	ResNet50	0,826211	0,835594	0,842334	0,837386
	VGG16	0,82906	0,828139	0,849646	0,835447
2	DenseNet121	0,797721	0,802026	0,816346	0,805882
	EfficientNetV2B3	0,789174	0,799951	0,833184	0,808446
	MobileNetV3Large	0,792023	0,811897	0,818205	0,813206
	ResNet50	0,823362	0,84679	0,851493	0,845817
	VGG16	0,757835	0,786107	0,792084	0,778887
3	DenseNet121	0,843305	0,847425	0,852917	0,849538
	EfficientNetV2B3	0,840456	0,863029	0,850758	0,851684
	MobileNetV3Large	**0,863248**	**0,871107**	**0,87319**	**0,867725**
	ResNet50	0,826211	0,836787	0,848825	0,842022
	VGG16	0,82906	0,838593	0,853158	0,843411

Scenario 3 demonstrated the highest effectiveness in enhancing model generalization, addressing class imbalance, and producing more accurate predictions. Evaluation metrics such as precision, recall, and F1-score showed significant improvements in the third scenario compared to the second, which did not utilize SMOTE. The following figure illustrates the training and validation accuracy and loss of the best-performing model.

Each model was trained for a maximum of 100 epochs, with EarlyStopping applied to halt training when validation accuracy ceased to improve for five consecutive epochs. The varying stopping points across architectures indicate stable convergence behavior. Notably, no significant signs of overfitting were observed, suggesting that the training process was well-regularized and the models generalized effectively to unseen data. The following figure illustrates the training and validation accuracy and loss of the best-performing model.

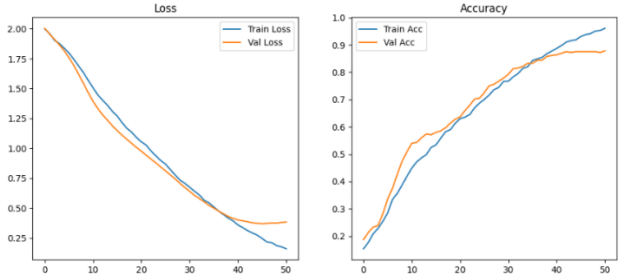

Fig 7. Accuracy and Loss Graph on MobileNetV3L Scenario 3

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

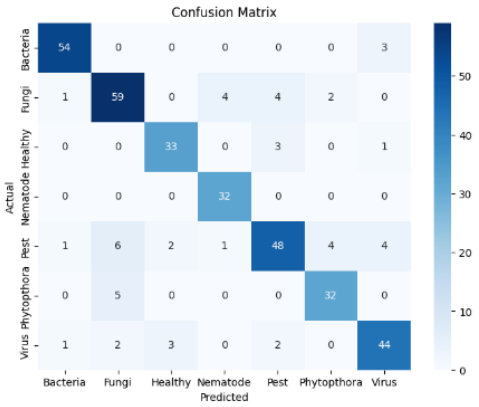

Fig 8. Confusion Matrix on scenario 3 - MobileNetV3-Large

The confusion matrix analysis in Scenario 3 further reinforces the effectiveness of this approach. As illustrated in the figure 8, the majority of model predictions align along the main diagonal, indicating accurate classification. Categories such as *Fungi*, *Bacteria*, and *Healthy* exhibit high classification accuracy, with 59, 54, and 33 correctly predicted samples, respectively. However, some misclassifications persist, particularly within the *Pest* class, which tends to overlap with visually similar classes such as *Fungi* and *Phytophthora*. This suggests that although SMOTE successfully enhances sensitivity toward minority classes, classification challenges remain for categories with overlapping visual characteristics. Nevertheless, the model generally demonstrates strong generalization capabilities on real-world data, with a well-distributed set of predictions and minimal bias toward majority classes.

3.2. Discussion

Based on the experimental results presented in Table Scenario 3, the MobileNetV3-Large architecture was selected as the feature extractor due to its superior classification accuracy. Subsequently, hyperparameter optimization was conducted using Optuna to identify the optimal combination of parameters, including learning rate, dropout rate, and the number of units in the dense layer. The visualization results are presented in the following figure.

The visualization of the tuning results indicates that dropout_rate3 and dropout_rate2 contributed most significantly to the model's performance, followed by dense_units1. The optimized parameters included a learning rate ranging from *1e-6 to 1e-3*, dropout rates (dropout_rate1–3) between *0.2 and 0.5*, and dense layer units (dense_units1–3) between *128 and 512*. The primary objective of the optimization process was to maximize the weighted F1-score on the validation set. The Optimization History Plot demonstrated a steady increase in F1-score across trials, reaching its peak performance of 0.9139 at the 11th trial.

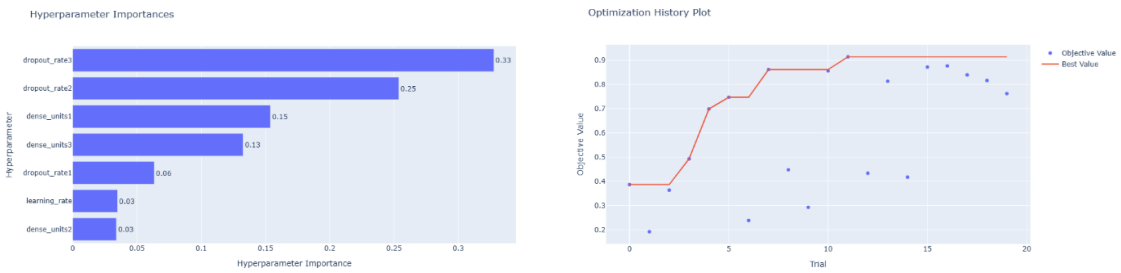

Fig 9. Hyperparameter Importance & Optimization History Plot: Objective and Best Values

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

After identifying the optimal configuration, the model was retrained using the best combination of hyperparameters. The training results demonstrated a sharply increasing and stable accuracy curve, with training accuracy approaching perfection and validation accuracy reaching nearly 90% by the final epoch. The implementation of MobileNetV3-Large following Optuna-based tuning yielded a significant improvement in model performance, highlighting the effectiveness of automated hyperparameter optimization in enhancing classification accuracy. The following figure shows the accuracy trends and comparison among models

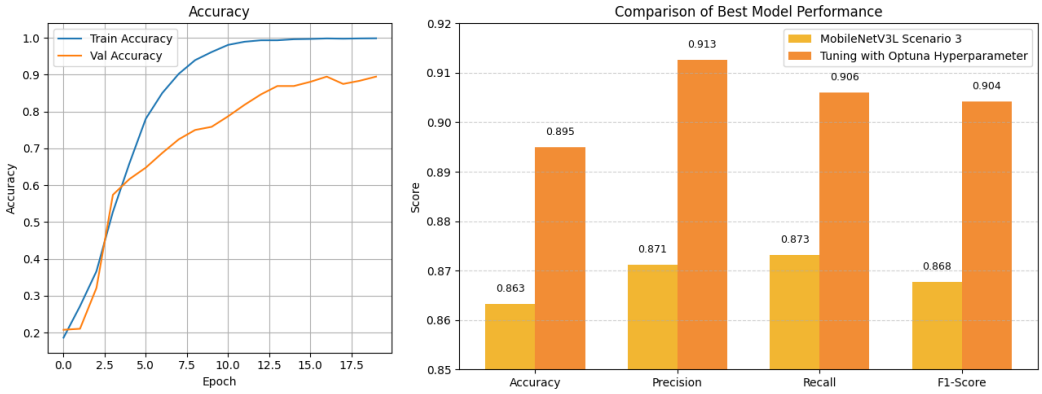

Fig 1. Accuracy and Comparison of Best Model Performance

The graph illustrates a notable improvement across four evaluation metrics; Accuracy, Precision, Recall, and F1-Score following hyperparameter tuning. Specifically, Accuracy increased from 0.863 to 0.895, Precision from 0.871 to 0.913, Recall from 0.873 to 0.906, and F1-Score from 0.868 to 0.904. These results demonstrate that tuning the best model using Optuna Hyperparameter optimization can improve the accuracy of the model in real-world image classification.

Table 3. Comparison of Results with State-of-the-Art Methods on Same Dataset

Model	Accuracy	Data Augmentation Technique	Balancing	Hyperparameter Optimization	Year, Ref
EfficientNet V2B3	73.63 %	Flipping, rotation, brightness	None (An experiment with 990–1,000 images per class was performed but resulted in lower accuracy)	Manual (Adam, LR=0.001)	2024, [9]
DSCSkipNet	80.45 %	Flipping, rotation, crop	None	Manual	2024, [20]
EfficientNet-LITE	87.82 %	Rotation, width/ height shift, zoom, horizontal flip, Sobel edge augmentation	None	Kernel Ensemble SVM (Linear+ Poly+RBF+Sigmoid)	2025, [19]
MobileNetV3-Large	89.50 %	Flipping, rotation, brightness, contrast	SMOTE	Optuna (Dropout, Dense, Learning Rate)	2025, Proposed Method

Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

In addition to the CNN architectural models presented in Table 3, the accuracy improvement of up to 89.50% in this study demonstrates the effectiveness of combining appropriate data augmentation strategies, class balancing using SMOTE, and hyperparameter optimization with Optuna. The results outperform previous studies that utilized the same dataset but did not incorporate adaptive optimization and synthetic balancing techniques.

However, the comparison results also reveal that excessive data augmentation can produce negative effects such as over-regularization or feature distortion, in which the model loses sensitivity to essential features because the augmented visual patterns deviate too far from the original image distribution. This observation explains why several previous studies experienced decreased accuracy when the quantity or variation of augmentation was not properly controlled.

These findings highlight that the performance of CNN models is strongly influenced by augmentation strategies and tuning processes that must be aligned with the complexity of real-world field data. Scientifically, this research carries significant relevance to the fields of Informatics and Artificial Intelligence, particularly in the application of computer vision for automatic detection in smart agriculture, where classification serves as a fundamental component in developing adaptive, accurate, and robust diagnostic systems capable of operating under uncontrolled environmental conditions.

4. CONCLUSION

The developed model employing various architectures demonstrated competitive classification performance with consistent results in terms of both accuracy and F1-score, where the highest accuracy in previous research was 0.736, as shown in Table 1. Experimental results across three scenarios; basic augmentation, advanced augmentation, and oversampling using SMOTE indicated a significant improvement in the third scenario, particularly in enhancing model generalization and addressing class imbalance. MobileNetV3-Large emerged as the best-performing architecture across all scenarios, achieving the highest performance in the third scenario with an accuracy of 0.863.

Further optimization using Optuna led to a notable improvement in evaluation metrics, where the accuracy of the MobileNetV3-Large model increased from 0.863 to 0.895, accompanied by improvements in precision, recall, and F1-score as shown in Fig 10. Hyperparameter tuning effectively enhanced the model's generalization ability, making it more accurate and reliable for classifying potato leaf images in uncontrolled environments. Overall, the integrated approach combining data augmentation, SMOTE, and hyperparameter optimization via Optuna proved to be effective in improving the model's performance and reliability in image classification tasks.

The findings of this study underscore the importance of implementing automated optimization and data-driven learning strategies in the fields of Informatics and Artificial Intelligence, particularly for computer vision applications in smart agriculture. The proposed framework not only successfully enhances the accuracy and robustness of potato leaf disease classification under uncontrolled environmental conditions but also establishes a foundation for developing intelligent and adaptive diagnostic systems capable of supporting real-world decision-making processes. Thus, this research contributes both methodologically and practically to the advancement of AI-based image analysis and intelligent agriculture systems.

However, this study still presents several limitations. Although the dataset used is diverse, it is restricted to a single plant species, limiting the model's generalization capability for other crops or broader environmental conditions. Moreover, the augmentation techniques applied were limited to basic geometric and photometric transformations, without exploring more advanced methods such as synthetic image generation using Generative Adversarial Networks (GANs) or domain randomization approaches.

For future work, it is recommended to integrate more sophisticated augmentation techniques, evaluate additional CNN architectures or explore Vision Transformers, and expand the dataset with real-time field images collected from different regions and seasons. Furthermore, deeper exploration of the hyperparameter search space using Optuna is necessary to identify more optimal parameter combinations. The use of leaf segmentation techniques is also suggested to improve feature localization and overall classification accuracy. These enhancements are expected to further strengthen the model's resilience and adaptability for broader applications in plant disease classification.

REFERENCES

[1] D. Sahara, Munir Eti Wulanjari, and J. Triastono, "Optimasi Penggunaan Input Produksi pada Usahatani Kentang di Dataran Tinggi Dieng, Jawa Tengah," *Jurnal Ilmu Pertanian Indonesia*, vol. 28, no. 4, pp. 612–619, Jul. 2023, doi: 10.18343/jipi.28.4.612.

[2] International Potato Center (CIP), "Late Blight: The Plant Disease that Changed History," International Potato Center. Accessed: Oct. 14, 2025. [Online]. Available: https://cipotato.org/late-blight/

[3] I. Pacal *et al.*, "A systematic review of deep learning techniques for plant diseases," *Artif Intell Rev*, vol. 57, no. 11, Nov. 2024, doi: 10.1007/s10462-024-10944-7.

[4] S. Shakeel, "Role of Artificial Intelligence in Plant Disease Detection: A Review," *Emerging Research Nexus*, vol. 2, no. 1, Jan. 2025, doi: 10.70788/ern.2.1.2025.11.

[5] A. Upadhyay *et al.*, "Deep learning and computer vision in plant disease detection: a comprehensive review of techniques, models, and trends in precision agriculture," *Artif Intell Rev*, vol. 58, no. 3, Mar. 2025, doi: 10.1007/s10462-024-11100-x.

[6] S. Wang *et al.*, "Advances in Deep Learning Applications for Plant Disease and Pest Detection: A Review," Feb. 01, 2025, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/rs17040698.

[7] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Front Plant Sci*, vol. 7, no. September, Sep. 2016, doi: 10.3389/fpls.2016.01419.

[8] J. Rashid, I. Khan, G. Ali, S. H. Almotiri, M. A. Alghamdi, and K. Masood, "Multi-level deep learning model for potato leaf disease recognition," *Electronics (Switzerland)*, vol. 10, no. 17, Sep. 2021, doi: 10.3390/electronics10172064.

[9] N. H. Shabrina *et al.*, "A novel dataset of potato leaf disease in uncontrolled environment," *Data Brief*, vol. 52, Feb. 2024, doi: 10.1016/j.dib.2023.109955.

[10] J. Liu and X. Wang, "Plant diseases and pests detection based on deep learning: a review," Dec. 01, 2021, *BioMed Central Ltd*. doi: 10.1186/s13007-021-00722-9.

[11] M. Ahmad, M. Abdullah, H. Moon, and D. Han, "Plant Disease Detection in Imbalanced Datasets Using Efficient Convolutional Neural Networks with Stepwise Transfer Learning," *IEEE Access*, vol. 9, pp. 140565–140580, 2021, doi: 10.1109/ACCESS.2021.3119655.

[12] T. Miftahushudur, H. M. Sahin, B. Grieve, and H. Yin, "A Survey of Methods for Addressing Imbalance Data Problems in Agriculture Applications," *Remote Sens (Basel)*, vol. 17, no. 3, Feb. 2025, doi: 10.3390/rs17030454.

[13] N. Upadhyay and N. Gupta, "Potato Leaves Disease Detection with Data Augmentation Using Deep Learning Approach," in *Lecture Notes in Networks and Systems*, Springer Science and Business Media Deutschland GmbH, 2023, pp. 589–599. doi: 10.1007/978-981-19-9638-2_51.

[14] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *J Big Data*, vol. 6, no. 1, Dec. 2019, doi: 10.1186/s40537-019-0197-0.

[15] S. Yang, W. Xiao, M. Zhang, S. Guo, J. Zhao, and F. Shen, "Image Data Augmentation for Deep Learning: A Survey," Apr. 2022, [Online]. Available: http://arxiv.org/abs/2204.08610

[16] A. Fernández, S. García, F. Herrera, and N. V Chawla, "SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary," 2018.

[17] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," 2002.

[18] X. Gao, N. Jamil, and M. I. Ramli, "CL-SR: Boosting Imbalanced Image Classification with Contrastive Learning and Synthetic Minority Oversampling Technique Based on Rough Set Theory Integration," *Applied Sciences (Switzerland)*, vol. 14, no. 23, Dec. 2024, doi: 10.3390/app142311093.

[19] G. Sangar and V. Rajasekar, "Optimized classification of potato leaf disease using EfficientNet-LITE and KE-SVM in diverse environments," *Front Plant Sci*, vol. 16, 2025, doi: 10.3389/fpls.2025.1499909.

[20] G. Boukhlifa and Y. Chibani, "DSCSkipNet: An Accuracy-Complexity Trade-off for Effective Potato Disease Identification in Uncontrolled Environments," in *Proceedings of EDiS 2024 - 2024 4th International Conference on Embedded and Distributed Systems*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 267–272. doi: 10.1109/EDiS63605.2024.10783376.

[21] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," Jul. 2019, [Online]. Available: http://arxiv.org/abs/1907.10902

[22] L. H. Lai *et al.*, "The Use of Machine Learning Models with Optuna in Disease Prediction," *Electronics (Switzerland)*, vol. 13, no. 23, Dec. 2024, doi: 10.3390/electronics13234775.

[23] S. Karthikeyan, R. Charan, S. Narayanan, and L. Jani Anbarasi, "Enhanced plant disease classification with attention-based convolutional neural network using squeeze and excitation mechanism," *Front Artif Intell*, vol. 8, 2025, doi: 10.3389/frai.2025.1640549.

[24] R. I. Hasan, S. M. Yusuf, and L. Alzubaidi, "Review of the state of the art of deep learning for plant diseases: A broad analysis and discussion," Oct. 01, 2020, *MDPI AG*. doi: 10.3390/plants9101302.

[25] J. G. A. Barbedo, "Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification," *Comput Electron Agric*, vol. 153, pp. 46–53, Oct. 2018, doi: 10.1016/j.compag.2018.08.013.

[26] Z. Salman, A. Muhammad, and D. Han, "Plant disease classification in the wild using vision transformers and mixture of experts," *Front Plant Sci*, vol. 16, 2025, doi: 10.3389/fpls.2025.1522985.

[27] L. Perez and J. Wang, "The Effectiveness of Data Augmentation in Image Classification using Deep Learning," Dec. 2017, [Online]. Available: http://arxiv.org/abs/1712.04621

[28] R. C. . Gonzalez and R. E. . Woods, *Digital image processing*. Pearson, 2018.

[29] N. Parashar *et al.*, "Enhanced residual-attention deep neural network for disease classification in maize leaf images," *Sci Rep*, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s41598-025-14726-1.

[30] D. W. Girmaw, A. O. Salau, B. S. Mamo, and T. L. Molla, "A novel deep learning model for cabbage leaf disease detection and classification," *Discover Applied Sciences*, vol. 6, no. 10, Oct. 2024, doi: 10.1007/s42452-024-06233-1.

[31] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," Oct. 2016, [Online]. Available: http://arxiv.org/abs/1610.02357

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition." [Online]. Available: http://image-net.org/challenges/LSVRC/2015/

[33] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, vol. 106, pp. 249–259, Oct. 2018, doi: 10.1016/j.neunet.2018.07.011.

[34] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *J Big Data*, vol. 6, no. 1, Dec. 2019, doi: 10.1186/s40537-019-0192-5.

[35] J. Lu, L. Tan, and H. Jiang, "Review on convolutional neural network (CNN) applied to plant leaf disease classification," Aug. 01, 2021, *MDPI AG*. doi: 10.3390/agriculture11080707.

[36] M. Tan and Q. V. Le, "EfficientNetV2: Smaller Models and Faster Training," Apr. 2021, [Online]. Available: http://arxiv.org/abs/2104.00298

[37] A. Howard *et al.*, "Searching for MobileNetV3," May 2019, [Online]. Available: http://arxiv.org/abs/1905.02244

[38] K. Simonyan and A. Zisserman, "VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION," 2015. [Online]. Available: http://www.robots.ox.ac.uk/

Vol. 7, No. 2, April 2026, Page. 743-759
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.4898

[39] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," Aug. 2016, [Online]. Available: http://arxiv.org/abs/1608.06993
[40] D. P. Kingma and J. Lei Ba, "ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION."
[41] F. Arshad *et al.*, "PLDPNet: End-to-end hybrid deep learning framework for potato leaf disease prediction," *Alexandria Engineering Journal*, vol. 78, pp. 406–418, Sep. 2023, doi: 10.1016/j.aej.2023.07.076.