

Tomato leaf disease classification using DenseNet-121 with data augmentation and fine-tuning

Sufajar Butsianto, Anggi Muhammad Rifa'i

Department of Informatics Engineering, Faculty of Engineering, Pelita Bangsa University, Bekasi, Indonesia

Article Info

Article history:

Received Dec 1, 2024

Revised May 3, 2026

Accepted May 11, 2026

Keywords:

Agricultural images

Data augmentation

DenseNet-121

Fine-tuning

Tomato leaf disease detection

ABSTRACT

In recent years, accurate classification of agricultural images has become increasingly important to support precision farming and crop disease monitoring. However, achieving reliable performance remains challenging due to visual similarity between disease classes and dataset variability. This study presents an applied evaluation of DenseNet-121 combined with data augmentation and fine-tuning for multi-class tomato leaf disease classification. Experiments were conducted using a publicly available tomato leaf image dataset consisting of 5,000 images across 10 classes. All images were resized to 64×64 pixels and split into 80% training and 20% testing sets using a stratified strategy. Data augmentation was applied exclusively to the training data to improve generalization. The experimental results show a progressive performance improvement across training stages, achieving a final classification accuracy of 98.44% with a loss of 4.72% after fine-tuning. Per-class evaluation indicates strong performance across most disease categories, with minor confusion observed among visually similar classes. While the results demonstrate the effectiveness of the proposed training strategy under controlled experimental conditions, further validation using real-field images is required. Overall, this study shows the potential of DenseNet-121 with transfer learning to support tomato leaf disease classification in precision agriculture applications.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Sufajar Butsianto

Department of Informatics Engineering, Faculty of Engineering, Pelita Bangsa University

South Cikarang, Bekasi, West Java, Indonesia

Email: sufajar@pelitabangsa.ac.id

1. INTRODUCTION

The agricultural sector is rapidly evolving with technological advancements aimed at improving efficiency and sustainability, yet accurate agricultural image classification remains a major challenge [1]–[4]. Variations in environmental conditions, crop diversity, image quality, and subtle disease symptoms significantly complicate classification tasks, often leading to misclassification that can result in incorrect treatment decisions, economic losses, and reduced trust in precision agriculture systems [5]–[10]. These challenges are further intensified by noisy, redundant, and limited datasets, highlighting the need for robust and adaptable classification approaches. Deep learning models, particularly convolutional neural networks (CNN) such as residual network (ResNet), visual geometry group (VGG), and Inception, have demonstrated strong performance in image classification by automating feature extraction [11]–[13]. However, their effectiveness often declines when applied to highly variable agricultural datasets due to feature redundancy, class overlap, limited labeled data, and overfitting issues [14]–[16]. DenseNet-121 addresses these limitations through efficient feature reuse, improved gradient flow, and reduced parameter complexity, making it well-suited for heterogeneous agricultural data [17]–[19].

This study evaluates the application of DenseNet-121 combined with data augmentation and fine-tuning strategies for tomato leaf disease classification using the FarmVillage dataset [20]–[22]. Rather than proposing a new architecture, the research focuses on empirically analyzing how preprocessing and transfer learning stages enhance classification performance under realistic agricultural conditions. The results demonstrate strong adaptability to dataset variability and inter-class similarity, although further validation using real-field images is necessary to confirm robustness in practical farming environments.

2. METHOD

The proposed methodology for tomato leaf disease detection using DenseNet-121 with data augmentation and fine-tuning consists of five main stages, as illustrated in Figure 1, data collection, exploratory data analysis (EDA), data preprocessing, model development, and model evaluation [23]. The process begins with systematic data collection, followed by dataset analysis to determine appropriate preprocessing and modeling strategies. Data preprocessing is then applied to ensure consistency and quality. Subsequently, the DenseNet-121 model is trained and tested, and the results are comprehensively evaluated to assess detection performance and model robustness. This structured workflow ensures systematic experimentation and enables reproducibility of the proposed approach.

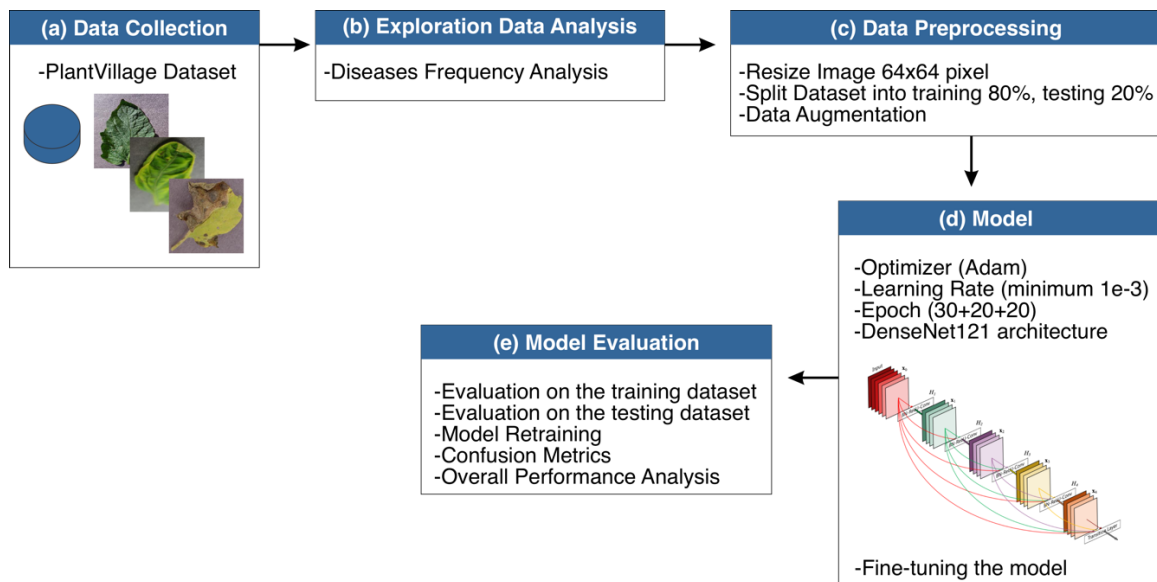


Figure 1. Methodology of the proposed DenseNet-121 with data augmentation

2.1. Data collection

The dataset utilized in this research was sourced from [24], the PlantVillage repository shown in Figure 2, which is widely recognized for its comprehensive and high-quality plant disease image collections. Figure 2(a) shows tomato target spot, Figure 2(b) shows tomato mosaic virus, Figure 2(c) shows tomato yellow leaf curl virus, Figure 2(d) shows tomato bacterial spot, Figure 2(e) shows tomato early blight, Figure 2(f) shows tomato healthy, Figure 2(g) shows tomato late blight, Figure 2(h) shows tomato leaf mold, Figure 2(i) shows tomato septoria leaf spot, and Figure 2(j) shows tomato spider mites two spotted spider mite. Each category exhibits distinct visual characteristics such as lesions, discoloration, deformation patterns, or the absence of symptoms in healthy leaves. This detailed categorization ensures dataset suitability for multi-class classification tasks and enables the model to learn subtle inter-class differences effectively. Specifically, this research focused on a subset of the dataset containing 5,000 labeled images of tomato leaves with size 256×256 pixel. These images are distributed across 10 distinct categories, each representing a unique condition or disease affecting tomato plants. The dataset includes both diseased and healthy leaves, providing a balanced representation of the challenges faced in tomato crop management. The images are captured in various lighting conditions and orientations, adding diversity and realism to the dataset, which is crucial for training a robust classification model.

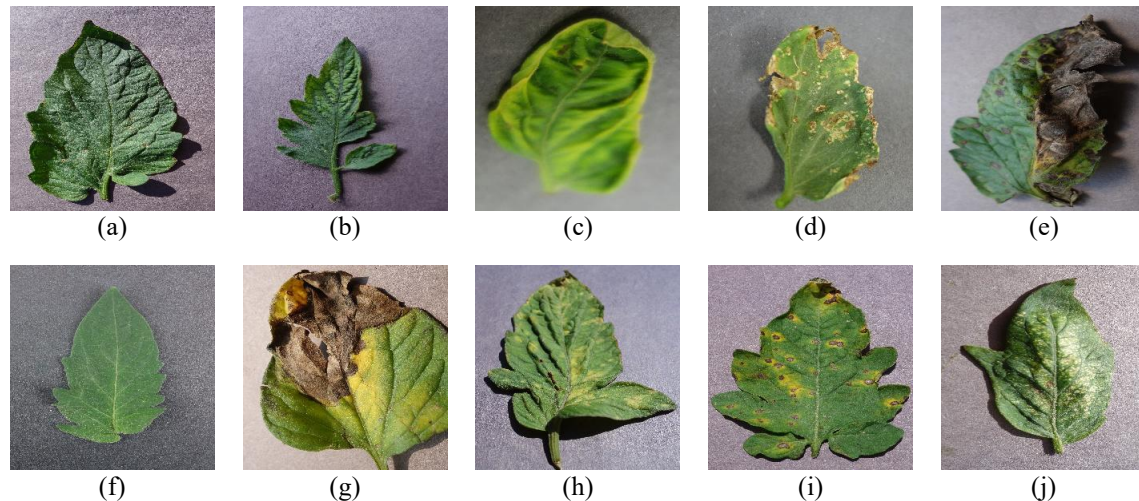


Figure 2. Sample dataset tomato leaf disease of (a) tomato target spot, (b) tomato mosaic virus, (c) tomato yellow leaf curl virus, (d) tomato bacterial spot, (e) tomato early blight, (f) tomato healthy, (g) tomato late blight, (h) tomato leaf mold, (i) tomato septoria leaf spot, and (j) tomato spider mites two spotted spider mite

2.2. Exploration data analysis

The EDA phase was conducted to gain deeper insights into the distribution and characteristics of the dataset [25]. One of the key analyses involved examining the number of images available for each disease category. This analysis revealed slight variations in the dataset class distribution, as depicted in Figure 3. While the dataset is generally well-balanced, some categories have marginally fewer samples, which could influence the model learning process and overall performance. Understanding these discrepancies is crucial to implementing strategies like data augmentation or class weighting to mitigate potential biases in the model.

The distribution of images across disease categories is illustrated in Figure 3. The dataset shows generally balanced class frequencies, where most categories, including leaf mold as shown in Figure 3(a), target spot as shown in Figure 3(b), healthy leaves as shown in Figure 3(c), and late blight as shown in Figure 3(d) contain approximately 513–515 images. Additional categories such as spider mites as shown in Figure 3(e), early blight as shown in Figure 3(f), bacterial spot as shown in Figure 3(g), septoria leaf spot as shown in Figure 3(h), and yellow leaf curl virus as shown in Figure 3(i) also present comparable distributions. The tomato mozaic virus class as shown in Figure 3(j) contains fewer samples 373 images, indicating mild class imbalance that motivates the use of preprocessing and data augmentation to support robust learning and generalization.

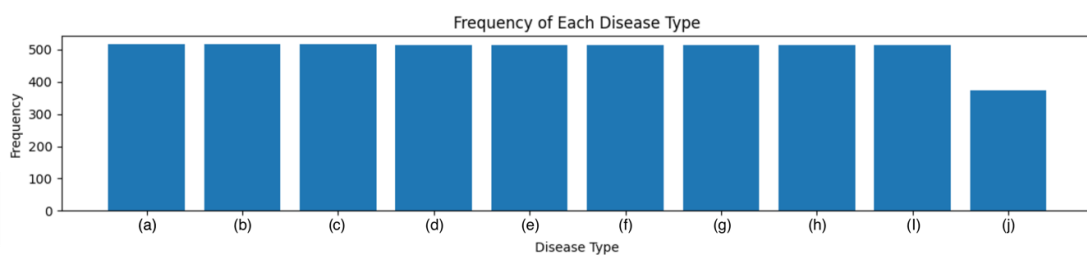


Figure 3. Frequency of each tomato disease type for (a) tomato leaf mold, (b) tomato target spot, (c) tomato healthy, (d) tomato late blight, (e) tomato spider mites two spotted spider mite, (f) tomato early blight, (g) tomato bacterial spot, (h) tomato septoria leaf spot, (i) tomato yellow leaf curl virus, and (j) tomato mozaic virus

Table 1 presents the final dataset structure, consisting of three columns: file, DiseaseID, and disease type. The file column contains relative image paths, the DiseaseID column assigns numerical labels (0–9) for efficient model training, and the disease type column provides interpretable class names. Sample entries shown in Table 1 demonstrate images from the tomato healthy category, illustrating a well-organized dataset format that supports seamless integration into the machine learning workflow.

Table 1. Sample final dataset format

No.	File	DiseaseID	Disease type
4995	Tomato_Healthy/8cc11f24-0b79-4eda-8a09-9c776e0...	9	Tomato healthy
4996	Tomato_Healthy/8d78a34f-1777-4bd8-a925-795c57c...	9	Tomato healthy
4997	Tomato_Healthy/9b8f108a-010b-4b8b-98ea-b30cbf6...	9	Tomato healthy
4998	Tomato_Healthy/7beda862-3964-4839-9d08-b9b3fa2...	9	Tomato healthy
4999	Tomato_Healthy/7d76aa96-b29e-4a60-a3e7-bd489f5...	9	Tomato healthy

2.3. Data preprocessing

The image resizing process is illustrated in Figure 4. The original image resolution of 256×256 pixels shown in Figure 4(a) was reduced to 64×64 pixels as presented in Figure 4(b). This resizing was adopted as a deliberate trade-off to reduce computational complexity, GPU memory usage, and training time while maintaining essential visual characteristics required for classification. Uniform resizing ensures consistent input dimensions and stable deep learning optimization.

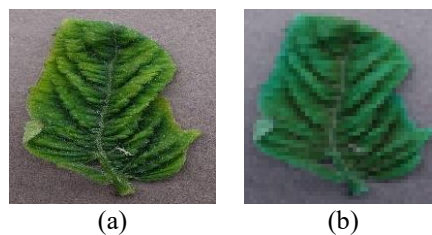


Figure 4. Resizing image of (a) before resize and (b) after resize

Although DenseNet-121 was originally pre-trained on ImageNet using 224×224 inputs, lower-resolution inputs can remain effective when disease characteristics are primarily defined by global color distribution, lesion patterns, and structural variations rather than extremely fine-grained textures [26]. Under controlled dataset conditions, the resized images preserve key discriminative visual cues required for tomato leaf disease classification, while significantly reducing computational cost. It is acknowledged that reducing spatial resolution may limit the representation of subtle texture-level features. Therefore, the selected resolution should be interpreted as a computationally efficient configuration rather than an optimal universal setting. A comparative evaluation using higher input resolutions 128×128 and 224×224 is planned as future work to quantitatively assess the trade-off between computational efficiency and fine-grained feature preservation. This clarification emphasizes that the present study focuses on evaluating training strategy effectiveness rather than resolution optimization.

The dataset, consisting of 5,000 tomato leaf images distributed across 10 classes, was partitioned into training and testing sets using a stratified 80:20 split, yielding 4,000 training samples and 1,000 testing samples. This stratification preserved the class distribution in both subsets, ensuring a fair and unbiased evaluation of the model. To enhance dataset diversity and mitigate overfitting, data augmentation was applied exclusively to the training set using the ImageDataGenerator framework [27]. The applied transformations included random rotations up to 20 degrees, horizontal and vertical shifts up to 20%, shearing, zooming, and horizontal flipping. These augmentations emulate real-world variations in leaf appearance, such as changes in orientation, scale, and viewpoint, as illustrated in Figure 5, thereby improving the model's ability to generalize.

The parameter `fill_mode='nearest'` was used to preserve visual continuity in transformed images by filling newly generated pixel regions using neighboring pixel values. This prevents artificial artifacts that could negatively affect feature learning. Data leakage was avoided by performing dataset splitting prior to augmentation and ensuring that no identical or near-duplicate images appeared in both training and testing sets [28]. The testing dataset remained unchanged to provide an unbiased estimation of model generalization performance.

2.4. Model

During the model development phase, the training process was initiated using the DenseNet-121 architecture, as shown in Figure 6, which represents a modern CNN. This architecture is distinguished by its dense connectivity mechanism, where each layer is linked to all following layers, enabling efficient feature propagation and reuse [29]. This design promotes efficient feature reuse by allowing features learned in

earlier layers to be propagated throughout the network, thereby reducing redundancy and improving parameter efficiency [30]. Furthermore, dense connections facilitate gradient propagation during backpropagation, effectively mitigating the vanishing gradient problem commonly encountered in deep neural networks [31].



Figure 5. Result of augmentation

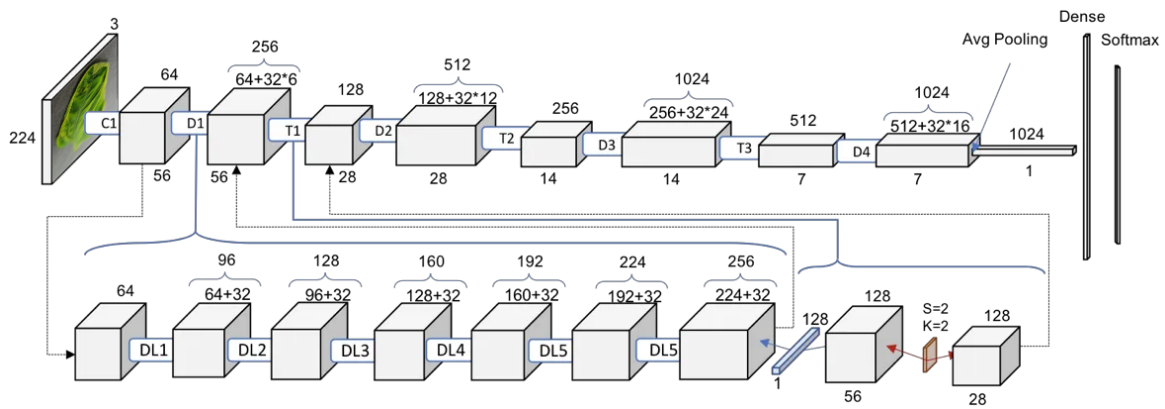


Figure 6. DenseNet-121 flow process

The architecture consists of 121 layers organized into four dense blocks separated by transition layers containing convolution, pooling, and batch normalization operations. Mathematically, the input to layer l can be represented as (1) [32].

$$x_l = H_l([x_0, x_1, \dots, x_{l-1}]) \tag{1}$$

Where H_l is the transformation (batch normalization, rectified linear unit (ReLU) activation, and convolution, and $[x_0, x_1, \dots, x_{l-1}]$ represents the concatenated outputs of all preceding layers. This mechanism allows the model to efficiently learn both low-level and high-level features simultaneously.

To accelerate convergence and leverage prior visual knowledge, DenseNet-121 pre-trained weights from the ImageNet dataset were employed [33]. Transfer learning provides a strong initialization by incorporating generic visual features such as edges, textures, and shapes commonly shared across image datasets [34]. The pre-trained network was subsequently fine-tuned to adapt feature representations specifically to tomato leaf disease characteristics while preserving generalizable visual patterns [35].

All experiments were conducted using TensorFlow v2.17.0 with the Keras API v3.4.1 in a Python 3.10 environment. Model training was performed on Google Colab with an NVIDIA Tesla T4 GPU. To ensure reproducibility, a fixed random seed was consistently applied across NumPy, TensorFlow, and Python’s random module during data splitting and model initialization.

The Adam optimizer was utilized for its adaptive learning rate capability and momentum-based update mechanism, which facilitate stable and efficient model convergence [36]. The initial learning rate was set to 1×10^{-3} , balancing convergence speed and training stability while preventing abrupt parameter updates. The detailed training configuration is summarized in Table 2. To ensure unbiased evaluation, the dataset was split using a stratified 80:20 train–test approach, with a validation subset derived from the training data to monitor learning and prevent overfitting. Although k-fold cross-validation was not applied due to computational constraints, consistent initialization, and controlled experimental settings were used to maintain result reliability. The training process was conducted in three stages: baseline training for 30 epochs, followed by 20 epochs with data augmentation to improve generalization, and a final 20-epoch fine-tuning stage by unfreezing all layers, resulting in improved robustness and classification performance.

Table 2. Training hyperparameters and experimental setup

Parameter	Value
Input image size	64×64 pixels
Batch size	32
Optimizer	Adam
Initial learning rate	1,00E-03
Learning rate strategy	Constant
Epochs (baseline training)	30
Epochs (with augmentation)	20
Epochs (fine-tuning)	20
Loss function	Categorical cross-entropy
Data augmentation	Rotation ($\pm 20^\circ$), width/height shift (20%), zoom, shear, and horizontal flip
Augmentation applied to	Training set only

2.5. Model evaluation

Model evaluation was initiated by measuring performance on the training dataset using metrics such as accuracy, precision, recall, and F1-score to ensure effective feature learning across all ten tomato leaf disease classes. Training loss and accuracy trends were monitored to ensure stable learning and to detect potential underfitting or overfitting. Subsequently, model generalization was evaluated on the testing dataset containing unseen samples. This evaluation analyzed performance gaps between training and testing results to identify overfitting or underfitting and emphasized class-wise accuracy to ensure balanced classification across all disease categories.

Based on these evaluations, retraining was performed using DenseNet-121 with data augmentation and fine-tuning to enhance robustness and adaptability [14]. Data augmentation was used to simulate realistic variations in the data, while fine-tuning enabled the model to better learn domain-specific disease characteristics. A confusion matrix was then utilized for detailed class-level evaluation by summarizing true positives, true negatives, false positives, and false negatives, which serve as the basis for calculating precision, recall, and F1-score, as well as for identifying strengths and weaknesses across classes [14].

The confusion matrix also served as a diagnostic tool to uncover patterns of misclassification. For instance, frequent misclassification between similar classes, such as tomato late blight and tomato early blight, might indicate overlapping visual features that require additional augmentation or fine-tuning. Furthermore, a normalized confusion matrix was used to visualize the classification accuracy for each category as percentages, making it easier to identify underperforming classes. This matrix, when combined with the derived metrics, linked the qualitative insights with quantitative evidence, and offering actionable strategies for improvement. It also validated the model's robustness by demonstrating consistent performance across all disease categories, essential for ensuring reliability in real-world tomato leaf disease detection scenarios.

3. RESULTS AND DISCUSSION

Figure 7 shows the results demonstrate the influence of preprocessing and training strategies on the performance of DenseNet-121 for tomato leaf disease classification. Initially, images were resized from 256×256 to 64×64 pixels to reduce computational cost while retaining key visual features. Using the Adam optimizer with a minimum learning rate of 1×10^{-3} , the accuracy curve shown in Figure 7(a), indicates that the model achieved an initial validation accuracy of 75.52%, while the loss progression in Figure 7(b) shows a corresponding validation loss of 37.77%. This baseline performance demonstrates the ability of DenseNet-121 to extract features effectively even from resized images, albeit with room for improvement.

To enhance model performance and reduce overfitting, data augmentation was incorporated during training. The applied techniques included random rotations, translations, shearing, zooming, and horizontal

flipping. These transformations introduced variability into the dataset, mimicking real-world conditions and improving the model’s ability to generalize to unseen data. Figure 8 shows the result after add data augmentation. The accuracy curve in Figure 8(a) shows a significant improvement to 95.31%, while the loss curve in Figure 8(b) decreases to 11.74%. This marked improvement emphasizes the importance of data diversity in enhancing the model's ability to generalize.

In the final stage, fine-tuning was performed by unfreezing selected layers of the pre-trained DenseNet-121 model and retraining it using the augmented dataset. This process enabled the network to adjust its learned representations to the specific characteristics of tomato leaf diseases, resulting in more discriminative and task-relevant features. Figure 9 shows the result after fine-tuning. The accuracy curve in Figure 9(a) demonstrates further improvement, reaching 98.44%, while the loss curve in Figure 9(b) decreases to 4.72%. The substantial improvement in both accuracy and loss underscores the effectiveness of transfer learning and the critical role of fine-tuning in achieving optimal performance.

Analysis of the training and validation curves indicates that data augmentation and fine-tuning significantly reduced overfitting. The baseline model showed a noticeable gap between training and validation accuracy, suggesting limited generalization performance. This gap progressively narrowed after data augmentation, accompanied by smoother loss convergence, and further diminished following fine-tuning, where both curves converged more closely, demonstrating improved training stability and generalization performance. Comparative results across the three training stages confirm the cumulative benefits of each enhancement. Model accuracy increased from 75.52%–98.44%, while loss decreased from 37.77%–4.72%, highlighting the effectiveness of data augmentation and fine-tuning in strengthening DenseNet-121 feature learning.

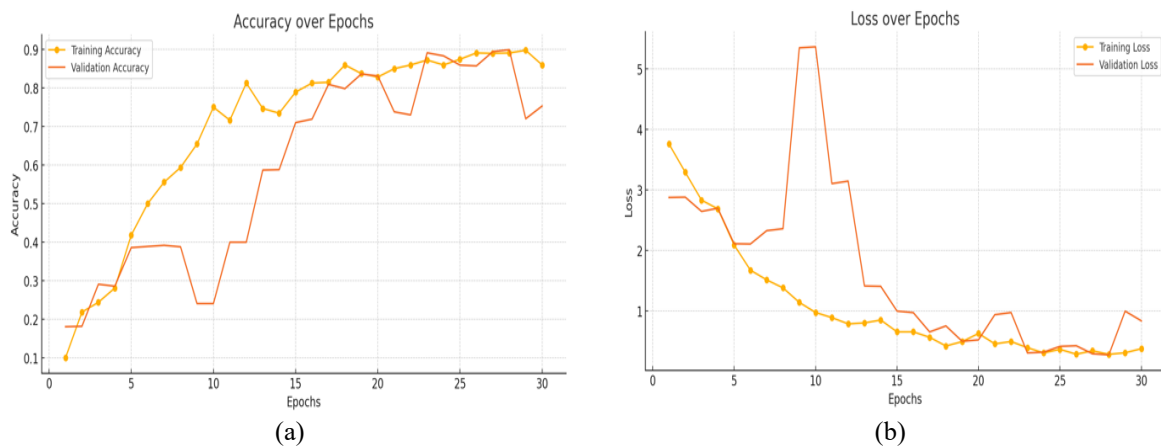


Figure 7. Result use DenseNet-121 of (a) accuracy and (b) loss

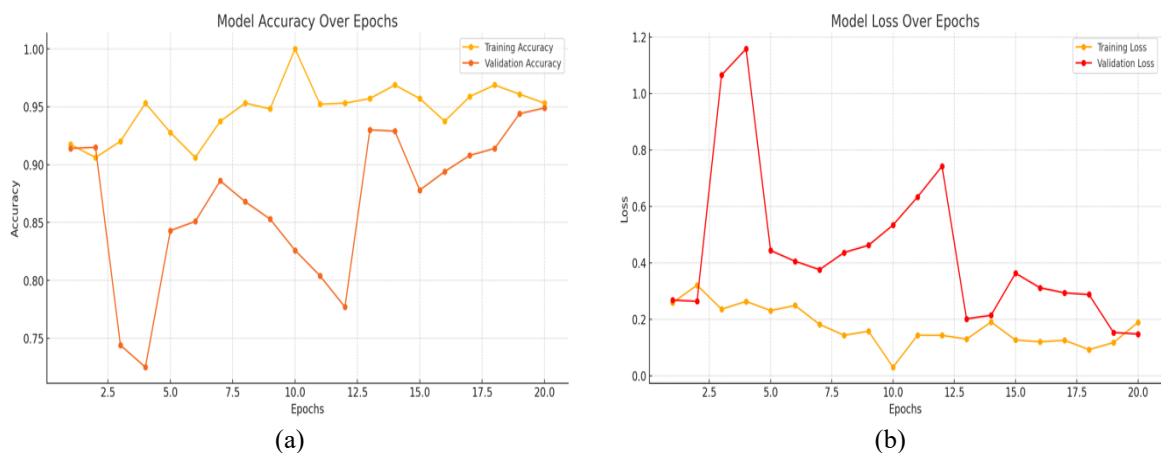


Figure 8. Result after add data augmentation of (a) accuracy and (b) loss

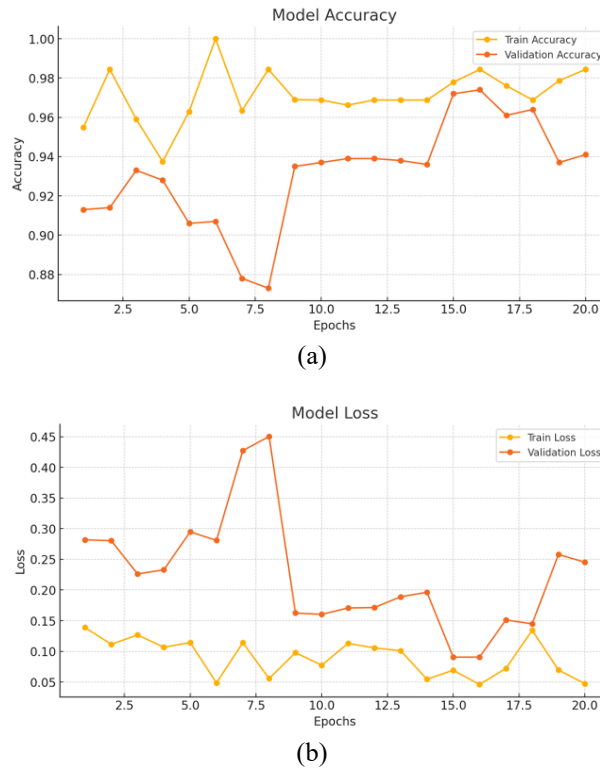


Figure 9. Result after fine-tuning of (a) accuracy and (b) loss

Figure 10 presents the accuracy results obtained from each fold in the 5-fold stratified cross-validation. The model demonstrates consistently high performance across all folds, with accuracy values of 98.38%-fold 1, 98.51%-fold 2, 98.47%-fold 3, 98.37%-fold 4, and 98.47%-fold 5. The variation between folds is minimal, indicating stable model behavior and strong generalization capability across different data partitions. The average accuracy of 98.44%, as indicated by the dashed line, confirms the robustness and reliability of the proposed DenseNet-121 model with data augmentation and fine-tuning.

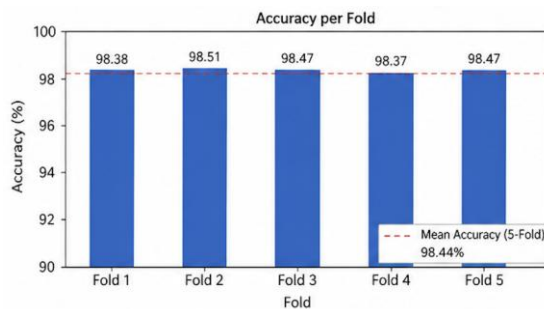


Figure 10. Accuracy per fold

The reported performance was obtained under a controlled experimental configuration using fixed initialization settings and identical training conditions. A fixed random seed was applied during dataset splitting and model initialization to reduce stochastic variability during optimization. Consequently, the reported accuracy reflects a reproducible outcome under the defined experimental setup rather than a stochastic best-case observation.

Although repeated run statistical analysis and confidence interval estimation could further quantify performance variability, the primary objective of this study is to evaluate the effectiveness of a progressive training strategy within a single consistent experimental pipeline. The staged improvements observed across

baseline training, augmentation, and fine-tuning provide internal comparative evidence indicating that the observed performance gains are primarily associated with the proposed training procedure.

These results indicate that a structured pipeline integrating preprocessing, data augmentation, and fine-tuning is highly effective for tomato leaf disease classification. The achieved high accuracy and low loss demonstrate the model's reliability for practical agricultural applications. Future work may incorporate advanced approaches, such as attention mechanisms or ensemble methods, to further enhance performance and robustness under varying conditions.

The per-class evaluation results in Table 3 demonstrate that the proposed model achieves consistently high performance across most tomato leaf disease categories, with an overall accuracy of 98.44% after fine-tuning. Tomato mozaic virus achieves perfect recall 100% despite having fewer samples, indicating that its visual characteristics are highly distinguishable by the model. However, lower performance is observed in tomato spider mites and tomato target spot, as reflected in their reduced precision and recall values. This behavior is further supported by the confusion matrix, where these classes exhibit higher misclassification rates. The primary reason lies in the strong visual similarity between these categories, particularly in terms of texture patterns, lesion distribution, and color variations, which overlap significantly and make discrimination more challenging. In contrast, classes such as tomato healthy and tomato yellow leaf curl virus show near-perfect performance due to more distinct visual features. These findings highlight that while the model is robust overall, classification errors are mainly concentrated in visually similar disease categories, suggesting that additional feature enhancement or higher-resolution inputs could further improve class-level discrimination.

Data augmentation and fine-tuning particularly benefited classes with high visual variability, such as tomato early blight and tomato septoria leaf spot, where improved recall indicates enhanced generalization. Conversely, misclassification remained more frequent between visually similar disease pairs, notably early blight versus late blight and target spot versus spider mites. Although DenseNet-121 was originally designed for 224×224 inputs, resizing images to 64×64 was adopted to reduce computational cost and support deployment on resource-constrained platforms. Despite potential loss of fine-grained details, the model maintained high precision and recall, indicating that global color and structural features remain informative under controlled conditions. However, further validation using real-field data is needed to ensure robustness in diverse environments.

Table 3. Per-class performance of DenseNet-121

Disease class	Precision (%)	Recall (%)	F1-score (%)	Support
Tomato bacterial spot	95	93	94	87
Tomato early blight	91	97	94	96
Tomato late blight	100	93	97	122
Tomato leaf mold	100	90	95	92
Tomato septoria leaf spot	87	98	92	94
Tomato spider mites (two-spotted spider mite)	99	81	89	108
Tomato target spot	81	96	88	108
Tomato yellow leaf curl virus	99	94	97	109
Tomato mozaic virus	99	100	99	82
Tomato healthy	95	100	98	102

This study focuses on analyzing the impact of preprocessing, data augmentation, and fine-tuning within a fixed DenseNet-121 backbone rather than benchmarking multiple CNN architectures. The baseline configuration, DenseNet-121 trained without augmentation and fine-tuning served as an internal reference, enabling isolation of performance improvements attributable to the proposed training strategy. It should also be noted that model generalization was evaluated only within a single benchmark dataset, reflecting intra-dataset performance rather than cross-dataset robustness. In addition, interpretability analysis was beyond the scope of this study.

Table 4 presents a comparison between the proposed approach and several previous studies on plant leaf disease classification using different datasets, preprocessing strategies, and deep learning architectures. Prior works have explored a wide range of models, including lightweight architectures such as microcontroller unit network (MCUNet) [28], region-based approaches like region-based convolutional neural network (R-CNN) [26] and mask region-based convolutional neural network (MRCNN) combined with optimization techniques [37], as well as conventional CNN including VGG-16, VGG-19 [38], AlexNet [22], and hybrid ResNet-50–support vector machine (SVM) frameworks [21]. Reported accuracies vary from 91.32%–98.00%, depending on dataset complexity, number of classes, and preprocessing configurations.

Most existing studies employ image resizing and data augmentation as essential preprocessing steps to enhance model generalization. However, performance differences are strongly influenced by dataset

characteristics and classification complexity, particularly the number of disease categories. For example, several studies achieving high accuracy were conducted on datasets containing fewer classes, which generally reduces classification difficulty.

In comparison, the proposed DenseNet-121-based approach achieves an accuracy of 98.44% on the PlantVillage dataset with ten disease classes, incorporating image resizing, data augmentation, and fine-tuning within a transfer learning framework. The results demonstrate that optimizing the training strategy and leveraging dense feature reuse can achieve competitive performance even with increased class complexity. These findings suggest that methodological refinement through progressive training stages can be as influential as architectural variation in improving classification performance for plant disease detection tasks.

Table 4. Comparison result

Author	Dataset	Number of classes	Pre-processing	Model architecture	Accuracy (%)
[28]	PlantVillage	4	Image resizing, and data augmentation	MCUNet	91.32
[37]	AI challenger	8	Noise removal, data annotation, data augmentation, and data resize	MRCNN, Ant colony optimizer	97.66
[26]	PlantVillage	6	Grayscale image and image resizing	R-CNN	96.73
[38]	Kaggle	11	Data augmentation	VGG-16, VGG-19	95.00
[22]	Kaggle	10	-	AlexNet	98.00
[21]	PlantVillage	4	Image resizing	ResNet-50, SVM	93.33
Our proposed	PlantVillage	10	Image resizing, data augmentation, and fine-tuning	DenseNet-121	98.44

4. CONCLUSION

This study demonstrates that integrating image preprocessing, data augmentation, and fine-tuning effectively enhances DenseNet-121 for tomato leaf disease classification. Resizing images to 64×64 pixels provided a computationally efficient baseline while preserving essential visual characteristics, achieving an initial accuracy of 75.52%. The introduction of data augmentation improved generalization performance to 95.31%, and subsequent fine-tuning further increased accuracy to 98.44% with reduced loss, highlighting the cumulative impact of progressive training stages. These results indicate that optimizing training strategies within a transfer learning framework can produce a stable and high-performing classification model under controlled experimental conditions. Although the findings show promising potential for automated plant disease analysis, further validation using real-field datasets is required to confirm robustness in practical environments. Future work will investigate higher input resolutions, alternative architectures, and explainable AI approaches to further improve generalization capability and real-world applicability.

FUNDING INFORMATION

This work was supported by Department of Research and Community Service, Pelita Bangsa University, for supporting and funding this research. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Sufajar Butsianto	✓	✓		✓	✓			✓	✓		✓	✓	✓	✓
Anggi Muhammad Rifa'i	✓		✓	✓		✓	✓	✓		✓	✓			

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

CONFLICT OF INTEREST STATEMENT

The author declares no conflict of interest.

INFORMED CONSENT

We have obtained informed consent from all individuals included in this study.

DATA AVAILABILITY

The data supporting the findings of this study are openly available at reference [24]. The dataset is publicly accessible and available without restriction for academic and research purposes.





REFERENCES

- [1] A. Sridhar, A. Balakrishnan, M. M. Jacob, M. Sillanpää, and N. Dayanandan, "Global impact of COVID-19 on agriculture: role of sustainable agriculture and digital farming," *Environmental Science and Pollution Research*, vol. 30, no. 15, pp. 42509–42525, Mar. 2022, doi: 10.1007/s11356-022-19358-w.
- [2] H. Tian, T. Wang, Y. Liu, X. Qiao, and Y. Li, "Computer vision technology in agricultural automation—a review," *Information Processing in Agriculture*, vol. 7, no. 1, pp. 1–19, Mar. 2020, doi: 10.1016/j.inpa.2019.09.006.
- [3] G. Mohyuddin, M. A. Khan, A. Haseeb, S. Mahpara, M. Waseem, and A. M. Saleh, "Evaluation of machine learning approaches for precision farming in smart agriculture system: a comprehensive review," *IEEE Access*, vol. 12, pp. 60155–60184, 2024, doi: 10.1109/ACCESS.2024.3390581.
- [4] V. Kakani, V. H. Nguyen, B. P. Kumar, H. Kim, and V. R. Pasupuleti, "A critical review on computer vision and artificial intelligence in food industry," *Journal of Agriculture and Food Research*, vol. 2, Dec. 2020, doi: 10.1016/j.jafr.2020.100033.
- [5] J. Rane, Ö. Kaya, S. K. Mallick, and N. L. Rane, "Smart farming using artificial intelligence, machine learning, deep learning, and ChatGPT: applications, opportunities, challenges, and future directions," in *Generative Artificial Intelligence in Agriculture, Education, and Business*, Deep Science Publishing, 2024, doi: 10.70593/978-81-981271-7-4_6.
- [6] T. T. Nguyen *et al.*, "Monitoring agriculture areas with satellite images and deep learning," *Applied Soft Computing*, vol. 95, 2020, doi: 10.1016/j.asoc.2020.106565.
- [7] L. Li, S. Zhang, and B. Wang, "Plant disease detection and classification by deep learning—a review," *IEEE Access*, vol. 9, pp. 56683–56698, 2021, doi: 10.1109/ACCESS.2021.3069646.
- [8] S. Mohammed *et al.*, "The effects of data quality on machine learning performance on tabular data," *Information Systems*, vol. 132, Jul. 2025, doi: 10.1016/j.is.2025.102549.
- [9] N. A. Sharma, A. B. M. S. Ali, and M. A. Kabir, "A review of sentiment analysis: tasks, applications, and deep learning techniques," *International Journal of Data Science and Analytics*, vol. 19, no. 3, pp. 351–388, Apr. 2025, doi: 10.1007/s41060-024-00594-x.
- [10] A. Pawar, R. Amrutkar, S. Somani, and G. S. Mundada, "Plant leaf disease detection using CNN," in *1st International Conference on Deep Learning, IoT, Drone Technology, Smart Cities, and Applications*, 2024, doi: 10.1063/5.0229141.
- [11] I. Attri, L. K. Awasthi, T. P. Sharma, and P. Rathee, "A review of deep learning techniques used in agriculture," *Ecological Informatics*, vol. 77, Nov. 2023, doi: 10.1016/j.ecoinf.2023.102217.
- [12] M. H. Saleem, J. Potgieter, and K. M. Arif, "Automation in agriculture by machine and deep learning techniques: a review of recent developments," *Precision Agriculture*, vol. 22, no. 6, pp. 2053–2091, Dec. 2021, doi: 10.1007/s11119-021-09806-x.
- [13] W. Ding, M. A.-Basset, I. Alrashdi, and H. Hawash, "Next generation of computer vision for plant disease monitoring in precision agriculture: a contemporary survey, taxonomy, experiments, and future direction," *Information Sciences*, vol. 665, Apr. 2024, doi: 10.1016/j.ins.2024.120338.
- [14] A. M. Rifai, S. Raharjo, E. Utami, and D. Ariatmanto, "Analysis for diagnosis of pneumonia symptoms using chest X-ray based on MobileNetV2 models with image enhancement using white balance and contrast limited adaptive histogram equalization (CLAHE)," *Biomedical Signal Processing and Control*, vol. 90, Apr. 2024, doi: 10.1016/j.bspc.2023.105857.
- [15] Z. Li, D. Wang, T. Zhu, Y. Tao, and C. Ni, "Review of deep learning-based methods for non-destructive evaluation of agricultural products," *Biosystems Engineering*, vol. 245, pp. 56–83, Sep. 2024, doi: 10.1016/j.biosystemseng.2024.07.002.
- [16] H. N. Ngugi, A. E. Ezugwu, A. A. Akinyelu, and L. Abualigah, "Revolutionizing crop disease detection with computational deep learning: a comprehensive review," *Environmental Monitoring and Assessment*, vol. 196, no. 3, Mar. 2024, doi: 10.1007/s10661-024-12454-z.
- [17] T. S. Arulananth, S. W. Prakash, R. K. Ayyasamy, V. P. Kavitha, P. G. Kuppusamy, and P. Chinnasamy, "Classification of paediatric pneumonia using modified DenseNet-121 deep-learning model," *IEEE Access*, vol. 12, pp. 35716–35727, 2024, doi: 10.1109/ACCESS.2024.3371151.
- [18] X. Zhang, X. Chen, W. Sun, and X. He, "Vehicle re-identification model based on optimized DenseNet121 with joint loss," *Computers, Materials & Continua*, vol. 67, no. 3, pp. 3933–3948, 2021, doi: 10.32604/cmc.2021.016560.
- [19] B. Li, "Facial expression recognition by DenseNet-121," in *Multi-Chaos, Fractal and Multi-Fractional Artificial Intelligence of Different Complex Systems*, Elsevier, 2022, pp. 263–276, doi: 10.1016/B978-0-323-90032-4.00019-5.
- [20] A. De, N. Mishra, and H.-T. Chang, "An approach to the dermatological classification of histopathological skin images using a hybridized CNN-DenseNet model," *PeerJ Computer Science*, vol. 10, Feb. 2024, doi: 10.7717/peerj-cs.1884.
- [21] H. M. Chong, X. Y. Yap, and K. S. Chia, "Effects of different pretrained deep learning algorithms as feature extractor in tomato plant health classification," *Pattern Recognition and Image Analysis*, vol. 33, no. 1, pp. 39–46, Mar. 2023, doi: 10.1134/S1054661823010017.
- [22] H.-C. Chen *et al.*, "AlexNet convolutional neural network for disease detection and classification of tomato leaf," *Electronics*, vol. 11, no. 6, Mar. 2022, doi: 10.3390/electronics11060951.
- [23] R. Rosalina, G. Sahuri, and H. Desriva, "A mobile-optimized convolutional neural network approach for real-time batik pattern recognition," *IAES International Journal of Artificial Intelligence*, vol. 13, no. 3, pp. 3018–3027, Sep. 2024, doi: 10.11591/ijai.v13.i3.pp3018-3027.





- [24] D. P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," Apr. 2016, *arXiv: 1511.08060*.
- [25] G. A. L.-Ramírez, A. A.-Zavala, and C. V.-Rosales, "Exploratory data analysis for path loss measurements: unveiling patterns and insights before machine learning," *IEEE Access*, vol. 12, pp. 62279–62295, 2024, doi: 10.1109/ACCESS.2024.3394904.
- [26] H. S. Nagamani and H. Sarojadevi, "Tomato leaf disease detection using deep learning techniques," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 1, 2022, doi: 10.14569/IJACSA.2022.0130138.
- [27] A. A. Khan, O. Chaudhari, and R. Chandra, "A review of ensemble learning and data augmentation models for class imbalanced problems: combination, implementation and evaluation," *Expert Systems with Applications*, vol. 244, Jun. 2024, doi: 10.1016/j.eswa.2023.122778.
- [28] Y. Deng *et al.*, "An effective image-based tomato leaf disease segmentation method using MC-UNet," *Plant Phenomics*, vol. 5, 2023, doi: 10.34133/plantphenomics.0049.
- [29] A. Raza, M. S. Alshehri, S. Almakdi, A. A. Siddique, M. Alsulami, and M. Alhaisoni, "Enhancing brain tumor classification with transfer learning: leveraging DenseNet121 for accurate and efficient detection," *International Journal of Imaging Systems and Technology*, vol. 34, no. 1, Jan. 2024, doi: 10.1002/ima.22957.
- [30] J. Potsangbam and S. S. Devi, "Classification of breast cancer histopathological images using transfer learning with DenseNet121," *Procedia Computer Science*, vol. 235, pp. 1990–1997, 2024, doi: 10.1016/j.procs.2024.04.188.
- [31] N. Cinar, A. Ozcan, and M. Kaya, "A hybrid DenseNet121-UNet model for brain tumor segmentation from MR Images," *Biomedical Signal Processing and Control*, vol. 76, Jul. 2022, doi: 10.1016/j.bspc.2022.103647.
- [32] Y. V. K. Sundaram, and K. Kanagaraj, "Soybean leaf disease classification using enhanced DenseNet121," in *2024 10th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Mar. 2024, pp. 1515–1520, doi: 10.1109/ICACCS60874.2024.10717087.
- [33] G. Mishra, P. Gupta, and R. Tanwar, "Target recognition using pre-trained convolutional neural networks and transfer learning," *Procedia Computer Science*, vol. 235, pp. 1445–1454, 2024, doi: 10.1016/j.procs.2024.04.136.
- [34] K. S. Lakshmi and B. Sargunam, "Exploration of AI-powered DenseNet121 for effective diabetic retinopathy detection," *International Ophthalmology*, vol. 44, no. 1, Feb. 2024, doi: 10.1007/s10792-024-03027-7.
- [35] P. V. *et al.*, "Improved tomato leaf disease classification through adaptive ensemble models with exponential moving average fusion and enhanced weighted gradient optimization," *Frontiers in Plant Science*, vol. 15, May 2024, doi: 10.3389/fpls.2024.1382416.
- [36] E. Hassan, M. Y. Shams, N. A. Hikal, and S. Elmougy, "The effect of choosing optimizer algorithms to improve computer vision tasks: a comparative study," *Multimedia Tools and Applications*, vol. 82, no. 11, pp. 16591–16633, May 2023, doi: 10.1007/s11042-022-13820-0.
- [37] R. R. Kumar, J. Athimoolam, A. Appathurai, and S. Rajendiran, "Novel segmentation and classification algorithm for detection of tomato leaf disease," *Concurrency and Computation: Practice and Experience*, vol. 35, no. 12, May 2023, doi: 10.1002/cpe.7674.
- [38] S. G. Paul *et al.*, "A real-time application-based convolutional neural network approach for tomato leaf disease classification," *Array*, vol. 19, Sep. 2023, doi: 10.1016/j.array.2023.100313.

BIOGRAPHIES OF AUTHORS



Sufajar Butsianto     a dedicated lecturer in the Department of Informatics Engineering at Pelita Bangsa University, earned a master's degree from Budi Luhur University Jakarta. His research areas include machine learning, image processing, computer vision, and software engineering particularly in industrial information system. He has published several papers in international journals and conferences and he currently a researcher at Pelita Bangsa University. He can be contacted at email: sufajar@pelitabangsa.ac.id.



Anggi Muhammad Rifa'i     a dedicated lecturer in the Department of Informatics Engineering at Pelita Bangsa University, with expertise in artificial intelligence, earned a master's degree from AMIKOM University Yogyakarta and he is currently studying for his doctoral studies at IPB University. His research areas include data science, machine learning, deep learning, image processing, and computer vision, particularly in the field of bioinformatics. He has published several papers in international journals and conferences and is currently a researcher at Pelita Bangsa University. He can be contacted at email: anggimuhammad@pelitabangsa.ac.id.